



UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS

FACULTAD 7

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

**MÓDULO DE GESTIÓN TECNOLÓGICA PARA EL SISTEMA
DE GESTIÓN PARA INGENIERÍA CLÍNICA Y
ELECTROMEDICINA (SIGICEM) V2.0**

AUTORES: Gloria Madelaine Fernández Pérez
Daymer Rodríguez Fillad

TUTORES: MSc. Zoila Esther Morales Tabares
Ing. Ranniel Rivero Sevilla

La Habana, mayo del 2013
“Año 55 de la Revolución”

*Nunca consideres el estudio como una obligación sino
como una oportunidad para penetraren el bello
y maravilloso mundo del saber.*

Albert Einstein

DATOS DE CONTACTO

Nombre y Apellidos: Zoila Esther Morales Tabares

Graduada de Ingeniera en Ciencias Informáticas en el curso 2007-2008. Ha impartido las asignaturas de Administración de Empresas y Contabilidad y Finanzas. Se ha desempeñado como tutora de Tesis de Enseñanza Politécnica; así como, cotutora y oponente en el curso 2010-2011 de varias tesis de grado. Realizó en el curso 2010-2011 la discusión de Tesis de Maestría en Informática Aplicada Actualmente, profesora de la Facultad 7 y adjunta al Departamento Sistemas Especializados en Salud del Centro de Informática Médica.

Correo electrónico: zemorales@uci.cu

Nombre y Apellidos: Ranniel Rivero Sevilla

Graduado de Ingeniero en Ciencias Informáticas en el curso 2007-2008. Líder de desarrollo del proyecto SACCEM durante un año. Ha impartido las asignaturas Práctica Profesional, Inteligencia Artificial, Historia de la Informática y Segundo Perfil. Se ha desempeñado como tribunal y tutor de varias tesis de grado. Actualmente, profesor de la Facultad 7 y Líder del proyecto Sistema de Gestión para Ingeniería Clínica y Electromedicina.

Correo electrónico: rrivero@uci.cu

Agradecimientos

A mi papá por apoyarme incondicionalmente durante estos cinco años y ser mi ejemplo a seguir.

A mi mamá que a pesar de todas las dificultades siempre ha estado a mi lado cuando la he necesitado.

A mis dos hermanas Elaine y Elenita, por los momentos de alegría que hemos pasado juntas y por hacer que me esfuerce cada día por ser mejor profesional y personalmente.

A mis dos queridos abuelos Bertha y René por preocuparse por mí en todo momento y velar que yo tenga un futuro mejor.

A mi amigo, novio y compañero de tesis Daymer, por estar a mi lado en todo momento y darme fuerzas para seguir adelante cuando más las he necesitado; a sus familiares por ayudarme en todo y acogerme como un miembro más de la familia.

A mis amistades que de una forma u otra me han ayudado a lo largo de la carrera, a Daniellys y Alexis por ser de las primeras personas que confiaron en mí y ayudaron en todo momento; a mi compañero de viaje Abelito, a la Zula, Claudia, Yeni y Arle por sus consejos, y a los chicos del grupo por sus críticas constructivas y no tan constructivas pero que han hecho de mí una mejor persona.

A Zoila y Ranniel por confiar en nosotros y habernos ayudado en todo momento incondicionalmente.

A la profesora Idayana por ayudarme en lo que pudo cuando me hizo falta.

Gloria M. Fernández

A toda mi familia que siempre ha sabido apoyarme en cada una de las decisiones que he tomado y han estado siempre que los he necesitado.

A mi mamá que me ha dado fuerzas para seguir adelante y que siempre ha confiado en mí.

A mi papá que ha sabido respetar mis decisiones en cada momento y de ayudarme cuando lo he necesitado.

A mi hermano Dayker que ha sido mucho más que un hermano y ha estado apoyándome en todo momento.

A mi hermano Frank que lo quiero muchísimo y he pasado buenos momentos compartiendo junto a él.

A mi tío Jorge que es muy especial para mí y me ha ayudado en todo lo que ha estado a su alcance.

A mí cuñada Mareilys que ha sido como una hermana y me ha aconsejado mucho en la vida.

A mi novia Gloria que la amo muchísimo y que ha estado a mi lado en momentos felices y difíciles y me inspira a cumplir mis sueños, a su familia por haber sido tan especial.

A mis amigos, a todos mis amigos, no solo los de la universidad, si no todos los que he conocido mucho antes y que no se han olvidado de mí, y con los que comparto en la actualidad, que no los mencionaré debido a que son muchos y en verdad ellos saben quiénes son. Entre estas personas se encuentran las muchachitas el aula que me han soportado durante 5 largos años.

A mis tutores Zoila y Ranniel que han sido más que amigos y profesores, al igual que el profesor Dennys.

Si alguien no se encuentra en estas líneas no se sienta mal, porque a pesar de todo también le agradezco.

Daymer Rodriguez

Dedicatoria

A mi papá por apoyarme y motivarme a ser una persona cada día mejor, a mi hermana Ela porque nunca ha dejado de preocuparse por mi bienestar y a mi novio Daymer porque sin su ayuda esto no hubiera sido posible, gracias.

Gloria

A mi mamá y mi papá que han estado esperando este momento más que cualquier persona en esta vida y a mi sobrino Dayker Enrique, para inspirarlo a tener una meta superior a la que yo tengo en la vida.

Daymer

RESUMEN

El Centro de Ingeniería Clínica y Electromedicina (CICEM), ha promovido el desarrollo de un sistema informático para llevar el control y mantenimiento de equipos médicos existentes en el país. Debido al cambio de misión y alcance que tuvo el centro, se reconsideró elaborar un sistema que contenga la mayor cantidad de funcionalidades, en aras de facilitar el trabajo de los especialistas del mismo. En la actualidad se encuentra en desarrollo el Sistema de Gestión para Ingeniería Clínica y Electromedicina (SIGICEM), en el cual debido a la reestructuración del centro, se desarrolló una nueva versión del módulo de Gestión Tecnológica para un mejor procesamiento de la información en el control de la tecnología médica.

Para el desarrollo del módulo se ha asimilado la plataforma de trabajo definida por el Centro de Informática Médica (CESIM) y el Departamento de Sistemas Especializados en la Salud (SES), conformando el siguiente ambiente de desarrollo: Proceso Unificado de Desarrollo como metodología de desarrollo apoyado en el Lenguaje Unificado de Modelado 2.1 y sustentado por el Visual Paradigm 6.4 como herramienta CASE para el diseño de los artefactos. Además, de la utilización de los *frameworks* Symfony 1.4 y ExtJS 2.2, soportado sobre una plataforma LAMP integrada por las tecnologías Ubuntu 10.04, Apache 2.2, MySQL 5.1 y PHP 5.1.

Palabras clave: equipos, gestión, sistema, tecnología.

TABLA DE CONTENIDO

| | |
|---|----|
| INTRODUCCIÓN | 6 |
| CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA | 11 |
| 1.1 Términos relacionados con la investigación..... | 11 |
| 1.2 Sistemas de gestión tecnológica | 12 |
| 1.3 Metodología, tecnologías y herramientas | 16 |
| CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA | 20 |
| 2.1 Gestión tecnológica de piezas de repuesto para la tecnología médica | 20 |
| 2.2 Modelo de Negocio..... | 20 |
| 2.3 Propuesta de solución | 27 |
| 2.4 Especificación de requisitos de software | 27 |
| 2.5 Modelo de casos de usos del sistema | 33 |
| CAPÍTULO 3. ANÁLISIS Y DISEÑO DEL SISTEMA | 40 |
| 3.1 Modelo de Análisis | 40 |
| 3.2 Modelo de Diseño..... | 45 |
| 3.3 Patrón arquitectónico..... | 50 |
| 3.4 Patrones de diseño..... | 51 |
| CAPÍTULO 4. IMPLEMENTACIÓN | 55 |
| 4.1 Modelo de Datos | 55 |
| 4.2 Modelo de Implementación..... | 60 |
| 4.3 Estándares y estilos de codificación | 62 |
| 4.4 Tratamiento de errores | 65 |
| 4.5 Seguridad..... | 66 |
| CONCLUSIONES | 67 |
| RECOMENDACIONES | 68 |
| REFERENCIAS BIBLIOGRÁFICAS..... | 69 |
| BIBLIOGRAFÍA..... | 72 |
| ANEXOS..... | 77 |

INTRODUCCIÓN

La informática constituye una ciencia que ha sido capaz de adentrarse en todas las esferas de la vida, siendo la gestión empresarial, actividad pionera en su aplicación y donde se ha utilizado con mayor profundidad; ya que fue en esta esfera donde se realizaron las primeras aplicaciones informáticas, las cuales estaban orientadas al control de actividades elementales como los inventarios y la contabilidad. Hoy aquellos primitivos sistemas de procesamiento de transacciones parecen pobres informativa y técnicamente, pues con la evolución de la ciencia, el desarrollo científico, técnico y tecnológico, han surgido sofisticados sistemas, así como otras aplicaciones útiles en internet que han llevado a la humanidad a tener un enorme desarrollo social. (1)

“Paralelamente, al empleo de las computadoras y la informática en general, y en estrecha vinculación con las mismas, aparece otro grupo de tecnologías, todas relacionadas con el procesamiento y transmisión de la información y el conocimiento: la telefonía móvil, la robótica, la automática desarrollada, entre otros. Al conjunto de esas tecnologías, conjunto extremadamente dinámico, pues constantemente surgen otras nuevas, se le ha llamado Tecnologías de la Información y las Comunicaciones (TIC)”. (1)

Las TIC han facilitado la interconexión entre las personas e instituciones a nivel mundial, eliminando barreras espaciales y temporales; logrando así, que casi todos los países del mundo establezcan diversos proyectos, políticas y estrategias para promover su uso y aprovechar los beneficios que estas ofrecen. Cuba, país subdesarrollado y actualmente bloqueado, no está al margen del desarrollo mundial, pues realiza grandes esfuerzos e invierte diversos recursos, para así aprovechar las ventajas que trae consigo la aplicación de las TIC en las diferentes áreas de la sociedad. Entre dichas áreas, se encuentra como prioridad el sector de la salud, en el cual el gobierno cubano ha puesto a disposición avanzadas tecnologías médicas.

Debido a la necesidad de garantizar la sostenibilidad de la tecnología existente en los hospitales del país, mediante el correcto asesoramiento de las adquisiciones de equipos y piezas, y la prestación de servicios técnicos durante el tiempo de explotación de cada tecnología; se creó el Centro Nacional de Electromedicina (CNE). Posteriormente, se comenzó a desarrollar un sistema automatizado que permitiría gestionar todas las actividades del centro e integrar las informaciones de carácter general del Sistema Nacional de Salud (SNS), al cual se le denominó: Sistema Integral de Gestión para Electromedicina

(SIGEM). Dicho sistema contó con diversas funcionalidades, tales como el aseguramiento técnico, el inventario, la gestión de la estadística técnica y la gestión de recursos humanos.

A partir de los cambios que enfrenta el CNE, los cuales vinculan la Ingeniería Clínica con la Electromedicina, se decidió cambiar el nombre actual del centro por uno que respondiera a la nueva actividad que se realizará; nombrándose, Centro de Ingeniería Clínica y Electromedicina (CICEM). Luego, debido al desarrollo de la entidad, surge la necesidad de crear un nuevo sistema que permita realizar la gestión de la información clínica y electromédica existente en las instituciones médicas del país. Al mismo se le denominó Sistema de Gestión para Ingeniería Clínica y Electromedicina (SIGICEM). (2)

En la actualidad, el sistema se encuentra en desarrollo por un grupo de trabajo perteneciente al Departamento Sistemas Especializados en la Salud (SES), el cual conjuntamente a otros departamentos integran el Centro de Informática Médica (CESIM), subordinándose este, a la Facultad 7 de la Universidad de las Ciencias Informáticas (UCI).

El SIGICEM, tiene como tarea, la informatización de los principales procesos que se llevan a cabo en el CICEM. Uno de los módulos que posee este sistema es el de Gestión Tecnológica, en el cual se gestiona todo el proceso referente al aseguramiento de piezas necesarias para la atención al equipamiento médico.

De acuerdo al reordenamiento empresarial llevado a cabo actualmente en Cuba, el CICEM, entidad presupuestada del SNS, entrará en el proceso de cambio de entidad a empresa. Por lo cual, las unidades de salud del territorio nacional pasarán a ser clientes de esta; propiciándose la generación de ingresos por servicios prestados. Es por ello, que se hace necesaria la modificación e inserción de nuevas funcionalidades al módulo de Gestión Tecnológica para llevar un control más efectivo de las piezas destinadas a la reparación de equipos médicos.

El módulo presenta además diversas dificultades, entre las que se evidencia la incapacidad de aprovechar la información que se gestiona en el módulo Gestión de Almacén; impidiéndole a los jefes de brigada* conocer la disponibilidad de las piezas y su posible existencia en otros almacenes, provocando la emisión de una solicitud de compra innecesaria.

* *Personas encargadas de satisfacer el pedido de piezas.*

Otros de los problemas existentes, están dados por la imposibilidad de gestionar las solicitudes de piezas que se realizan a través del módulo Orden de Servicio y que los especialistas técnicos conozcan la disponibilidad de las piezas en el momento de la solicitud, impidiendo la toma de decisiones oportunas, sin necesidad de espera del proceso de satisfacción de piezas solicitadas. Todo ello, debido a que los módulos Orden de Servicio y Gestión Tecnológica no se encuentran integrados funcionalmente.

Igualmente, ofrece la posibilidad a los especialistas técnicos de la realización de solicitudes no acorde con el equipo que se desea reparar, ya que no se cuentan con las funcionalidades necesarias para realizar el despiece de un equipo médico; función que ofrecería la posibilidad de relacionar los equipos médicos con las piezas que lo componen, así como, los equipos y piezas que componen un sistema*, evitando las solicitudes de aquellas piezas que no correspondan con dicho equipo.

A partir de la situación descrita anteriormente, se refleja la necesidad de buscar una posible solución con la incorporación de un conjunto de funcionalidades al SIGICEM; para lograr así, una gestión de piezas más detallada y que se ajuste a las nuevas necesidades del cliente.

Por lo anteriormente expuesto, se identifica como **problema a resolver**: ¿Cómo viabilizar el proceso de gestión tecnológica en el Sistema de Gestión para Ingeniería Clínica y Electromedicina que garantice una mayor disponibilidad del equipamiento médico?

Definiéndose como **objeto de estudio**: Proceso de gestión tecnológica; el cual delimita el **campo de acción**: Gestión tecnológica de piezas de equipos médicos para el Sistema de Gestión para Ingeniería Clínica y Electromedicina.

Todo ello encaminado a cumplir el siguiente **objetivo general**: Desarrollar un módulo para la gestión tecnológica que garantice una mayor disponibilidad del equipamiento médico en el Sistema de Gestión para Ingeniería Clínica y Electromedicina.

Para dar cumplimiento al objetivo general se trazaron las siguientes **tareas de la investigación**:

1. Caracterización de los sistemas de gestión tecnológica existentes a nivel internacional y nacional, estableciéndose similitudes con la investigación en curso.

* Equipo médico compuesto por uno o más equipos.

2. Determinación de las nuevas particularidades del proceso de gestión de piezas de repuesto a través de entrevistas realizadas a los especialistas del Centro de Ingeniería Clínica y Electromedicina para la implementación de la solución propuesta.
3. Asimilación de la arquitectura definida por el proyecto Sistema de Gestión para Ingeniería Clínica y Electromedicina para el desarrollo de la solución propuesta, que garantice la gestión tecnológica de piezas de equipo médico.
4. Diseño de los artefactos del flujo de trabajo Modelamiento de Negocio, Requisito, Análisis y Diseño e Implementación sirviendo de base para el desarrollo de la solución propuesta.
5. Implementación de las funcionalidades identificadas en el proceso de gestión tecnológica aplicando las pautas de diseño definidas y siguiendo lo establecido en la especificación de requisitos de software.

Entre los **beneficios** que se esperan con el desarrollo de la investigación se encuentran:

- Garantizará la gestión de la información del equipamiento médico instalado en las unidades de salud del país, cumpliendo al mismo tiempo, con las nuevas características presentes en el centro cliente, debido a su reestructuración.
- Proveerá información de las necesidades reales de adquisición de piezas para cada unidad de salud, garantizando la disponibilidad de estas durante el proceso de atención tecnológica.
- Permitirá asegurar la continuidad del proceso de gestión tecnológica del equipamiento médico, a través del perfeccionamiento y automatización de los procesos relacionados con la tecnología médica instalada en el país.
- Contribuirá a una planificación más completa de mantenimiento para equipos médicos, garantizando un aumento en el tiempo de vida útil de los mismos.
- Impedirá en ocasiones la importación innecesaria de piezas de repuesto a utilizar en la reparación de equipos médicos.

El presente documento está estructurado en cuatro capítulos:

Capítulo 1. Fundamentación Teórica, se caracterizan los sistemas de gestión tecnológica existentes a nivel internacional y nacional, estableciéndose similitudes con la investigación en curso. Se describen

diversos conceptos que permiten una mejor comprensión de la terminología utilizada. Igualmente, se analiza la metodología, tecnologías y herramientas a usar en el desarrollo de la solución propuesta.

Capítulo 2. Características del Sistema, se determinan las nuevas particularidades del proceso de gestión de piezas de repuesto a través de entrevistas realizadas a los especialistas del Centro de Ingeniería Clínica y Electromedicina para la implementación de la solución propuesta. Del mismo modo, se realiza una descripción del modelo de negocio definido, así como de sus elementos más importantes. Igualmente, se especifican los requisitos funcionales y no funcionales, a partir de los cuales se representan los casos de uso del sistema y la descripción de los mismos.

Capítulo 3. Análisis y Diseño del Sistema, se describen los artefactos del flujo de trabajo de análisis y diseño; constituyendo la base para la implementación del módulo de Gestión Tecnológica del Sistema de Gestión para Ingeniería Clínica y Electromedicina v2.0.

Capítulo 4. Implementación, se expone una descripción del desarrollo de la solución inicialmente propuesta mediante la representación del Modelo de Implementación. Además, se presentan los estándares y estilos de codificación a utilizar; así como, algunos de los criterios tomados en cuenta para llevar a cabo el tratamiento de errores en el módulo de Gestión Tecnológica del Sistema de Gestión para Ingeniería Clínica y la Electromedicina v2.0.

CAPÍTULO 1. FUNDAMENTACIÓN TEÓRICA

El presente capítulo tiene como objetivo realizar una breve descripción de los principales conceptos asociados a la investigación propuesta. Se exponen características de sistemas homólogos existentes a nivel internacional y nacional como parte del estudio del arte. Además, se analiza la metodología, tecnologías y herramientas a emplear en el desarrollo de la solución propuesta.

1.1 Términos relacionados con la investigación

En la actualidad las empresas se encuentran ante un entorno muy diferente al existente en períodos anteriores, los que al caracterizarse por elevados niveles de estabilidad y escasos niveles de incertidumbre permitieron una amplia implementación de modelos y técnicas de gestión, entre los que no se encontraba la gestión de la información de la tecnología o de los recursos tecnológicos.

“A partir de la mitad de la década de los ochenta, el factor tecnológico ha pasado a constituir un vector estratégico que permite que la empresa mejore su posición competitiva, pues su ausencia produce una grave insuficiencia para generar innovaciones en productos y procesos. Es necesario, por tanto, gestionar estos recursos tecnológicos con la misma eficiencia que los demás recursos para que la empresa adquiera una mayor capacidad de adaptación y, sobre todo, la posibilidad de anticipar e, incluso, provocar rupturas que le permitan renovar sus ventajas competitivas en el momento oportuno” (3). Por lo que se hace necesario, antes de dar cumplimiento a las tareas trazadas en la investigación y lograr un mejor desarrollo empresarial, conocer qué es la gestión tecnológica.

Para lograr dicha comprensión es importante definir algunos conceptos, entre los cuales se encuentra el de **tecnología**: “conjunto de teorías y de técnicas que permiten el aprovechamiento práctico del conocimiento científico” (4). Por otra parte el término **gestión** hace referencia a la acción y a la consecuencia de administrar o gestionar algo (5). Entendiéndose por gestionar: llevar a cabo diligencias que hacen posible la realización de una operación comercial o de un anhelo cualquiera.

A partir de estos dos conceptos se presenta el de **gestión tecnológica**: disciplina en desarrollo que contiene un conjunto de actividades y estrategias que se implementan en diversas organizaciones empresariales con el objetivo de orientar el manejo y control de la tecnología; todo ello a través de procesos y metodologías, que sirven para perfeccionar la productividad de una organización y así

maximizar los resultados de la misma. Además, trata de implantar una conciencia elevada de la necesidad de desarrollar diversas innovaciones y mejoras periódicas (6). En publicaciones del CINDA (Centro de Desarrollo Interuniversitario) la conceptualizan como “proceso de adopción y ejecución de decisiones sobre las políticas, estrategias, planes y acciones relacionadas con la creación, difusión y uso de la tecnología”. (7)

Hoy en día las empresas se enfrentan a retos muy significativos como la competitividad, velocidad de los cambios, globalización y tecnología, entre otros requisitos empresariales. Retos que para equilibrarlos puede ser un proceso difícil y desalentador. De ahí el surgimiento de los sistemas de gestión, al permitir aprovechar y desarrollar el potencial existente en una organización. Con una implementación eficaz de estos las empresas pueden:

- Mejorar la efectividad operativa
- Reducir costos
- Aumentar la satisfacción de clientes y partes interesadas
- Proteger la marca y la reputación
- Lograr mejoras continuas
- Potenciar la innovación

1.2 Sistemas de gestión tecnológica

Un **sistema de gestión** es una “estructura probada para la gestión y mejora continua de las políticas, los procedimientos y procesos de una organización” (8). Actualmente, dada su importancia y necesidad de implantación en las diversas organizaciones empresariales, en todo el mundo, se han desarrollado sistemas de gestión de disímiles temas; entre los cuales se pueden encontrar de calidad, de información, de contenidos, de paquetes, de seguridad, de recursos humanos y ambientales. Otra de esas nuevas propuestas de administración de recursos que se ha realizado debido a la evolución de los mercados, es la gestión de tecnología.

Los **sistemas de gestión de tecnología** tienen una gran importancia en las empresas, ya que son sistemas de gestión de negocio cuyo propósito es administrar las ventajas competitivas de las mismas. Cuba a pesar de ser un país que posee una gran desventaja en cuanto a desarrollo tecnológico, no ha dejado de hacer extensivo en todas sus empresas la aplicación de estos sistemas, para así lograr la

máxima eficacia, eficiencia y competitividad. En los siguientes enunciados se abordarán algunos de los sistemas de gestión tecnológica existentes a nivel internacional y nacional.

Sistemas de gestión tecnológica existentes a nivel internacional

- Sistema Informático de Gestión de Departamentos de Ingeniería Clínica

Sistema desarrollado en Argentina, orientado a la gestión de todas aquellas tecnologías involucradas en el cuidado de la salud y en el desarrollo de la misma. El mismo, asiste a la gestión de tecnologías médicas, al permitir a sus usuarios realizar las tareas de: gestión de inventario, estudio del ciclo de vida de equipos médicos y generación de planes de mantenimiento. Además, emplea el sistema Microsoft Visual FoxPro, que está dirigido a la construcción de bases de datos y posee un lenguaje nativo fácil de entender, tanto para programadores expertos, como para principiantes. Actualmente, es una aplicación propietaria, utilizada en el Instituto de Maternidad y Ginecología Nuestra Señora de las Mercedes, y en el Hospital Avellaneda que radica en la provincia de Tucumán en Argentina. (9)

- EasyMaint

Software realizado por *aBESofT Technologies* para la gestión, control y planificación del mantenimiento de la tecnología. EasyMaint ofrece un completo control sobre las operaciones que se realizan en una empresa; eliminando el registro redundante de datos, generando reportes personalizados y asegurando la consistencia y veracidad de todo el proceso realizado en la misma. Igualmente, permite programar las órdenes de trabajo de mantenimiento y calcular los costos estimado de las mismas (10). Este a pesar de las características presentes tiene como principal dificultad que es una aplicación de escritorio; además, la forma de gestionar las órdenes de servicio no se ajusta a las nuevas especificaciones del centro cliente.

- CWORKS

Sistema informatizado de gestión de mantenimiento, gestionado en Malasia, Estados Unidos y Australia, por un grupo de profesionales de la tecnología del mantenimiento y de la información. Es un sistema en línea, el cual permite la gestión de órdenes de servicio, inventarios y reportes de equipos médicos. El sistema, permite a los usuarios registrarse gratuitamente, asignándole una capacidad de almacenamiento muy limitada para su trabajo; la cual puede ser aumentada mediante acuerdos de pago. Una de las principales limitantes que posee, es que la empresa proveedora solo brinda el servicio, nunca el sistema

como tal, lo que hace que las empresas no puedan realizarle cambios y así adaptarlo a sus propias necesidades. (11)

Sistemas de gestión tecnológica existentes a nivel nacional

- Sistema Automatizado Cubano para el Control de Equipos Médicos (SACCEM)

Aplicación informática desarrollada “para el control y seguimiento de los equipos médicos instalados en el Sistema Nacional de Salud. La misma está compuesta por doce módulos de aplicación, entre los cuales se encuentra un portal de vigilancia, un módulo para el reporte de eventos adversos por parte de las instituciones de salud y otro por parte del fabricante, así como un generador de encuestas para el monitoreo de los equipos médicos”. (12)

El SACCEM está estructurado por componentes desarrollados con herramientas y tecnologías de software libre que responden a tendencias arquitectónicas de actualidad; pero no cumple con las necesidades del cliente, debido a que solo permite el control para la gestión de la información del equipamiento médico en la etapa de post comercialización.

- Sistema de Mantenimiento Asistido por Computadoras Orientado a Riesgos (SMACOR)

Sistema desarrollado por el Centro de Bioingeniería (CEBIO) del Instituto Superior Politécnico José Antonio Echeverría (ISPJAE). Entre sus principales actividades, se encuentra: planificar el mantenimiento preventivo, llevar el control del inventario de todo el equipamiento médico, emitir órdenes de trabajo preventivas y correctivas; así como, generar documentos de procedimientos de mantenimiento y pruebas de aceptación (13). SMACOR, sistema no soportado para trabajar en redes, que tiene como principal dificultad la falta de soporte técnico por parte de su equipo de desarrollo.

- Software ReporTech

“Permite registrar la información de los equipos médicos, así como su ubicación en los centros de salud de los países introducidos en el sistema. Ofrece la posibilidad de introducir criterios de comparación para cada modalidad de equipo médico y además, controlar la información del personal asignado para instalar y brindar mantenimiento a los equipos de cada centro. Asimismo, brinda la posibilidad de configurar los reportes que se requieran generar”. (14)

Actualmente es muy utilizado en el país, principalmente en el campo de salud, aunque puede ser usado en otras áreas. ReporTech, es una aplicación de escritorio que no está soportada para trabajar en redes, siendo esta una de las desventajas que presenta ante los sistemas de gestión web. Además, se implementó en el lenguaje C Sharp con la herramienta de desarrollo privada Microsoft Visual Studio.

Sistema de Gestión para Ingeniería Clínica y la Electromedicina (SIGICEM) v1.0

Como se había anticipado, la tecnología es entendida como el equipamiento y todos aquellos instrumentos y materiales diseñados por el hombre para facilitar sus tareas; sin embargo, en nuestros días se entiende por Tecnología en Salud (TS) a “los equipos, dispositivos médicos y quirúrgicos usados en la atención médica”. (15)

Con un uso apropiado de la TS es posible una mejora de los servicios de la salud, y de ahí la importancia de llevar a cabo una buena gestión de la tecnología en la salud, pues esta “es un abordaje sistemático y cuantificable para asegurar que la relación costo/efectividad, eficiencia, seguridad y tecnología disponible sea lo apropiado para cubrir con calidad la demanda por el cuidado de los pacientes”. (15)

Dada la necesidad de gestionar la tecnología existente en los hospitales de Cuba; se desarrolló un sistema informático que cubre todo el proceso de gestión llevado a cabo por los especialistas del CICEM, garantizando la fiabilidad y actualización de la información referente al equipamiento médico instalado en el país. El actual SIGICEM cuenta con varios módulos, entre los cuales se destaca el de gestión tecnológica, encargado de gestionar todo el proceso referente a las piezas de los equipos médicos.

Dicho módulo, actualmente no cuenta con funcionalidades que permitan: aprovechar la información de los productos de tipo pieza que se gestionan en el módulo Gestión de Almacén, proporcionar un seguimiento adecuado a las solicitudes de piezas que se realizan a través del módulo Orden de Servicio y permitir realizar el despiece a los sistemas o equipos médicos. Además, no satisface las necesidades actuales del centro cliente.

Teniendo en cuenta que el módulo anterior no posee las funcionalidades que se requieren y debido a que los sistemas de gestión tecnológica analizados son dependientes de plataformas predeterminadas, son software propietarios y poseen características propias de la institución para la cual fueron diseñados por lo que no cumplen con las nuevas necesidades del negocio; surge la necesidad de implementar nuevos

requerimientos e incluso modificar otros ya implementados en el SIGICEM. No obstante, en el desarrollo de la solución propuesta, se utilizará como guía para la implementación de algunas de las funcionalidades del módulo, el software Reportech.

1.3 Metodología, tecnologías y herramientas

Uno de los procesos fundamentales en la construcción de un software es la selección de las herramientas y tecnologías a utilizar. Para el diseño e implementación del módulo de Gestión Tecnológica v2.0 serán utilizadas las establecidas por el proyecto Sistema de Gestión para Ingeniería Clínica y Electromedicina y el Centro de Informática Médica.

Lenguaje Unificado de Modelado 2.1

Lenguaje de modelado de sistemas de software utilizado por la metodología RUP, del inglés *Rational Unified Process*, en la creación de diseños que capturen las ideas de los desarrolladores de forma convencional y de fácil comprensión, para un mejor entendimiento y comunicación con otras personas. A través de este lenguaje gráfico se podrá visualizar, especificar, construir y documentar el módulo propuesto en esta investigación, permitiendo además la descripción de aspectos concretos relacionados con el lenguaje de programación utilizado, los esquemas de la base de datos, así como los componentes reutilizables. (16)

Herramienta CASE

Ingeniería de Software Asistida por Ordenador, es “una combinación de herramientas de software y metodologías de desarrollo” (17); las cuales ayudan a los ingenieros de software y analistas durante una o más fases del desarrollo de software. Aceleran el desarrollo de los sistemas y aumentan la calidad del software.

- Visual Paradigm for UML 6.4

Para el modelado del módulo de Gestión Tecnológica v2.0 se utilizará como herramienta CASE el Visual Paradigm, pues constituye una herramienta profesional que soporta el ciclo completo del desarrollo del software y emplea como lenguaje de modelado UML. Tiene la capacidad de ejecutarse sobre diferentes sistemas operativos, lo que le otorga la característica de ser multiplataforma. La utilización de este software, permitirá realizar una rápida construcción de la solución planteada y brindará la posibilidad de

representar los diagramas de clases, así como parte de la documentación de esta. Asimismo, proporcionará abundantes tutoriales de UML que igualmente ofrecerán un conjunto de ventajas. (18)

Entorno de Desarrollo Integrado

El IDE es un programa informático que contiene un amplio conjunto de herramientas de programación que utilizan los programadores para la implementación del producto deseado. Consiste en un editor de código, un compilador, un depurador y constructor de interfaz gráfica. Es un entorno de programación que suministra un marco de trabajo amigable.

- NetBeans 6.9

Para el desarrollo del módulo se utilizará como herramienta IDE, NetBeans en su versión 6.9, debido a su enorme versatilidad; pues permite realizar programas en diferentes lenguajes de programación como PHP y Java Scripts. Además, cuenta con soporte para el *framework* Symfony y ofrece un entorno completamente gráfico que permite desarrollar aplicaciones web. Contiene una enorme cantidad de extensiones y plugins, que pueden incorporarse al sistema para facilitar las tareas de programación. También, se destaca su intuitiva interfaz de usuario, su sistema de depuración de programas y las demás herramientas de ayuda, que facilitarán el desarrollo del módulo de Gestión Tecnológica v2.0.

Plataforma de Desarrollo: LAMP

Como el módulo de Gestión Tecnológica v2.0 se integrará al Sistema de Gestión para Ingeniería Clínica y Electromedicina, se decidió utilizar la plataforma de desarrollo LAMP. La misma está compuesta por un conjunto de aplicaciones *open source* que trabajando entre sí, permiten disponer de un servidor web, ya sea para desarrollo o para producción.

La plataforma, está integrada por:

- Linux (L) con su distribución Ubuntu 10.04 como sistema operativo.
- Apache (A) como servidor web en su versión 2.2.
- MySQL (M) en su versión 5.1, como sistema gestor de base datos.
- PHP (P) en su versión 5.1 como lenguaje de programación.

Frameworks

Un *framework* facilita el desarrollo de una aplicación mediante la automatización de determinados patrones utilizados para resolver las tareas comunes. Proporciona una estructura al código fuente, forzando a los desarrolladores a crear códigos más legibles y más fáciles de mantener. Además, facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas. (19)

- **Symfony 1.4**

“Symfony es un completo *framework* diseñado para optimizar, gracias a sus características, el desarrollo de las aplicaciones web. Para empezar, separa la lógica de negocio, la lógica de servidor y la presentación, de la aplicación web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.” (19)

Con el objetivo de simplificar el desarrollo de la solución propuesta en la investigación, se empleará el Framework de desarrollo Symfony en su versión 1.4; pues además de las ventajas antes mencionadas, es compatible con un gran número de gestores de bases de datos y puede ser ejecutado tanto en plataformas Linux como en Windows. Gracias a su facilidad de extensión, en el SIGICEM, se cuenta con novedosos componentes que facilitan el tedioso trabajo de crear complejos formularios con el framework ExtJS.

- **ExtJS 2.2**

ExtJS es un conjunto de librerías Java Script del lado del cliente utilizada para el desarrollo de aplicaciones web interactivas, la cual usa tecnologías AJAX, DHTML y DOM. Framework de presentación completamente Orientado a Objetos y multiplataforma; el cual brinda a los desarrolladores múltiples posibilidades para el trabajo con las validaciones y manejo de errores en el cliente. Además, permite realizar completas interfaces de usuario fáciles de usar y muy parecidas a las conocidas aplicaciones de escritorio. (20)

En el desarrollo del capítulo, se analizaron algunos de los sistemas de gestión tecnológica existentes a nivel internacional y nacional; llegándose a la conclusión, que los sistemas encontrados no constituyen una solución factible para aplicarlos en el CICEM, debido a que la mayoría presentan características

propias de la institución para la cual fueron diseñados o son aplicaciones de escritorio que no están soportadas para trabajar en redes.

De ahí, la necesidad de desarrollar una solución que garantice la gestión tecnológica de piezas de equipos médicos en todo el país. Para lo cual serán aplicadas las pautas establecidas por el proyecto SIGICEM y el Centro de Informática Médica, donde se emplea la metodología de desarrollo RUP, apoyado en el lenguaje de modelado UML 2.1. Las herramientas: NetBeans 6.9 como IDE y Visual Paradigm 6.4 como herramienta CASE. Igualmente, se utilizan los frameworks de desarrollo Symfony 1.4 y ExtJS 2.2, soportados sobre una plataforma LAMP.

CAPÍTULO 2. CARACTERÍSTICAS DEL SISTEMA

A partir de entrevistas realizadas a los especialistas del Centro de Ingeniería Clínica y Electromedicina, se determinaron las nuevas particularidades del proceso de gestión tecnológica de piezas de repuesto a tener en cuenta en la implementación de la solución propuesta. Describiéndose, en el presente capítulo, el modelo de negocio definido y sus elementos más importantes. Igualmente, se especifican los requisitos funcionales y no funcionales identificados, mediante los cuales se representan los casos de uso del sistema y la descripción de los mismos.

2.1 Gestión tecnológica de piezas de repuesto para la tecnología médica

El Centro de Ingeniería Clínica y Electromedicina realiza, entre otras actividades, la gestión de las piezas de repuesto de la tecnología médica existente en el país; para ello, cuenta con un sistema informático compuesto por un módulo encargado de realizar dicha tarea. Actualmente, el módulo no cumple con todos los requisitos necesarios para llevar a cabo el proceso, por lo que se hace necesaria la modificación e inserción de nuevas funcionalidades al mismo.

Para el desarrollo de la solución propuesta en la investigación, se realizaron entrevistas a los especialistas del CICEM con el objetivo de conocer todo el proceso referente a la gestión tecnológica llevada a cabo en el centro. Se identificaron los procedimientos de gestión de piezas de equipos médicos y se detallaron las actividades necesarias para ejecutar eficientemente estos procedimientos.

2.2 Modelo de Negocio

El modelado de negocio es la base para comprender mejor la operación de una organización. El objetivo principal, es mostrar una descripción bien detallada de los procesos identificados en el negocio mediante el Modelo de Casos de Usos del Negocio; para así, ayudar al equipo desarrollador en la comprensión del mismo y establecer futuras funcionalidades del sistema.

A partir de las investigaciones realizadas, se identificaron los tres procesos fundamentales que se realizan en el Departamento de Gestión Tecnológica de la entidad:

Despiece: tiene como objetivo llevar un registro de las piezas que componen cada tecnología médica y las que pueden ser usadas a la hora de efectuarse la reparación de las mismas; identificándose el caso de uso del negocio: Realizar despiece.

Registro y satisfacción: tiene como objetivo llevar un registro de los reportes de piezas de repuesto y de la satisfacción de los mismos; identificándose el caso de uso del negocio: Satisfacer solicitud de material.

Gestión de adquisición: tiene como objetivo gestionar la compra de piezas de repuesto; identificándose el caso de uso del negocio: Satisfacer solicitud de compra de piezas.

Los casos de uso del negocio identificados, representan una secuencia de acciones realizadas en el negocio, que producen un resultado de valor observable para ciertos actores del negocio; los cuales representan a cualquier individuo, grupo, entidad, organización, máquina o sistema de información externos, con los que el negocio interactúa. Igualmente, en el modelado de negocio se describen los trabajadores del mismo, utilizándose como punto de partida para derivar un primer conjunto de actores y casos de usos para el sistema en construcción. (21)

Los actores y trabajadores que se describen a continuación son los relacionados con el negocio del módulo de Gestión Tecnológica.

Tabla 1. Actores del negocio

| Actores del negocio | Descripción |
|---------------------|---|
| Operador | Es la persona que se encargada, una vez obtenido un nuevo equipo médico, reportar la instalación del mismo. |
| Unidad de Salud | Es el encargado de reportar la rotura de un equipo médico determinado. |

Tabla 2. Trabajadores del negocio

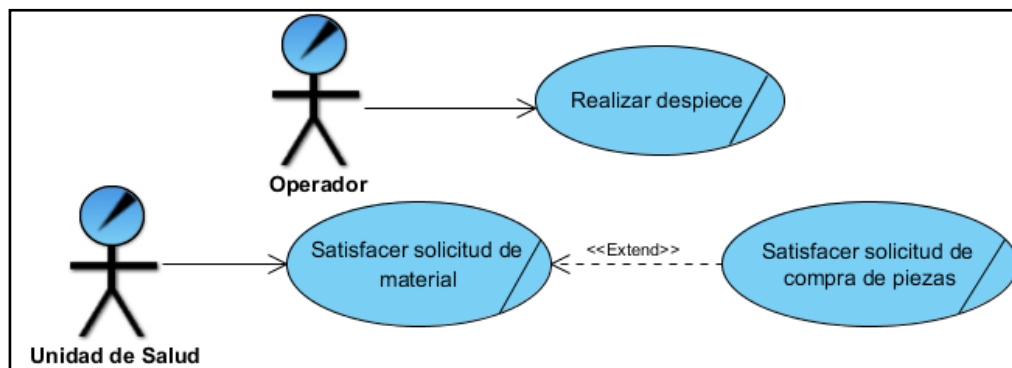
| Trabajadores del negocio | Descripción |
|--------------------------|---|
| Especialista de Despiece | Es la persona encargada de registrar un equipo o sistema; así como, el despiece del mismo, luego de recibir un reporte de instalación y realizar la Defectación. |
| Especialista Técnico | Es la persona encargada de crear una orden de servicio a partir de un reporte y enviarla al jefe de brigada; para luego a partir de esta obtener una solicitud de material. |

| | |
|------------------------|---|
| Jefe de Brigada | Es el responsable de confeccionar una solicitud de material. Además, se encarga de buscar la existencia de pieza en otros almacenes y solicitar un movimiento de pieza de almacén en caso de encontrarla. Por otro lado, si la pieza necesaria no puede ser obtenida mediante otros almacenes, solicita su adquisición a través de un reporte de pieza. |
| Almacenero | Es el responsable de consultar la solicitud de material y completarla en caso de que las piezas solicitadas se encuentren disponibles en el almacén o informar al jefe de brigada la inexistencia de la misma. Igualmente, se encarga de consultar las solicitudes de movimiento de pieza de almacén y aceptarlas de ser posible. |
| Especialista General | Es la persona encargada de confeccionar una solicitud de pieza a través de un reporte. |
| Especialista Principal | Es el encargado de confeccionar una solicitud de compra a través de las solicitudes de piezas y enviarla al importador. Además, analiza el listado de ofertas y en caso de aceptar alguna, la asigna a una solicitud de compra. |
| Importador | Es el encargado buscar las ofertas más favorables y darlas a conocer al especialista general; así como, actualizar la información referente al proceso de compra. |

Diagrama de casos de uso del negocio

Para tener una visión general de los diferentes procesos del negocio, se construyó el diagrama de casos de usos del negocio, en el cual aparece cada proceso del negocio como un caso de uso. Además, se muestra la relación existente entre actores y casos de usos del negocio.

Figura 1. Diagrama de casos de uso del negocio

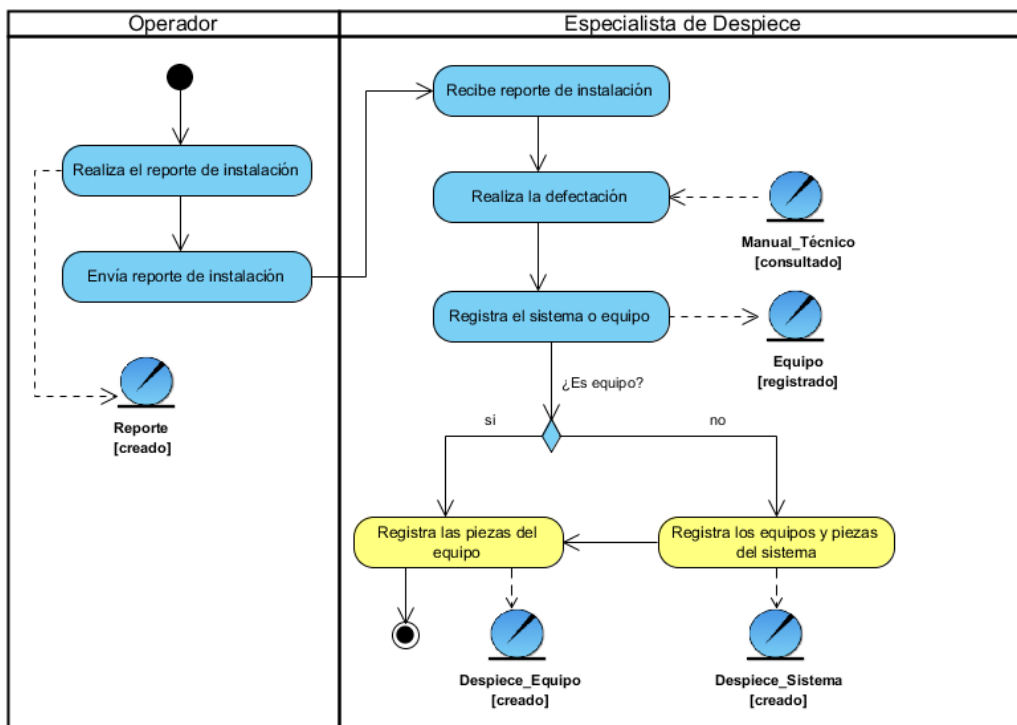


Diagramas de Actividades

- Caso de uso del negocio: Realizar despiece

El negocio comienza, cuando el operador efectúa un reporte de instalación de un nuevo equipo médico. Una vez realizado el reporte, el especialista de despiece, registra las piezas que contiene el equipo y en caso de ser un sistema, registra además los equipos que lo componen y sus respectivas piezas; terminando de esta forma el negocio.

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

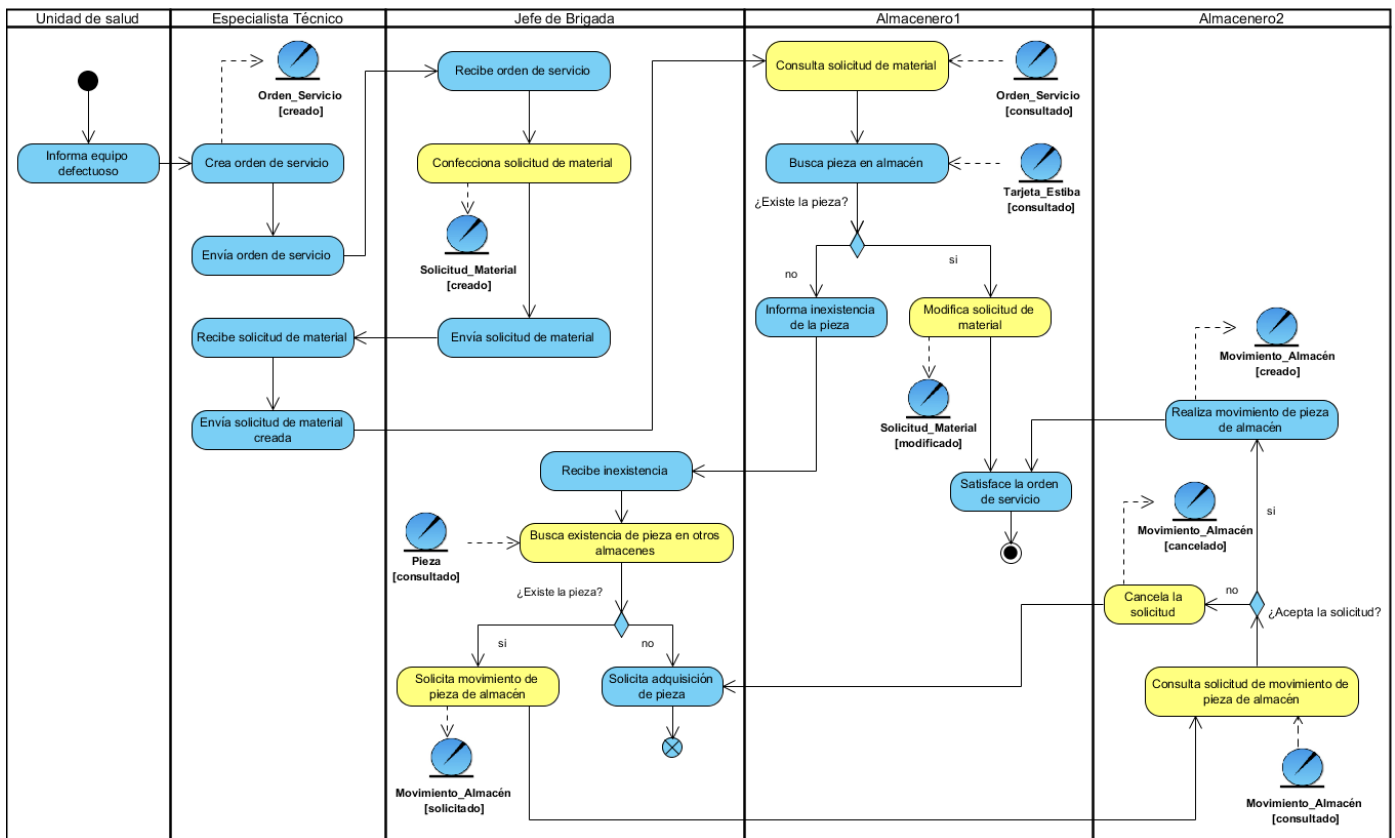


- Caso de uso del negocio: Satisfacer solicitud de material

El negocio comienza, cuando la unidad de salud informa a un especialista técnico de la rotura de un equipo médico determinado; encargándose este último, de crear una orden de servicio en la cual refleja las piezas necesarias a utilizar en la reparación. El jefe de brigada es el responsable de confeccionar a partir de la orden de servicio una solicitud de material, la cual es enviada al almacenero encargado de satisfacer dicha solicitud. Si las piezas se encuentran disponibles en el almacén se despachan las piezas y se concluye el negocio.

El almacenero, en caso de no disponer de alguna de las piezas solicitadas informa la inexistencia de la misma al jefe de brigada; el cual trata de obtenerla mediante otros almacenes y de no ser posible, solicita la adquisición de la misma, concluyendo de manera parcial el negocio. Si la pieza se encuentra disponible en otro almacén, el jefe de brigada, envía una solicitud de movimiento al almacenero correspondiente; el cual, de ser posible remite la pieza al almacén que la solicitó para satisfacer la orden de servicio y concluir nuevamente el negocio.

Figura 3. Diagrama de actividades del caso de uso del negocio: Satisfacer solicitud de material

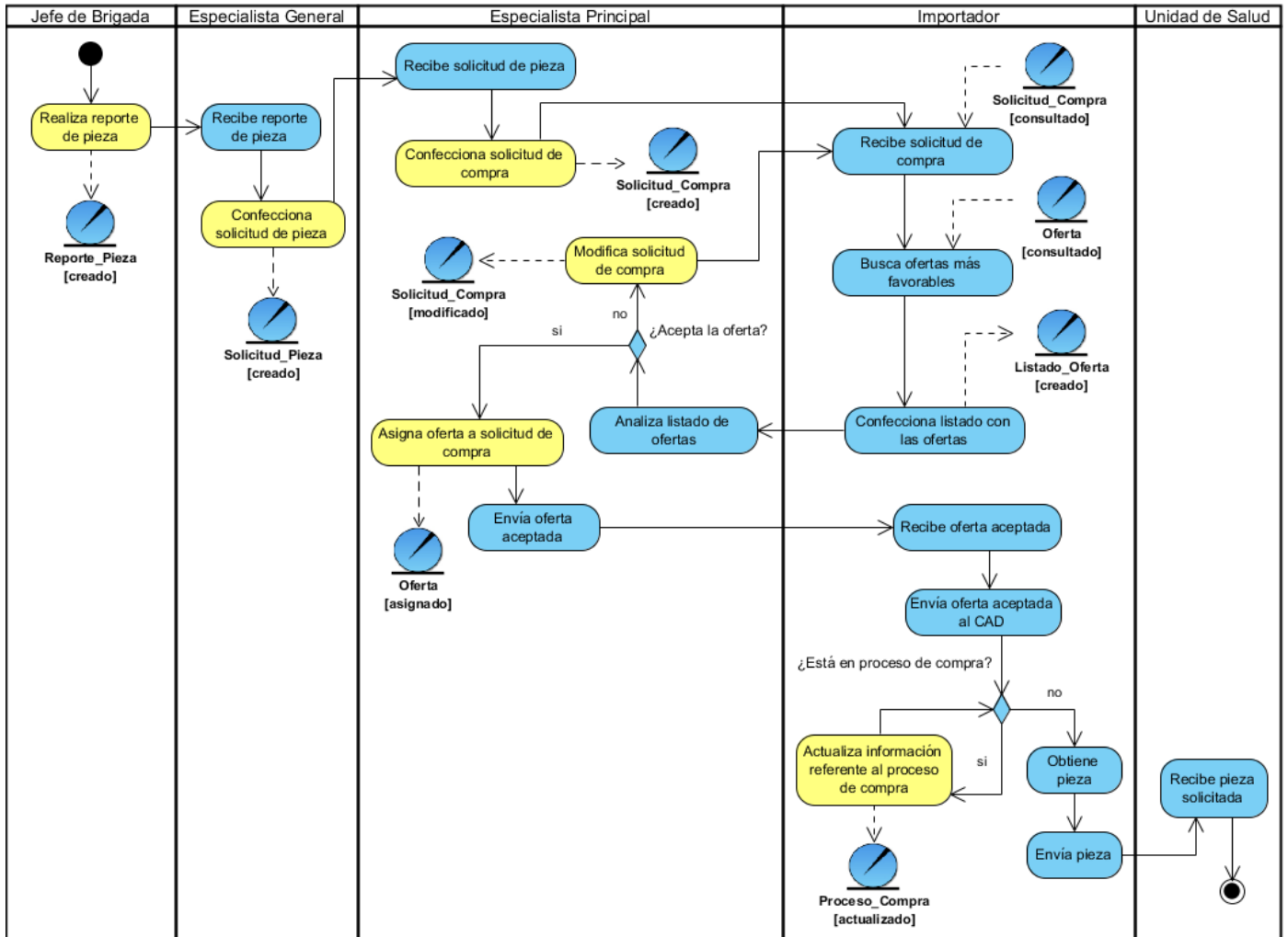


- Caso de uso del negocio: Satisfacer solicitud de compra de piezas

El negocio comienza, cuando el jefe de brigada realiza un reporte de pieza donde le da a conocer al especialista general la inexistencia de una o varias piezas necesarias en la satisfacción de una orden de servicio. El especialista general, es el encargado de realizar la solicitud de pieza; a partir de la cual el especialista principal confecciona la solicitud de compra y la envía al importador, encargándose este de

buscar las ofertas más favorables y darlas a conocer al especialista principal; el cual selecciona la más conveniente, efectuándose posteriormente el proceso de compra. Una vez obtenidas las piezas, estas son enviadas a las unidades de salud para las cuales fueron solicitadas; terminando de esta forma el negocio.

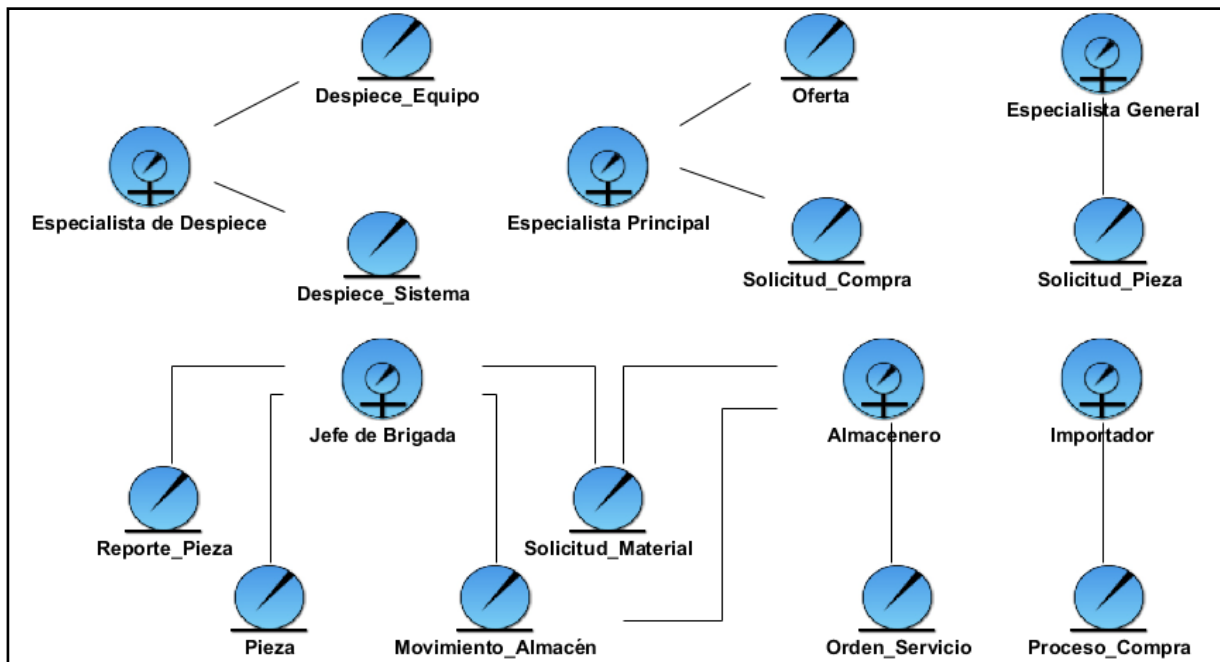
Figura 4. Diagrama de actividades del caso de uso del negocio: Satisfacer solicitud de compra de piezas



Modelo de objetos

El modelo de objetos del negocio es un modelo interno a un negocio. Describe como cada caso de uso del negocio, es llevado a cabo por parte de un grupo de trabajadores que utilizan un conjunto de entidades del negocio y unidades de trabajo (21).

Figura 5. Modelo de objetos



Reglas del negocio

Una regla de negocio es una declaración que define o limita algún aspecto del negocio. Su intención es valorar la estructura del negocio o controlar o influenciar el comportamiento del mismo (22). Estas deben ser declarativas, expresadas en lenguaje natural, construidas de maneras independientes y orientadas al negocio. Definiéndose para la construcción de la solución propuesta las siguientes reglas del negocio:

- Debe ser consultado el manual técnico de un equipo antes de realizar el despiece.
- Cada orden de servicio debe estar asociada a un reporte.
- Las solicitudes de materiales deben ser obtenidas a través de órdenes de servicio.
- Si no existe una solicitud de material no se puede despachar la pieza.
- Una pieza solicitada no puede ser comprada sin la aprobación del CAD.
- La solicitud de material debe ser confeccionada mediante el modelo establecido en el centro.
- Si una solicitud de material se modifica se debe conservar la original.

2.3 Propuesta de solución

A partir del análisis realizado, se propone el desarrollo de un módulo para el Sistema de Gestión para Ingeniería Clínica y Electromedicina, que permita gestionar todo el proceso referente al aseguramiento de piezas necesarias en la atención a la tecnología médica; por tanto este debe permitir:

- Realizar despiece a nivel de equipo o sistema.
- Gestionar solicitudes de materiales.
- Gestionar solicitudes de movimiento de almacén.
- Gestionar solicitudes de compra.
- Gestionar ofertas.
- Generar informes de pedidos de piezas para órdenes de servicio.
- Gestionar los estados de las solicitudes de compra.
- Gestionar los tipos de monedas.
- Gestionar proveedores.

2.4 Especificación de requisitos de software

En la especificación de requisitos de software se realiza una descripción completa del comportamiento del módulo a desarrollar. Los requisitos, según la metodología RUP, se dividen en dos: en funcionales, aquellos requisitos que el usuario necesita que el software efectúe y no funcionales, las propiedades o cualidades que el producto de software debe tener.

Durante el levantamiento de requisitos fueron identificados 57 requisitos funcionales, los cuales se describen a continuación:

Tabla 3. Descripción de requisitos funcionales

| Requisitos Funcionales | Descripción |
|-----------------------------------|--|
| 1. Realizar despiece desde equipo | Permite realizar el despiece a un equipo a partir de los siguientes datos: nomencladores de equipo, cantidad equipo, cantidad de pieza y nomencladores de pieza. |
| 2. Realizar despiece desde pieza | Permite realizar el despiece a partir de una pieza, introduciendo los siguientes datos: nomencladores de equipo y cantidad de equipo. |

| | |
|---|--|
| 3. Adicionar pieza | Permite adicionar una pieza a través de los siguientes datos: nomenclador de pieza, descripción, referencia, precio y gastable. |
| 4. Modificar pieza | Permite modificar una pieza a través de los siguientes datos: nomenclador de pieza, descripción, referencia, precio y gastable. |
| 5. Visualizar pieza | Permite visualizar una pieza con los siguientes datos: nomenclador de pieza, descripción, referencia, precio y gastable. |
| 6. Eliminar pieza | Permite eliminar una pieza. |
| 7. Buscar pieza | Permite buscar una pieza. |
| 8. Visualizar órdenes de servicios pendientes de solicitud de material. | Permite visualizar las órdenes de servicios pendientes de solicitud de material con los siguientes datos: unidad de salud, denominación, marca, modelo y número de serie del equipo. |
| 9. Crear solicitud de material para orden de servicio | Permite crear una solicitud de material a través de los siguientes datos: autorizado por, código de unidad y centro de costo. |
| 10. Visualizar estado de gestión de pieza | Permite visualizar la descripción de las piezas con sus respectivos estados de gestión. |
| 11. Crear solicitud de movimiento de almacén | Permite crear una solicitud de movimiento de almacén a través de los siguientes datos: pieza, cantidad, almacén y código. |
| 12. Buscar existencia de piezas en otros almacenes | Permite mostrar si la pieza se encuentra disponible en el almacén. |
| 13. Crear reporte de pieza para orden de servicio | Permite crear un reporte a partir de la información de una pieza. |
| 14. Visualizar piezas de orden de servicio | Permite visualizar la descripción de las piezas de una orden de servicio determinada. |
| 15. Crear solicitud de material | Permite crear una solicitud de material a partir de los siguientes datos: especialista, almacén, orden de servicio, código y empresa. |

| | |
|---|--|
| 16. Eliminar solicitud de material | Permite eliminar una solicitud de material. |
| 17. Atender solicitud de material | Permite satisfacer una solicitud de material a partir de los siguientes datos: recibido por, número de lote, número de solicitud y vale de entrega. |
| 18. Visualizar solicitudes de material pendientes | Permite visualizar la siguiente información de la solicitud de material pendiente: unidad de salud, especialista y fecha. |
| 19. Visualizar solicitudes de material atendidas | Permite visualizar la siguiente información de la solicitud de material atendida: unidad de salud, especialista y fecha. |
| 20. Generar PDF de solicitud de material | Permite generar un PDF con toda la información referente a la solicitud de material. |
| 21. Atender solicitudes de movimiento de pieza de almacén | Permite aceptar una solicitud de movimiento de pieza de almacén, recogiendo la siguiente información: recibido por, autorizado por, número de lote, número de solicitud y vale de entrega. |
| 22. Rechazar solicitudes de movimiento de pieza de almacén | Permite rechazar una solicitud de movimiento, actualizando el estado de las piezas que se solicitaron. |
| 23. Visualizar solicitudes de movimiento de pieza de almacén atendidas | Permite visualizar los siguientes datos: almacén especialista y fecha. |
| 24. Visualizar solicitudes de movimiento de pieza de almacén pendientes | Permite visualizar los siguientes datos: almacén especialista y fecha. |
| 25. Generar PDF de solicitud de movimiento de almacén | Permite generar un PDF con toda la información referente a la solicitud de movimiento. |
| 26. Visualizar piezas solicitadas para equipos rotos | Permite visualizar la siguiente información: marca del equipo, modelo del equipo, número de serie del equipo, código de la pieza y cantidad de piezas. |
| 27. Generar PDF de piezas solicitadas para equipos rotos | Permite generar un PDF con toda la información referente a las piezas solicitadas para equipos rotos. |
| 28. Crear solicitud de pieza para orden de servicio | Permite crear una solicitud de pieza a partir de los siguientes datos: descripción y cantidad de piezas. |
| 29. Visualizar pieza de orden de servicio para solicitud | Permite visualizar los siguientes datos de la pieza: código, unidad de salud, pieza y cantidad. |
| 30. Crear solicitud de pieza para prevención | Permite crear una solicitud de pieza a partir de la selección |

| | |
|--|--|
| | de la descripción de una pieza y su respectiva cantidad. |
| 31. Visualizar piezas de solicitudes | Permite visualizar la descripción de las piezas y la cantidad de pieza de cada tipo solicitada. |
| 32. Buscar piezas de solicitudes | Permite buscar una pieza a partir de su descripción. |
| 33. Crear solicitud de compra | Permite crear una solicitud de compra a partir de los siguientes datos: pieza, cantidad, especialista, consumidor y fecha. |
| 34. Visualizar solicitud de compra | Permite visualizar los siguientes datos de la solicitud: pieza, cantidad, especialista, consumidor y fecha. |
| 35. Modificar solicitud de compra | Permite modificar los siguientes datos de la solicitud: pieza, cantidad, especialista, consumidor y fecha. |
| 36. Buscar solicitud de compra | Permite buscar una solicitud de compra a partir de los siguientes datos: pieza, cantidad, especialista, consumidor y fecha. |
| 37. Generar PDF de modelo de solicitud de compra | Permite generar un PDF con todos los datos referentes a la solicitud de compra. |
| 38. Adicionar Oferta | Permite adicionar una oferta a partir de los siguientes datos: pieza, número de oferta, solicitud de compra, proveedor y tipo de moneda. |
| 39. Visualizar Oferta | Permite visualizar una oferta con los siguientes datos: pieza, número de oferta, solicitud de compra, proveedor y tipo de moneda. |
| 40. Buscar Oferta | Permite buscar una oferta a partir de los siguientes datos: pieza, número de oferta, solicitud de compra, proveedor y tipo de moneda. |
| 41. Adicionar proveedor | Permite adicionar un proveedor a partir de los siguientes datos: nombre y país. |
| 42. Modificar proveedor | Permite modificar un proveedor a partir de los siguientes datos: nombre y país. |
| 43. Visualizar proveedor | Permite visualizar un proveedor con los siguientes datos: nombre y país. |
| 44. Eliminar proveedor | Permite eliminar un proveedor a partir de los siguientes |

| | |
|---|--|
| | datos: nombre y país. |
| 45. Buscar proveedor | Permite buscar un proveedor a partir de los siguientes datos: nombre y país. |
| 46. Adicionar tipo de moneda | Permite adicionar un tipo de moneda a partir de los siguientes datos: moneda, abreviatura y rate de cambio. |
| 47. Modificar tipo de moneda | Permite modificar un tipo de moneda a partir de los siguientes datos: moneda, abreviatura y rate de cambio. |
| 48. Visualizar tipo de moneda | Permite visualizar un tipo de moneda con los siguientes datos: moneda, abreviatura y rate de cambio. |
| 49. Eliminar tipo de moneda | Permite eliminar un tipo de moneda a partir de los siguientes datos: moneda, abreviatura y rate de cambio. |
| 50. Buscar tipo de moneda | Permite buscar un tipo de moneda a partir de los siguientes datos: moneda, abreviatura y rate de cambio. |
| 51. Adicionar estado de solicitud de compra | Permite adicionar un estado de solicitud de compra. |
| 52. Modificar estado de solicitud de compra | Permite modificar un estado de solicitud de compra. |
| 53. Visualizar estado de solicitud de compra | Permite visualizar los estados de solicitud de compra. |
| 54. Eliminar estado de solicitud de compra | Permite eliminar un estado de solicitud de compra. |
| 55. Buscar estado de solicitud de compra | Permite buscar un estado de solicitud de compra. |
| 56. Actualizar información referente al proceso de compra | Permite actualizar la información referente al proceso de compra a partir de los siguientes datos: número de contrato, estado de solicitud y fecha de recepción. |
| 57. Visualizar solicitudes con oferta asignada | Permite visualizar los siguientes datos de la solicitud: solicitud, oferta y estado. |

Requisitos no funcionales (RnF)

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 web 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.

Confiabilidad

- RnF 2.1. El sistema podrá estar inactivo durante cinco minutos, al cabo de ese tiempo se cierra la cesión del usuario que interactúa con este.
- RnF 2.2. Al ocurrir un error el sistema muestra un mensaje indicando dónde es que se ha producido y así pueda ser corregido por el usuario.

Soporte

- RnF 3.1. La implementación del sistema se regirá por las normas de codificación definidas en el documento de arquitectura de software del Centro de Informática Médica para obtener un producto legible y homogéneo.

Restricciones de diseño

- RnF 4.1. El sistema estará desarrollado en base a las políticas del software libre que fueron ajustadas al SNS como: uso de servidores GNU/Linux.
- RnF 4.2. Se utilizará como gestor de base de datos MySQL y Symfony como framework de desarrollo de aplicaciones web, el cual utiliza PHP como lenguaje de programación del lado del servidor. Además, como IDE de desarrollo se utilizará NetBeans.
- RnF 4.3. Para el diseño de las interfaces se utilizará Ext JS, librería java script para el desarrollo de aplicaciones RIA (*Rich Internet Application*).
- RnF 4.4. 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 web contará con una ayuda donde el usuario podrá suplir las dudas que se le puedan presentar durante la utilización de esta. El usuario del módulo deberá recibir una capacitación previa a la utilización del sistema, con el propósito de explotar las prestaciones del sistema sin contratiempos, ocasionados por la falta de preparación técnica.

Interfaz

- RnF 6.1. La interfaz de la aplicación será similar a la interfaz del sistema operativo Windows XP para garantizar que el personal que trabaje con el sistema esté lo más familiarizado posible, de manera que agilice y facilite el trabajo con el software.

- RnF 6.2. La entrada de datos incorrecta será detectada claramente e informada al usuario.
- RnF 6.3. Todos los textos y mensajes en pantalla aparecerán tanto en idioma español como en inglés, en dependencia de la opción que seleccione el usuario.

Estándares aplicables

- RnF 7.1. El desarrollo de la aplicación estará basado en la Integración de Modelos de Madurez de Capacidades (CMMI, por sus siglas en inglés), un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software.

2.5 Modelo de casos de usos del sistema

“El modelo de casos de uso ayuda al cliente, a los usuarios y a los desarrolladores a llegar a un acuerdo sobre cómo utilizar el sistema” (21). Este está conformado por actores y casos de usos del sistema, así como, diagramas de casos de usos. Los actores del sistema no siempre representan a personas; pueden ser actores, otros sistemas o hardware externo que interactuará con el sistema en cuestión.

Tabla 4. Descripción de actores del sistema

| Actores del sistema | Descripción |
|--------------------------|--|
| Especialista de Despiece | Responsable de realizarle el despiece a un sistema o equipo determinado. |
| Jefe de Brigada | Responsable de confeccionar la solicitud de material y satisfacerla a través de una solicitud de movimiento de almacén o de un reporte de pieza. |
| Almacenero | Responsable de satisfacer la solicitud de material a nivel de almacén y de no poder informar la inexistencia de pieza. |
| Especialista General | Responsable de confeccionar la solicitud de pieza. |
| Especialista Principal | Responsable de confeccionar la solicitud de compra y aceptar la oferta más favorable en cada uno de los casos. |
| Importador | Responsable de confeccionar el listado de ofertas, así como, enviar al CAD* aquellas escogidas por el especialista principal. |

* Comisión de Aprobación de Divisas

Descripción de los Casos de Uso del Sistema (CUS)

“Un caso de uso especifica una secuencia de acciones, incluyendo variantes, que el sistema puede llevar a cabo, y que producen un resultado observable de valor para un actor concreto”. (21) Seguidamente, se muestra una descripción de los principales casos de uso del sistema.

CUS Realizar despiece desde equipo

Tabla 5. CUS Realizar despiece desde equipo

| | |
|--------------------|--|
| Objetivo | Permite realizar el despiece a un equipo o sistema determinado. |
| Actor | Especialista de Despiece |
| Resumen | El caso de uso inicia cuando el actor accede a la opción realizar despiece desde la interfaz equipo. Luego, el sistema muestra la interfaz correspondiente, permitiéndole al actor asignar las piezas que pueden ser usadas en el equipo; terminando así el caso de uso. |
| Complejidad | Alta |
| Prioridad | Media |
| Referencia | RF 1 |

CUS Visualizar órdenes de servicios pendientes de solicitud de material

Tabla 6. CUS Visualizar órdenes de servicios pendientes de solicitud de material

| | |
|--------------------|--|
| Objetivo | Visualizar todas las órdenes de servicio mediante las cuales se han solicitado piezas y aún no se le ha confeccionado una solicitud de material. |
| Actor | Jefe de Brigada |
| Resumen | El caso de uso inicia cuando el actor accede a la opción Confeccionar solicitud de material. Luego, el sistema muestra la interfaz correspondiente, permitiéndole al actor visualizar las órdenes de servicio sin solicitud de material creada; terminando así el caso de uso. |
| Complejidad | Baja |
| Prioridad | Alta |
| Referencia | RF 8 |

CUS Crear solicitud de material para orden de servicio

Tabla 7. CUS Crear solicitud de material para orden de servicio

| | |
|-----------------|--|
| Objetivo | Crear una solicitud de material con el autorizo del jefe de Brigada, con el fin de obtener |
|-----------------|--|

| | |
|--------------------|---|
| | las piezas solicitadas por la orden de servicio. |
| Actor | Jefe de Brigada |
| Resumen | El caso de uso inicia cuando el actor selecciona la opción Confeccionar solicitud de material y dentro, la opción Confeccionar. Luego, el sistema muestra la interfaz correspondiente, permitiéndole al actor entrar los datos según correspondan y guardar; terminando así el caso de uso. |
| Complejidad | Media |
| Prioridad | Alta |
| Referencia | RF 9 |

CUS Visualizar estado de gestión de piezas

Tabla 8. CUS Visualizar estado de gestión de piezas

| | |
|--------------------|---|
| Objetivo | Visualizar el estado de gestión en las que se encuentran las piezas solicitadas por las órdenes de servicio. |
| Actor | Jefe de Brigada |
| Resumen | El caso de uso inicia cuando el actor selecciona la opción Estado gestión de piezas. Luego, el sistema muestra la interfaz correspondiente, permitiéndole al actor visualizar cada una de las piezas y su estado actual; terminando así el caso de uso. |
| Complejidad | Media |
| Prioridad | Media |
| Referencia | RF 10 |

CUS Visualizar piezas solicitadas para equipos rotos

Tabla 9. CUS Visualizar piezas solicitadas para equipos rotos

| | |
|--------------------|---|
| Objetivo | Permite visualizar un historial de todas las piezas que han sido solicitadas por órdenes de servicio y su existencia en almacén en el momento que fueron solicitadas. |
| Actor | Jefe de Brigada |
| Resumen | El caso de uso inicia cuando el actor accede a la opción Piezas solicitadas para Equipos rotos. Luego, el sistema muestra la interfaz correspondiente, permitiéndole al actor visualizar el historial de las piezas solicitadas por órdenes de servicio; terminando así el caso de uso. |
| Complejidad | Media |
| Prioridad | Alta |

| | |
|------------|-------|
| Referencia | RF 26 |
|------------|-------|

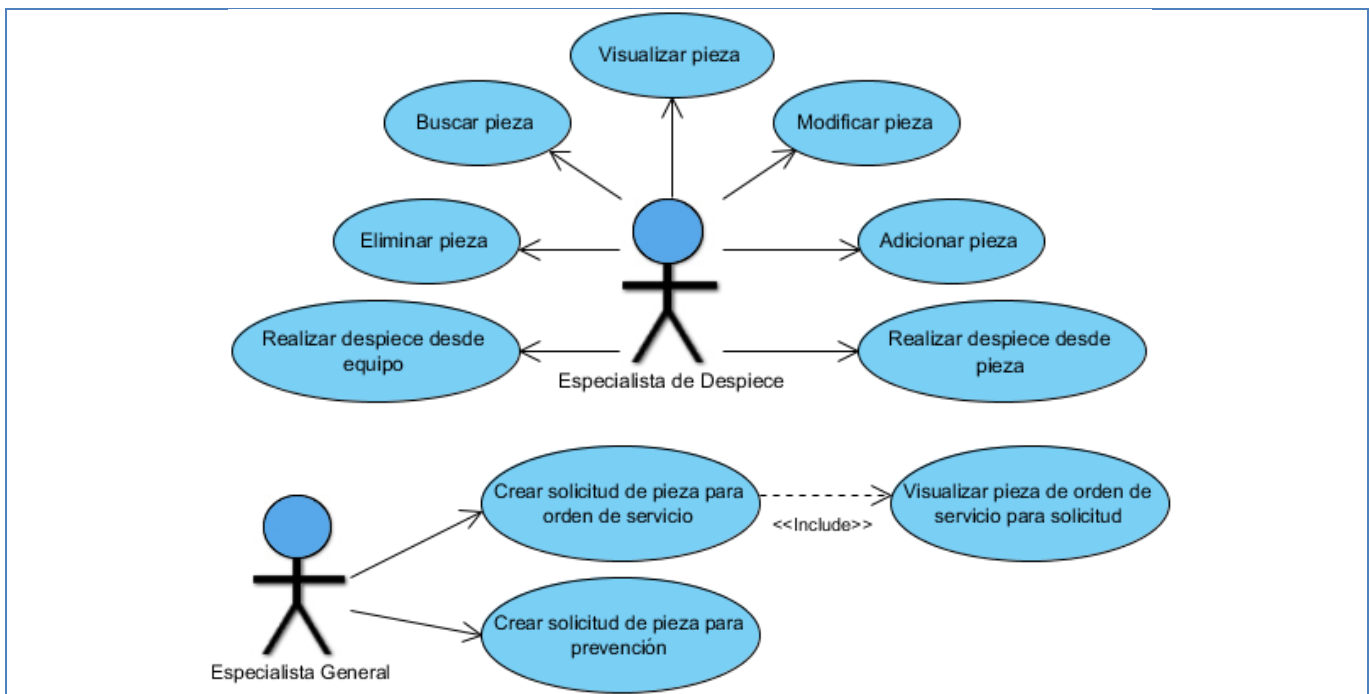
CUS Crear solicitud de compra

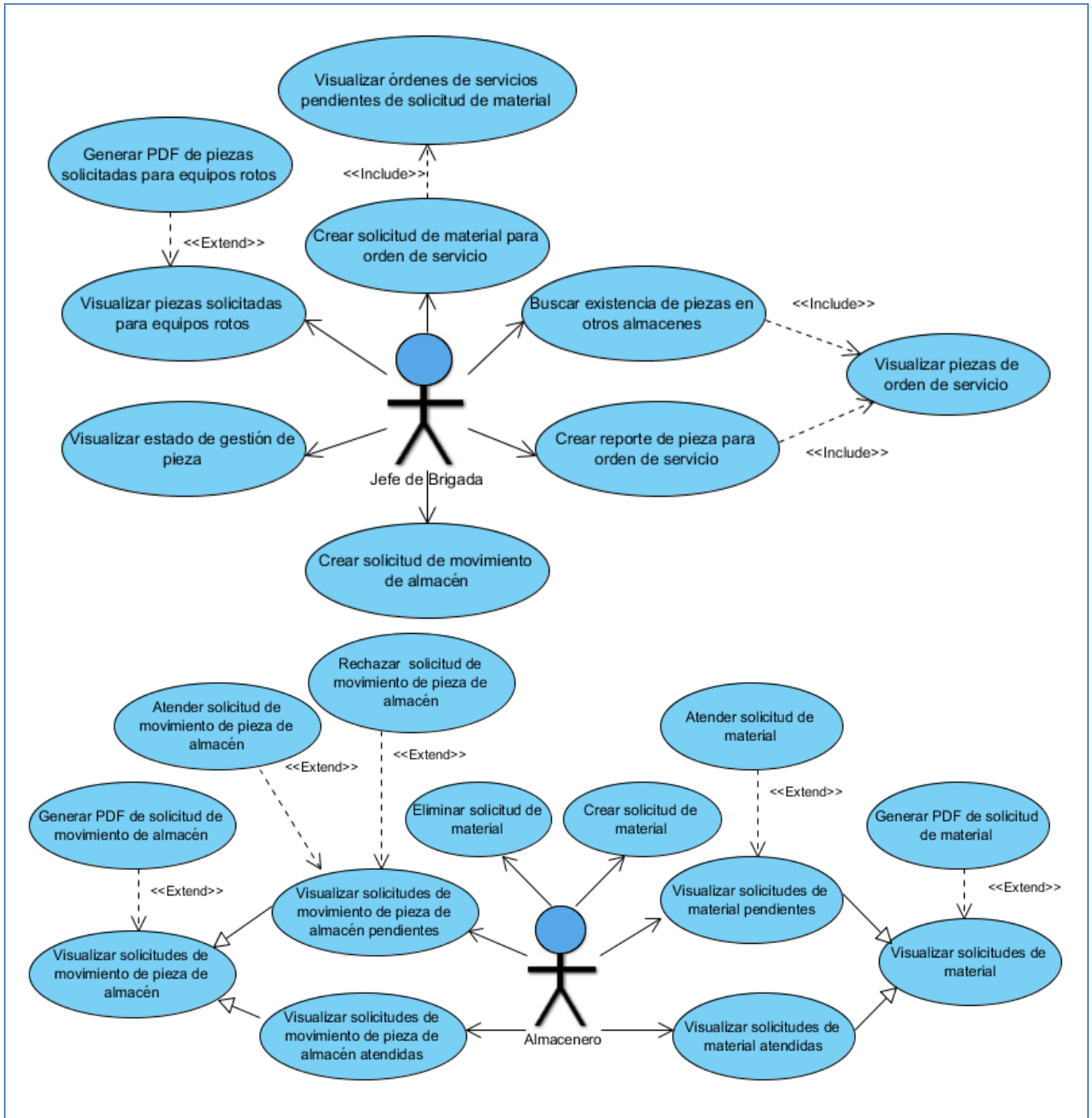
Tabla 10. CUS Crear solicitud de compra

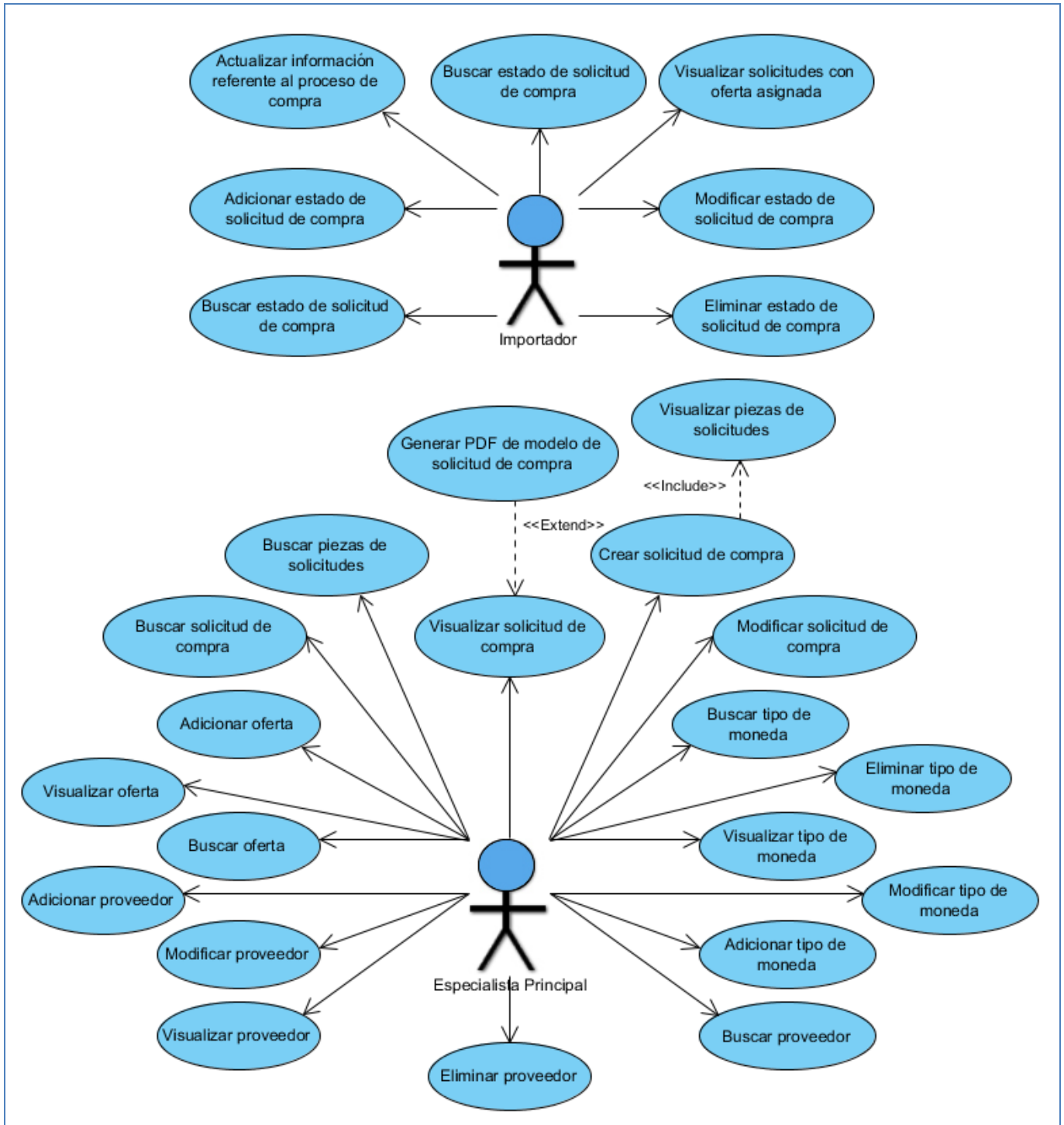
| | |
|-------------|--|
| Objetivo | Permite crear una solicitud de compra a partir de las piezas que se han solicitado, ya sea por plan u órdenes de servicio. |
| Actor | Especialista Principal |
| Resumen | El caso de uso inicia cuando el actor accede a la opción Confeccionar solicitud de compra. Luego, el sistema muestra la interfaz correspondiente, permitiéndole al actor seleccionar las piezas mediante las cuales se va a confeccionar la solicitud de compra e inserta los datos necesarios a registrar en el sistema; terminando así el caso de uso. |
| Complejidad | Media |
| Prioridad | Alta |
| Referencia | RF 33 |

Diagrama de casos de uso del sistema

Figura 6. Diagrama de casos de uso del sistema







En el desarrollo del capítulo, se describieron las principales características a considerar para el desarrollo de la solución propuesta, las cuales se evidencian en las descripciones de los casos de usos del negocio. Igualmente, se representó el flujo del negocio a través del diagrama de actividades, y a partir de una entrevista no estructurada se identificaron los requisitos funcionales y no funcionales del módulo a desarrollar. Finalmente, se describieron los actores del sistema, así como los casos de usos más significativos para el desarrollo de la aplicación.

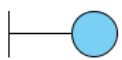
CAPITULO 3. ANÁLISIS Y DISEÑO DEL SISTEMA

El análisis y diseño, tiene como propósito transformar los requerimientos a diseños del sistema; desarrollar una arquitectura robusta para el sistema; así como, adaptar el diseño para hacerlo corresponder con el ambiente de implementación y ajustarla para un desempeño esperado. El presente capítulo describe los artefactos del flujo de trabajo de análisis y diseño; constituyendo la base para la implementación del módulo de Gestión Tecnológica v2.0.

3.1 Modelo de Análisis

En la construcción del modelo de análisis, se analizan los requisitos identificados durante la captura de requisitos, refinándolos y estructurándolos. Este se describe utilizando el lenguaje de los desarrolladores, y puede por tanto introducir un mayor formalismo y ser utilizado para razonar sobre los funcionamientos internos del sistema. Además, permite estructurar los requisitos de un modo que facilite su comprensión, su preparación, su modificación y en general su mantenimiento.

Dentro del modelo de análisis, los casos de usos se representan mediante clases de análisis; las cuales, son identificadas a través de tres estereotipos básicos que proporciona UML, estos son: (21)



Interfaz: se utilizan para modelar la interacción entre el sistema y sus actores. Esta interacción a menudo implica recibir información y peticiones de los usuarios y sistemas externos.



Entidad: se utilizan para modelar información que posee una vida larga y que es a menudo persistente.



Control: representan coordinación, secuencia, transacciones y control de otros objetos; se emplean con frecuencia para encapsular el control de un caso de uso en concreto.

A continuación se muestran, los diagramas de clases del análisis y los diagramas de comunicación, pertenecientes a los principales casos de uso del sistema.

Diagramas de clases del análisis

Figura 7. Diagrama de clase del análisis del CU: Realizar despiece desde equipo

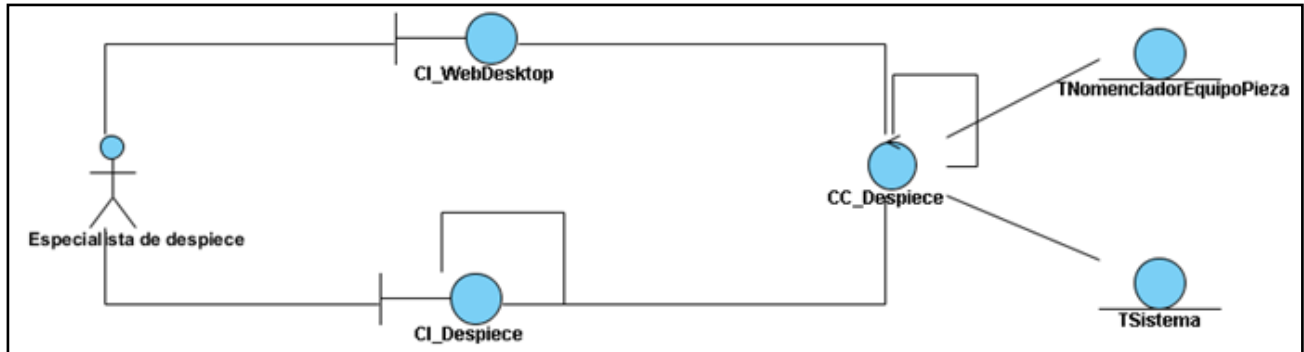


Figura 8. Diagrama de clase del análisis del CU: Visualizar órdenes de servicios pendientes de solicitud de material

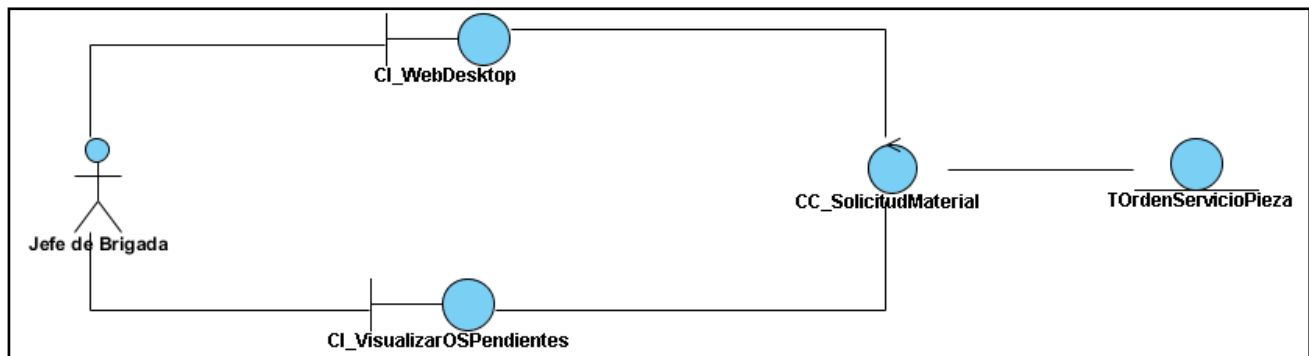


Figura 9. Diagrama de clase del análisis del CU: Crear solicitud de material para orden de servicio

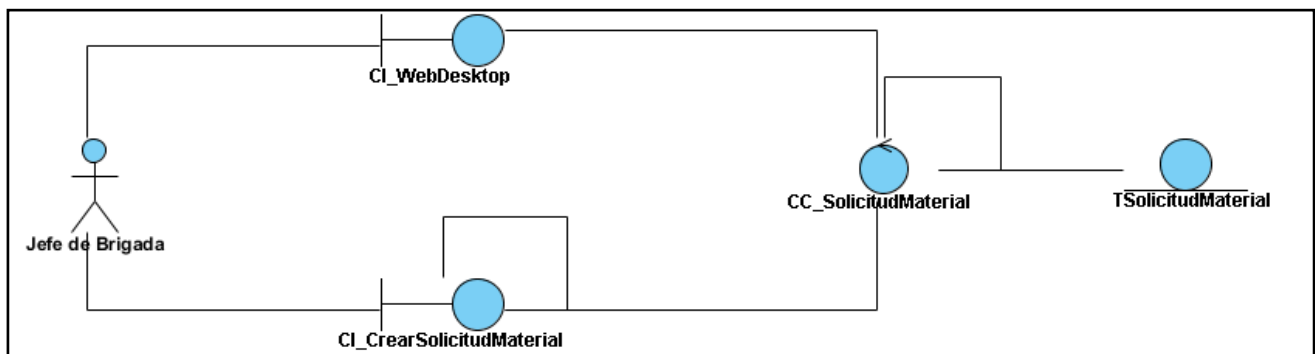


Figura 10. Diagrama de clase del análisis del CU: Visualizar estado de gestión de pieza

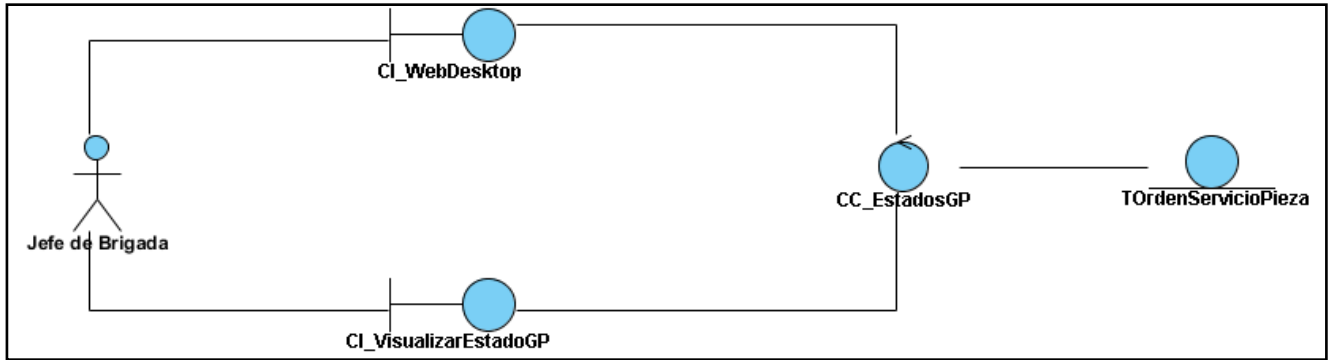


Figura 11. Diagrama de clase del análisis del CU: Visualizar piezas solicitadas para equipos rotos

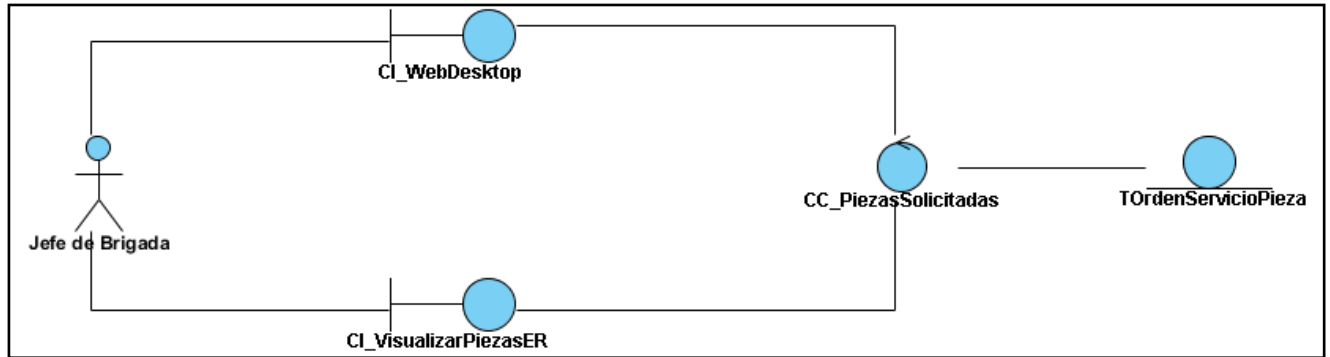


Figura 12. Diagrama de clase del análisis del CU: Crear solicitud de compra

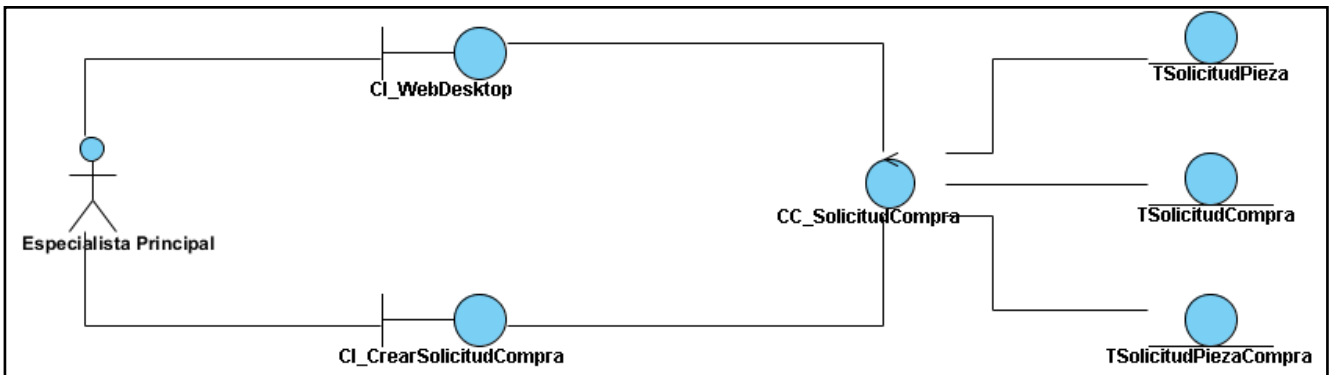


Diagrama de interacción. Comunicación

“Los diagramas de comunicación, resaltan la organización estructural de los objetos o roles que envían y reciben mensajes en la interacción. Son usados para modelar el comportamiento dinámico de un caso de uso”. (23)

Figura 13. Diagrama de comunicación del CU: Realizar despiece desde equipo

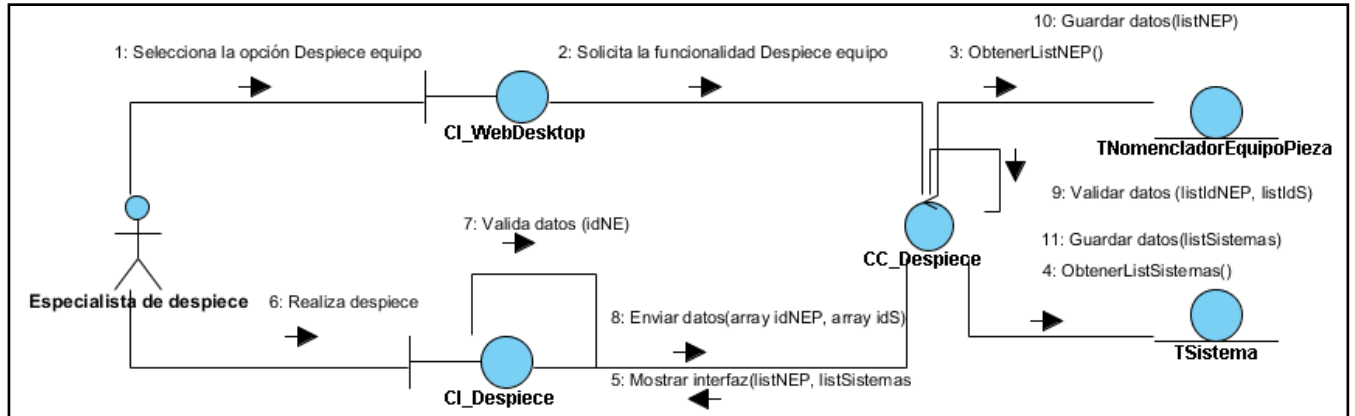


Figura 14. Diagrama de comunicación del CU: Visualizar órdenes de servicios pendientes de solicitud de material

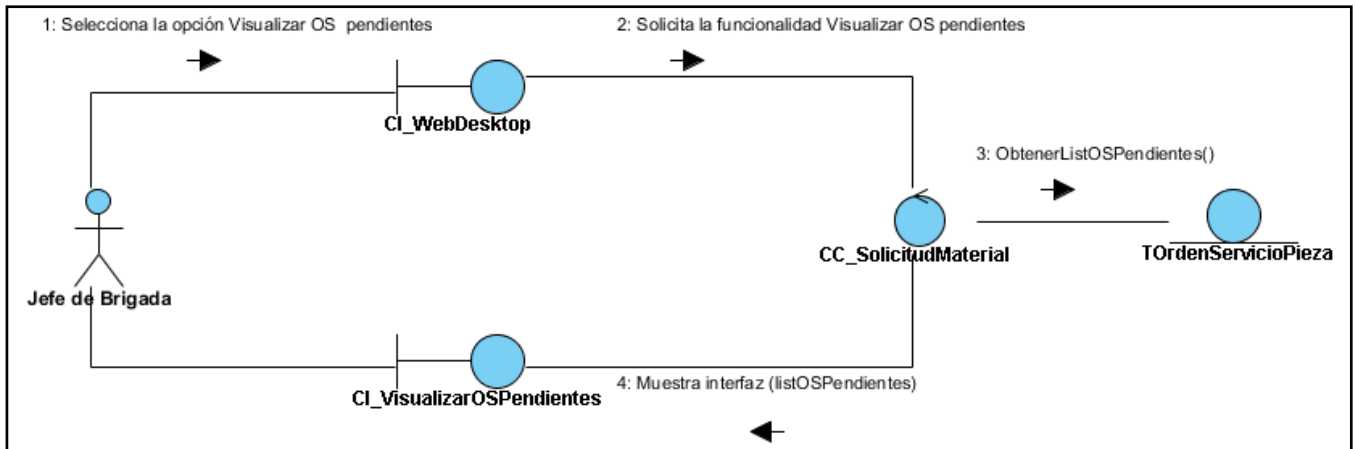


Figura 15. Diagrama de comunicación del CU: Crear solicitud de material para orden de servicio

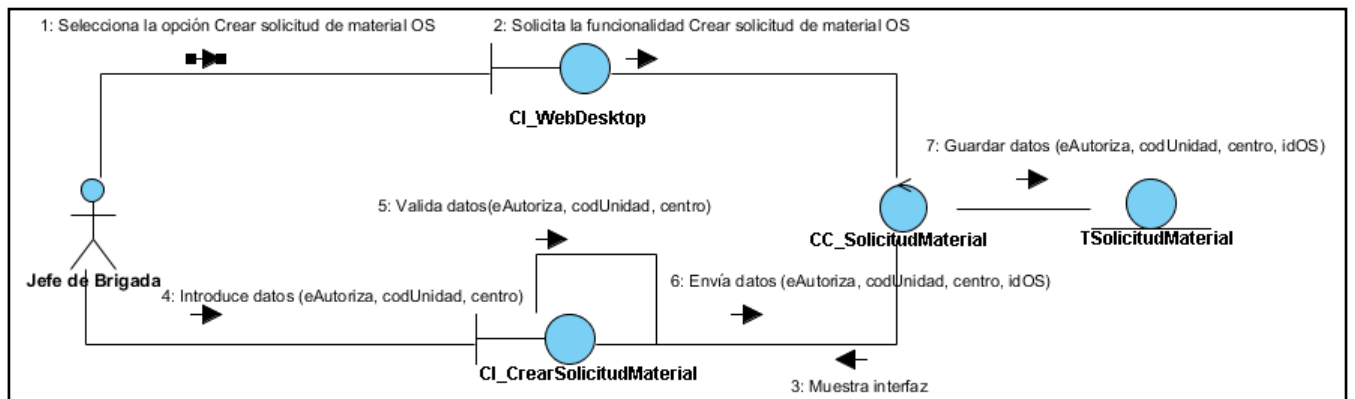


Figura 16. Diagrama de comunicación del CU: Visualizar estado de gestión de pieza

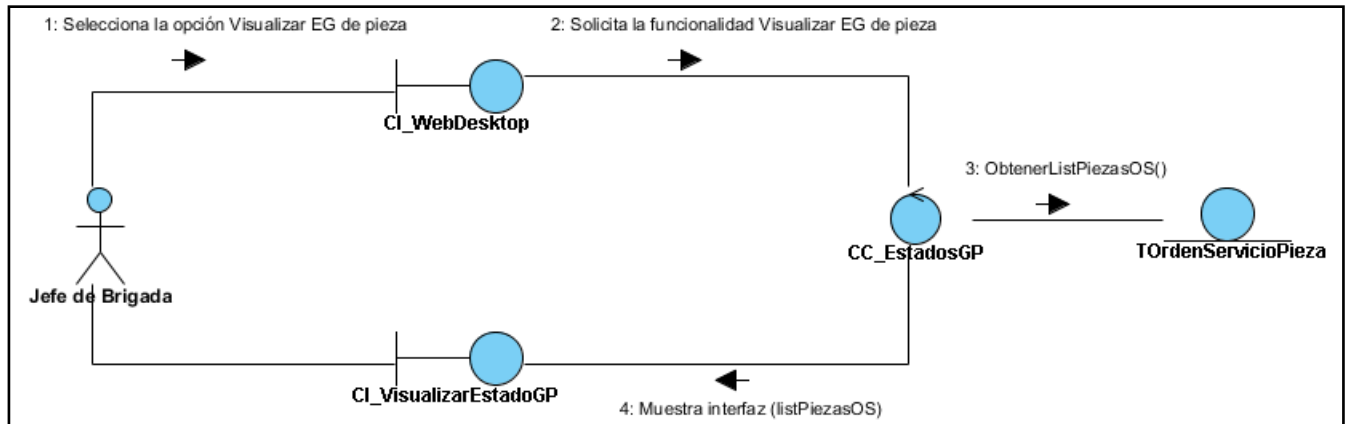


Figura 17. Diagrama de comunicación del CU: Visualizar piezas solicitadas para equipos rotos

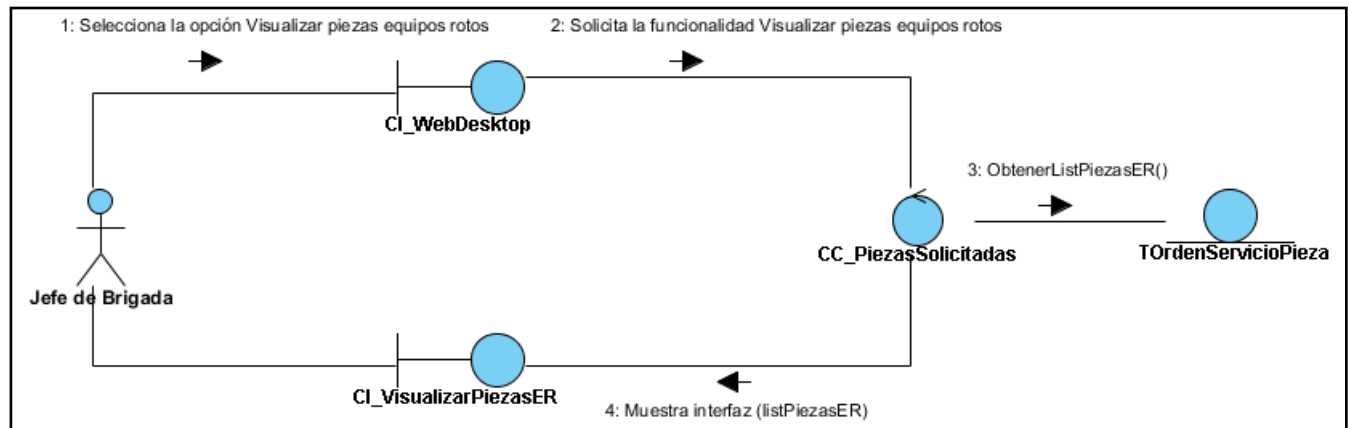
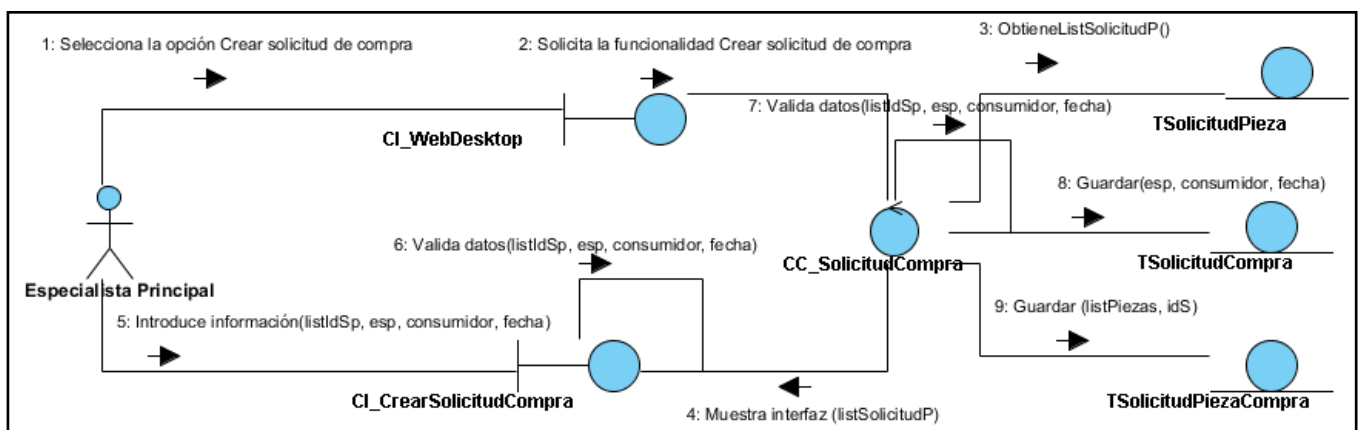


Figura 18. Diagrama de comunicación del CU: Crear solicitud de compra



3.2 Modelo de Diseño

El diseño es el primer paso de la fase de desarrollo de cualquier producto o sistema de ingeniería; es el “proceso de aplicar distintas técnicas y principios con el propósito de definir un dispositivo, proceso o sistema con los suficientes detalles como para permitir su realización física”. (24)

El modelo de diseño, permite traducir con precisión los requisitos del cliente en una representación del software, que se acerca mucho al código fuente. Es la base de todas las partes posteriores del desarrollo y de la fase de prueba. Tiene una gran importancia, debido a que sin él, se corre el riesgo de construir un sistema inestable, difícil de probar y cuya calidad no pueda ser correctamente evaluada.

Diagramas de clases del diseño

Figura 19. Diagrama de clases del diseño del CU: Realizar despiece desde equipo

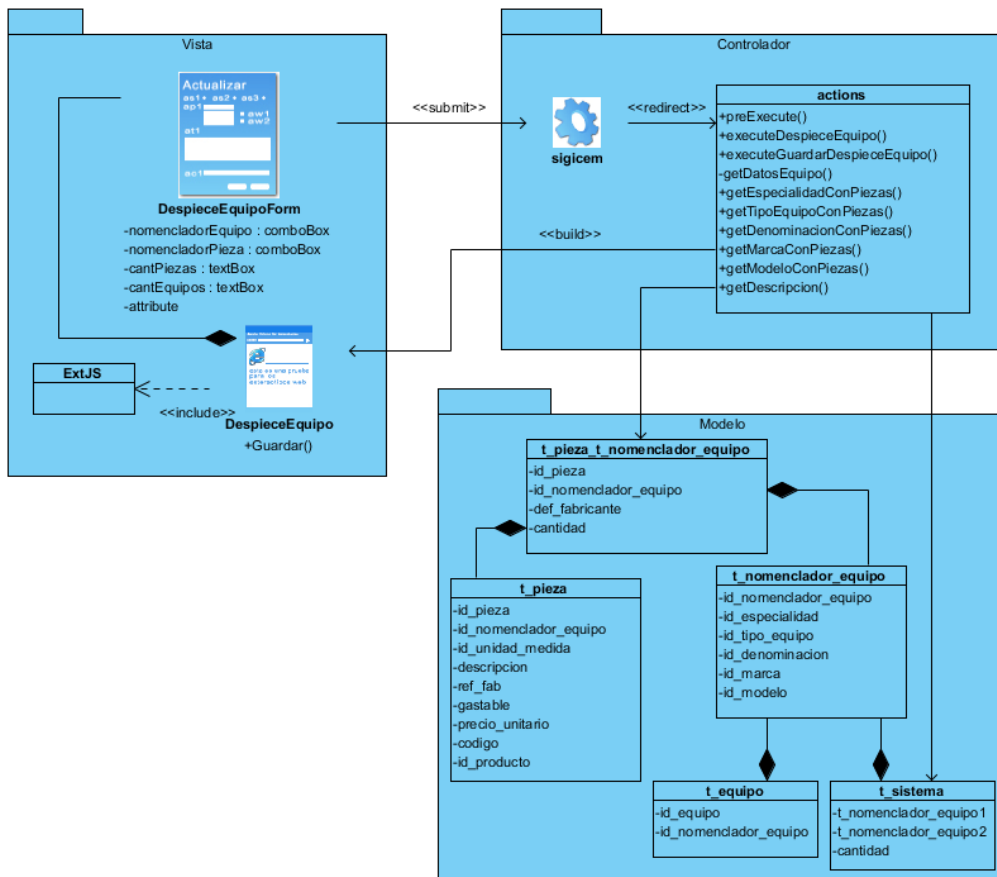


Figura 20. Diagrama de clases del diseño del CU: Visualizar órdenes de servicios pendientes de solicitud de material

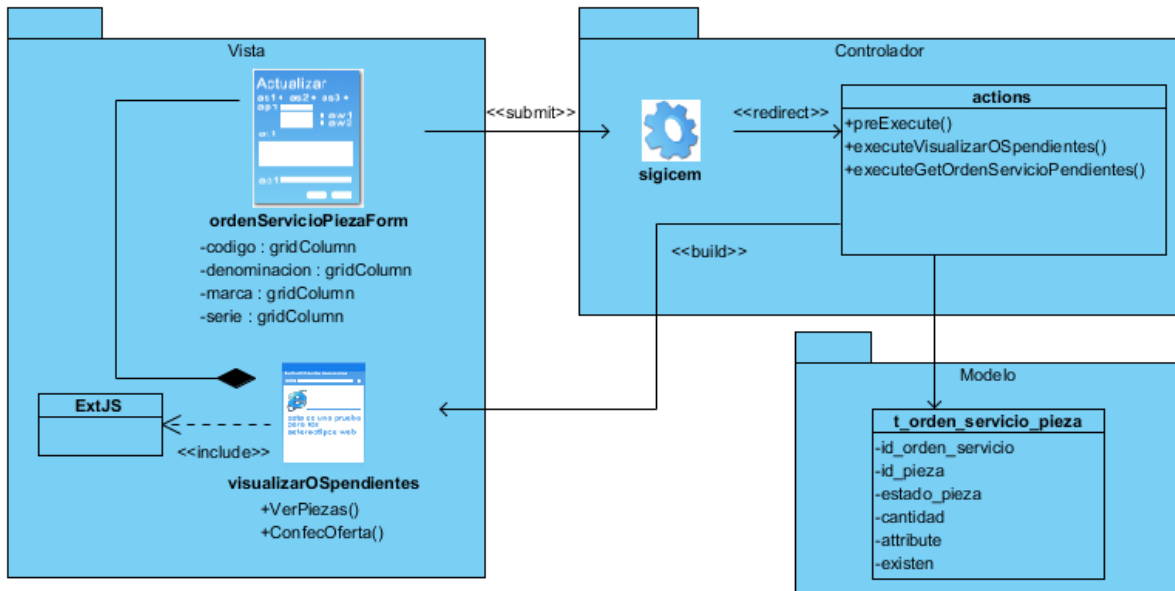


Figura 21. Diagrama de clases del diseño del CU: Crear solicitud de material para orden de servicio

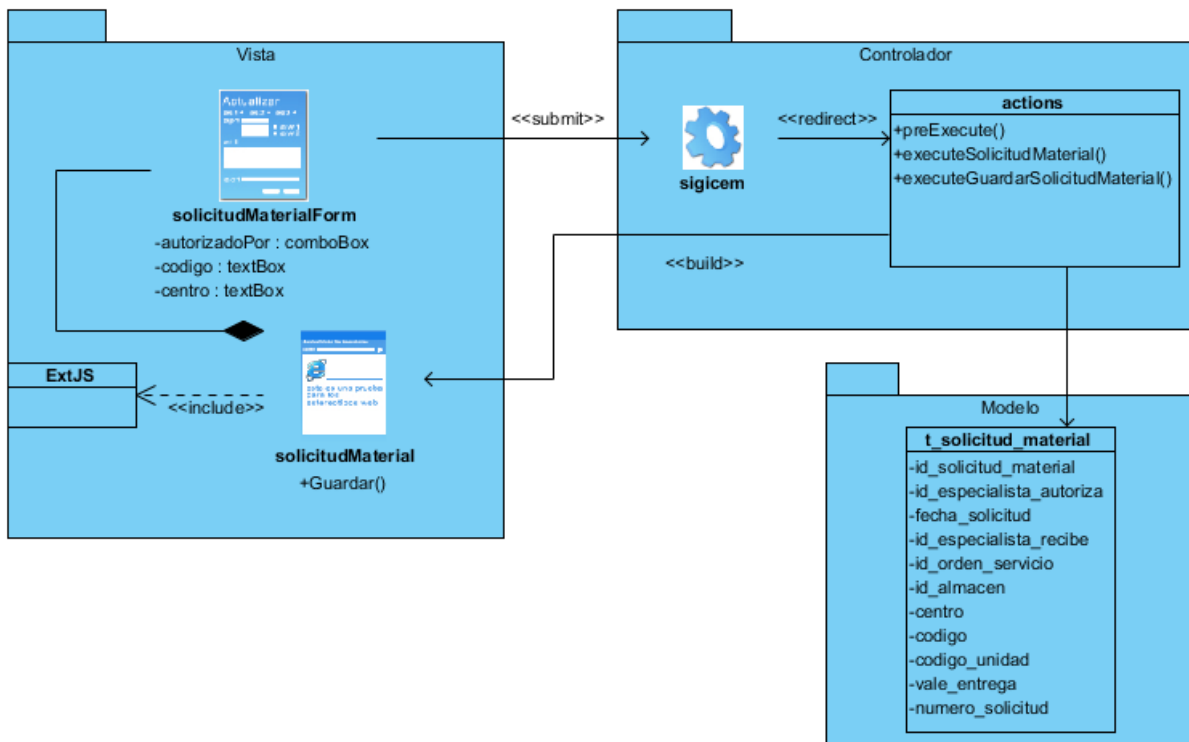


Figura 22. Diagrama de clases del diseño del CU: Visualizar estado de gestión de pieza

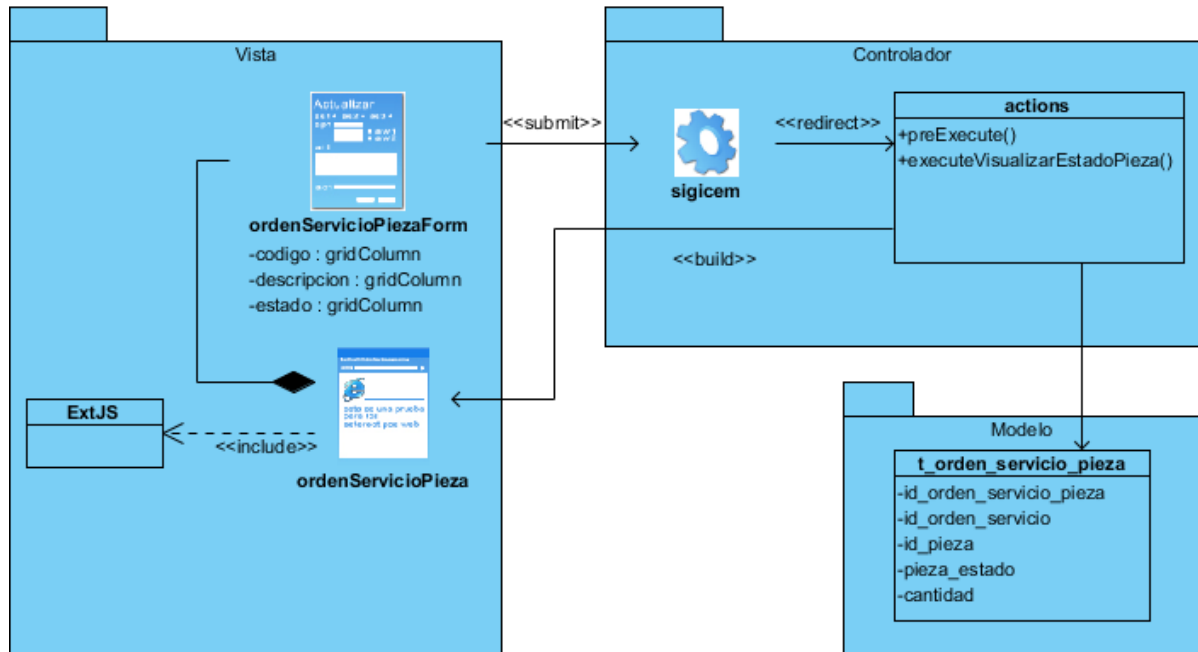


Figura 23. Diagrama de clases del diseño del CU: Visualizar piezas solicitadas para equipos rotos

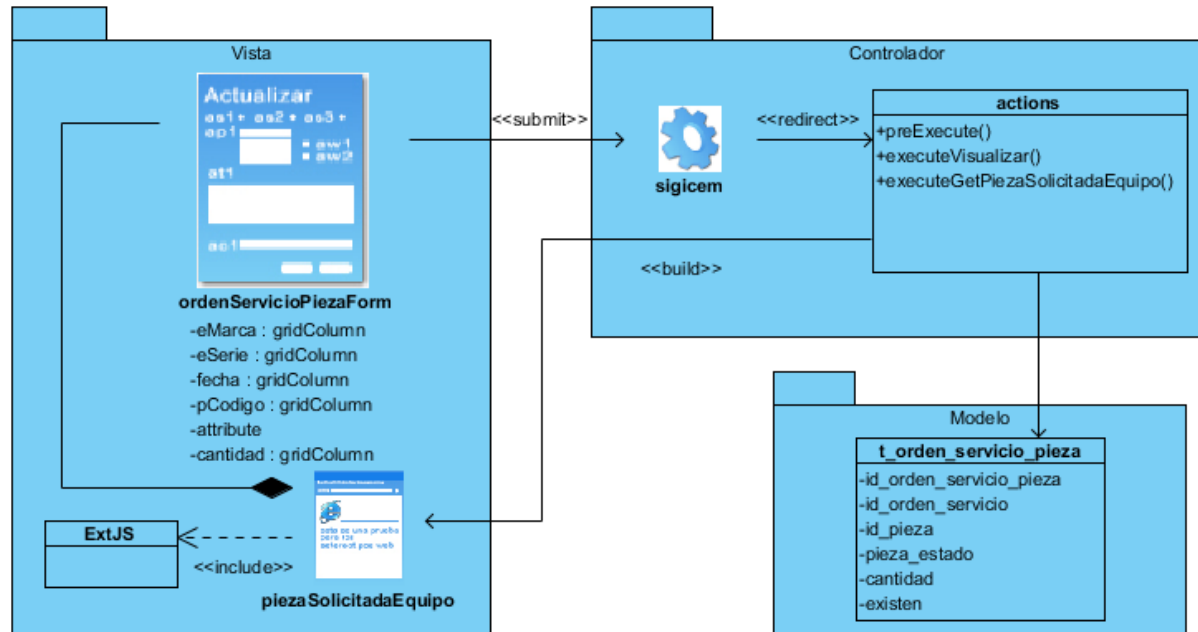
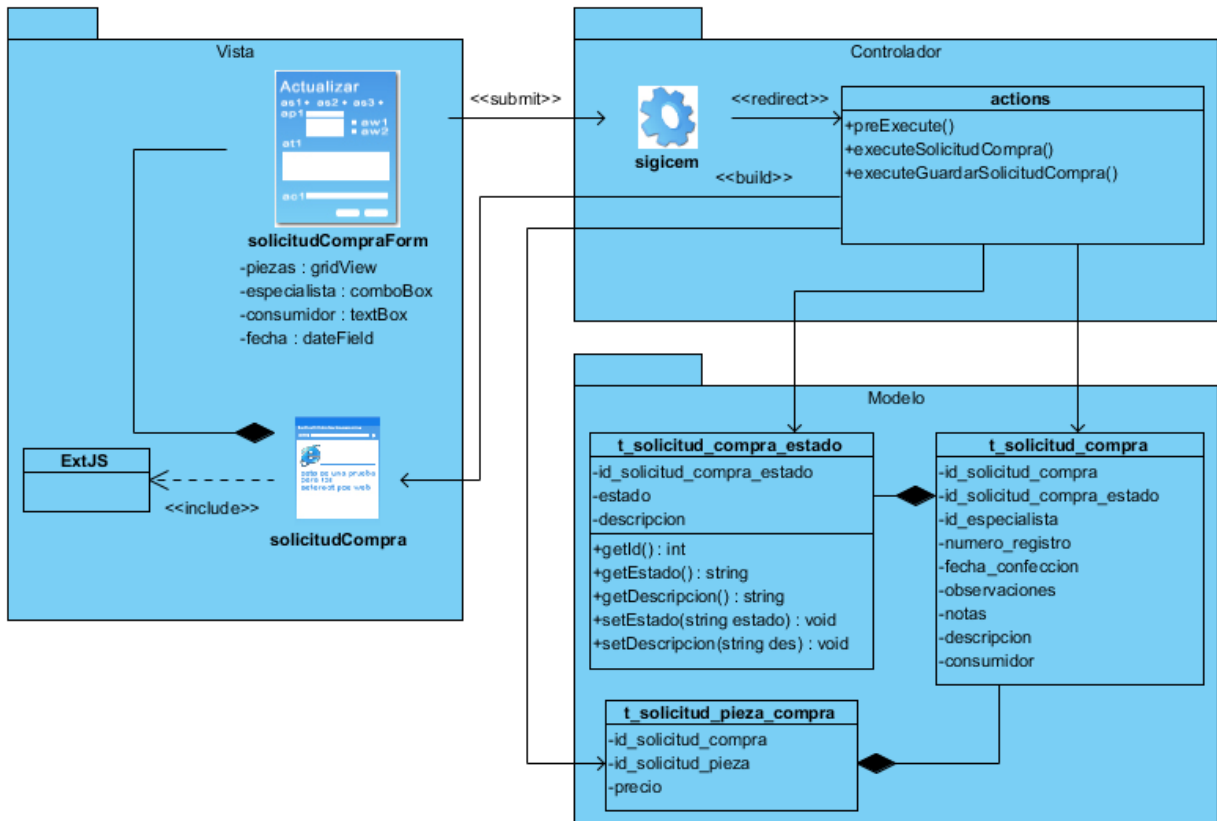


Figura 24. Diagrama de clases del diseño del CU: Crear solicitud de compra



Descripción de las clases del diseño

Nombre de la Clase: sigicem

Descripción: es el punto de entrada único de toda la aplicación, denominada comúnmente como controlador frontal, donde todas las peticiones realizadas son gestionadas. Hace uso del sistema de enrutamiento para realizar la asociación del nombre de una acción y el nombre de un módulo mediante la URL definida por el usuario.

Las clases *acciones* representan la página controladora de un módulo determinado. En ella se encuentra la lógica de la aplicación; además, utilizan el modelo y definen variables para la vista. A continuación, se muestra un ejemplo del *acciones* para el CU Realizar despiece desde equipo, debido a que todas operan de forma similar.

Nombre de la Clase: actions

Descripción: contiene un método a través del cual se muestra la interfaz correspondiente y define además, las variables necesarias para la vista. Otro de los métodos existentes, es el encargado de guardar en la base de datos todos los cambios realizados en la vista. El resto de los métodos son utilizados para mostrar la información necesaria en el proceso de despiece.

Las páginas clientes representan la vista y permiten el acceso del usuario al sistema; esta es la encargada de atender todas las selecciones y peticiones que pueda realizar un determinado usuario al sistema. Seguidamente, se muestra un ejemplo de la página cliente para el CU Realizar despiece desde equipo.

Nombre de la Clase: despieceEquipo

Descripción: contiene todo el código necesario para crear un formulario, mediante el cual el usuario interactúa, para realizar el despiece de un equipo determinado. Además, esta página cliente muestra información referente al equipo seleccionado.

Las clases formularios forman parte de la página cliente, estos representan la entrada y salida de datos en el sistema. Seguidamente, se muestra un ejemplo de la descripción de un formulario, debido a que todos se comportan de manera similar.

Nombre de la Clase: despieceEquipoForm

Descripción: contiene todos los elementos necesarios para recoger la información que un usuario introduce, para así realizar el despiece de un equipo determinado. La misma, es enviada a la clase controladora donde se procesa la información recogida.

Las clases del modelo son la representación Orientada a Objetos (OO) de cada una de las tablas de la base de datos, a través de la cuales es posible el tratamiento en forma de objeto de las entidades. Se debe tener en cuenta que a cada una de las tablas de la base de datos le corresponde una clase perteneciente al modelo. A continuación se muestra un ejemplo.

Nombre de la Clase: TPiezaTNomencladorEquipo

Descripción: esta clase representa a la tabla t_pieza_t_nomenclador_equipo de la base de datos. La misma, contiene un atributo por cada uno de los campos de la tabla y sus respectivos métodos get y set, para modificar u obtener los mismos. En este caso en particular, contiene el atributo id_pieza,

id_nomenclador_equipo, cantidad y def_fabricante. Además, contiene métodos como el *save* y *delete*, los cuales permiten guardar o eliminar respectivamente objetos de la base de datos. Existen también, otros métodos que permiten obtener información de las entidades con las que se relaciona esta clase, sin necesidad de realizar consultas.

3.3 Patrón arquitectónico

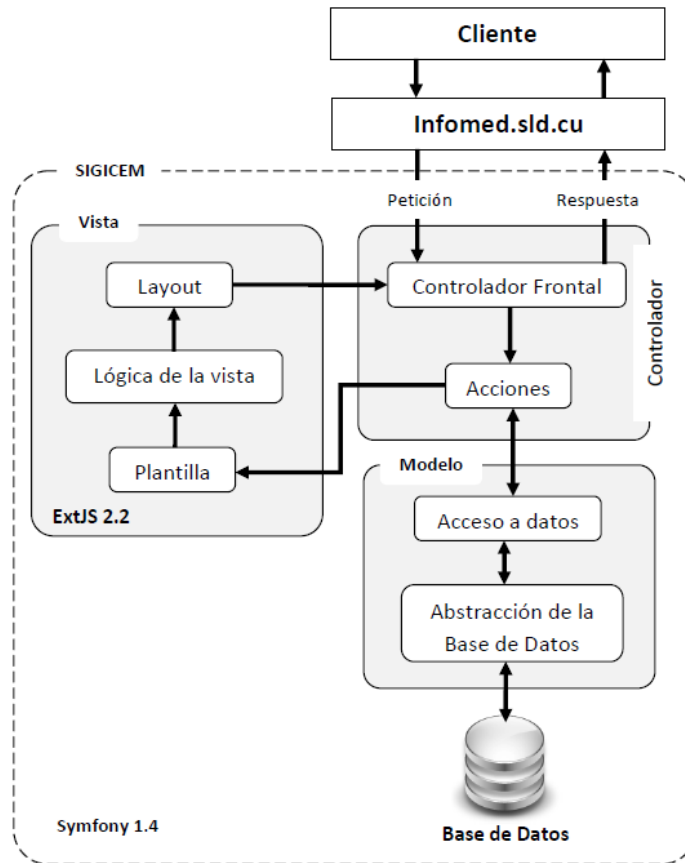
Como se ha podido apreciar, en la realización de las clases del diseño, el patrón arquitectónico empleado en la solución propuesta es el Modelo Vista Controlador (MVC). Este patrón de arquitectura de software, separa los datos de una aplicación, la interfaz de usuario y la lógica de control, en tres capas distintas.

- “El **Modelo** representa la información con la que trabaja la aplicación, es decir, su lógica de negocio.
- La **Vista** transforma el modelo en una página web que permite al usuario interactuar con ella.
- El **Controlador** se encarga de procesar las interacciones del usuario y realiza los cambios apropiados en el modelo o en la vista”. (19)

El MVC, al separar la presentación de la programación, consigue que la aplicación en desarrollo sea más escalable y pueda mantenerse mejor. Permite, que el resultado sea más claro y que la repartición de tareas dentro del equipo de trabajo sea fácil. Además, la depuración de la aplicación es más sencilla y, finalmente, puede utilizarse un marco de trabajo (*o framework*) bien testeado. (25)

A continuación, se muestra la estructura arquitectónica del SIGICEM. (Ver figura 25)

Figura 25. Arquitectura del sistema (2)



3.4 Patrones de diseño

“Los patrones de diseño brindan una solución ya probada y documentada a problemas de desarrollo de software que están sujetos a contextos similares” (26). Su utilización hace posible identificar: clases, instancias, roles, colaboraciones y la distribución de responsabilidades, además de ayudar a construir clases y a estructurar sistemas de clases. (27)

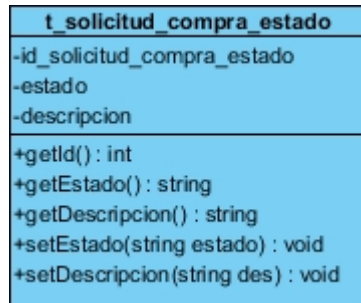
Para el desarrollo de la solución propuesta se utilizó el *framework* de desarrollo Symfony, el cual se encuentra concebido de manera tal que puede afirmarse que obliga al uso de varios patrones. A continuación se describen y ejemplifican aquellos que han sido considerados a utilizar:

GRASP: Patrones de Principios Generales para Asignar Responsabilidades

“Describen los principios fundamentales del diseño de objetos y la asignación de responsabilidades, expresados como patrones”. (28)

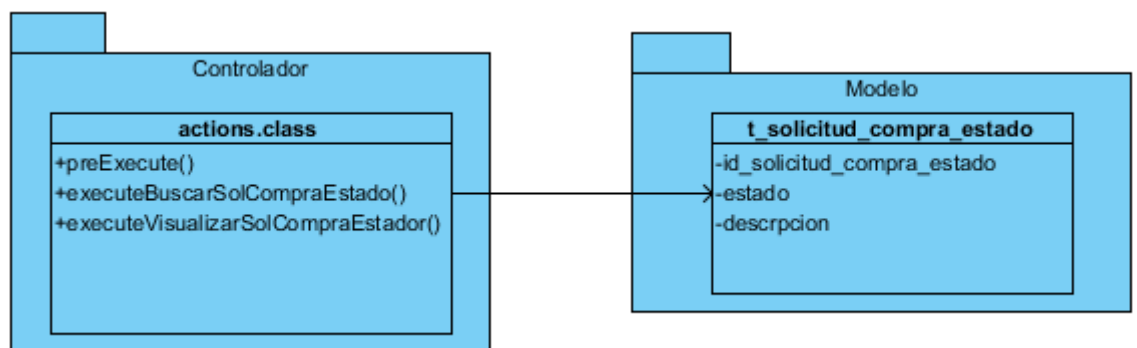
- **Experto:** este patrón se pone de manifiesto al utilizar Propel como ORM, ya que al generar las clases del modelo estas tienen las responsabilidades correspondientes a la información que cada una controla (28). En el siguiente ejemplo se muestra como la clase `t_solicitud_compra_estado` es la responsable de toda la información referente a los estados de la solicitud de compra.

Figura 26. Ejemplo del patrón Experto



- **Creador:** consiste en asignar a una clase determinada la responsabilidad de crear una instancia de otra clase (28). Si estas asignaciones se hacen de la manera correcta se garantiza que no exista aumento en el acoplamiento. En el ejemplo siguiente, se puede observar como la clase `action` de la entidad `t_solicitud_compra_estado` es la responsable de crear las instancias de la clase `TSolicitudCompraEstado`.

Figura 27. Ejemplo del patrón Creador

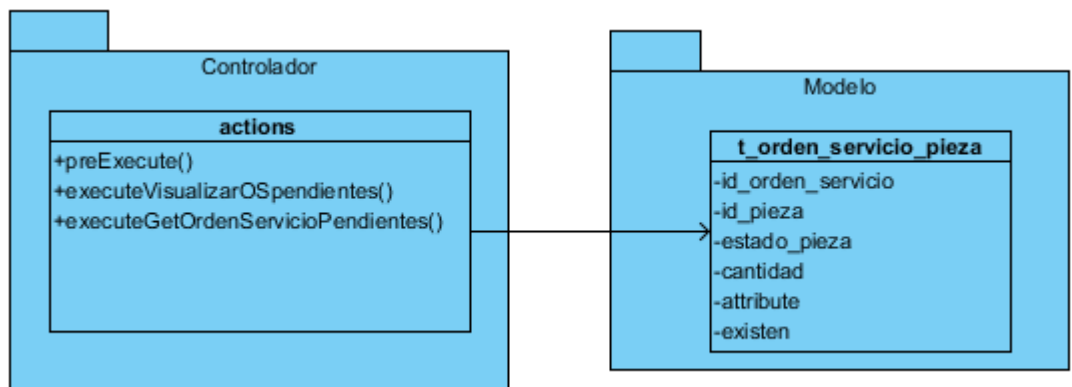


- **Bajo Acoplamiento:** tiene como objetivo asignar una responsabilidad de manera que el acoplamiento permanezca bajo (28). En Symfony se evidencia este patrón en las clases del

modelo, las cuales no se asocian con las vistas sino con los controladores; manteniendo de esta forma una baja dependencia, dando resultado a un bajo acoplamiento.

- **Alta Cohesión:** una clase con alta cohesión tiene un número relativamente pequeño de métodos con funcionalidad altamente relacionada, y no realiza mucho trabajo (28). Un ejemplo de este patrón se observa mediante las clases controladoras, las cuales presentan un pequeño número de funcionalidades relacionadas con el objetivo de dicha clase, evidenciado en la clase *action* de la entidad *t_orden_servicio_pieza*. Esto proporciona un fácil mantenimiento y reutilización de la clase en algunos casos.

Figura 28. Ejemplo del patrón Alta Cohesión



- **Controlador:** sugiere que la lógica de negocios debe estar separada de la capa de presentación, para así aumentar la reutilización del código y a la vez tener un mayor control (29). La existencia del controlador frontal que posee Symfony es un ejemplo claro de este patrón. Dicho controlador se encarga de manejar toda la información referente a las peticiones de los usuarios, la seguridad y la configuración de toda la aplicación, aunque existen otros controladores que tienen como objetivo disminuir la carga del controlador frontal.

GOF: Gang of four o Pandilla de los cuatro

- **Decorador:** Symfony contiene una plantilla global (*layout.php*) para todo el proyecto, en la cual se guarda todo el código que es común en todas las interfaces de la aplicación, evitando tener que repetirlo en cada una de las interfaces. De esta forma el contenido de cada interfaz se integra a la plantilla global, evidenciándose una implementación del patrón Decorador.

En el capítulo, se presentaron los elementos fundamentales de la fase de análisis y diseño, obteniéndose los diagramas de clases del análisis. Igualmente, en el diseño se quedaron definidos los diagramas de comunicación, elaborándose a partir de los artefactos correspondientes a esa etapa; permitiendo la asignación de responsabilidades a las clases, en aras de lograr un mejor entendimiento del funcionamiento interno del módulo. Mediante los diagramas de clases del diseño se logró la interacción entre los componentes para conformar la arquitectura de este.

CAPITULO 4. IMPLEMENTACIÓN

En el capítulo, se expone una descripción del desarrollo de la solución inicialmente propuesta mediante la representación del Modelo de Implementación. Además, se presentan los estándares y estilos de codificación a utilizar; así como, algunos de los criterios tomados en cuenta para llevar a cabo el tratamiento de errores en el módulo de Gestión Tecnológica del Sistema de Gestión para Ingeniería Clínica y la Electromedicina.

4.1 Modelo de Datos

“El modelado de datos responde a una serie de preguntas importantes para cualquier aplicación de procesamiento de datos; tales como: ¿Cuáles son los objetos de datos primarios que van a procesar el sistema? ¿Cuál es la composición de cada objeto de datos y que atributos describe el objeto? ¿Dónde residen actualmente los objetos? ¿Cuál es la relación entre los objetos y los procesos que los transforman?”. (30)

Para dar respuesta a las preguntas antes mencionadas, los métodos de modelado de datos hacen uso del diagrama entidad-relación (DER). Dicho diagrama, permite a través de una notación gráfica, la identificación de los objetos de datos y sus relaciones. El DER, define los datos que se introducen, se almacenan, se transforman y se producen dentro de una aplicación. (Ver anexo 1)

Descripción de las tablas de la base de datos

Tabla 11. Descripción de tablas de la base de datos

| Nombre: t_sistema | | |
|---|-------------|---|
| Descripción: Es la tabla que almacena las relaciones de los sistemas con otros equipos. | | |
| Atributo | Tipo | Descripción |
| id_nomenclador_equipo1 | Integer(10) | Relación con la tabla t_nomenclador_equipo |
| Id_nomenclador_equipo2 | Integer(10) | Relación con la tabla t_nomenclador_equipo |
| cantidad | Integer(11) | Almacena la cantidad de equipos de un tipo que contiene el sistema. |
| Nombre: t_proveedor | | |
| Descripción: Es la tabla que almacena los proveedores de piezas en el momento en que se crean las ofertas. | | |
| Atributo | Tipo | Descripción |

| | | |
|---|-------------|--|
| id_proveedor | Integer(10) | Identificador de la tabla. |
| nombre | Varchar(45) | Almacena el nombre del proveedor. |
| id_pais | Integer(10) | Relación con la tabla t_pais. |
| Nombre: t_oferta | | |
| Descripción: Es la tabla que almacena las ofertas. | | |
| Atributo | Tipo | Descripción |
| id_oferta | Integer(10) | Identificador de la tabla. |
| id_proveedor | Integer(11) | Relación con la tabla t_proveedor. |
| id_solicitud_compra | Integer(10) | Relación con la tabla t_solicitud_compra. |
| id_tipo_moneda | Integer(11) | Relación con la tabla t_tipo_moneda. |
| id_solicitud_compra_estado | Integer(11) | Relación con la tabla t_solicitud_compra_estado. |
| id_especialista | Integer(16) | Relación con la tabla t_especialista. |
| tiempo_entrega | Varchar(20) | Almacena el tiempo de entrega. |
| fecha_oferta | Date | Almacena la fecha de oferta. |
| fecha_tope | Date | Almacena la fecha tope. |
| numero_contrato | Integer(11) | Almacena el número de contrato de la oferta. |
| fecha_recepcion | Date | Almacena la fecha de recepción. |
| numero_oferta | Varchar(20) | Almacena el número de la oferta. |
| Nombre: t_solicitud_compra_estado | | |
| Descripción: Es la tabla que almacena los estados en los que se puede encontrar una solicitud de compra. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_compra_estado | Integer(10) | Identificador de la tabla. |
| descripcion | Text | Almacena la descripción del estado en el que se encuentra la solicitud de compra |
| estado | Varchar(50) | Almacena el estado. |
| Nombre: t_solicitud_compra | | |
| Descripción: Es la tabla que almacena todas las solicitudes de compra que se han realizado. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_compra | Integer(10) | Identificador de la tabla. |
| id_solicitud_compra_estado | Integer(10) | Relación con la tabla t_solicitud_compra_estado. |
| id_especialista | Integer(16) | Relación con la tabla t_especialista. |
| numero_registro | Integer(10) | Almacena el número de registro. |

| | | |
|---|--------------|---|
| fecha_confeccion | Date | Almacena la fecha de confección. |
| observaciones | Varchar(45) | Almacena las observaciones. |
| notas | Varchar(45) | Almacena las notas. |
| descripcion | Varchar(45) | Almacena las descripciones. |
| consumidor | Varchar(45) | Almacena el consumidor. |
| Nombre: t_solicitud_compra_pieza | | |
| Descripción: Es la tabla que almacena las piezas de las solicitudes de compra. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_compra | Integer(10) | Relación con la tabla t_solicitud_compra. |
| id_pieza | Integer(10) | Relación con la tabla t_pieza. |
| precio | Integer(10) | Almacena el precio |
| Nombre: t_pieza | | |
| Descripción: Se utiliza para representar existencia de las piezas. | | |
| Atributo | Tipo | Descripción |
| id_pieza | Integer(10) | Identificador de la tabla. |
| id_nomenclador_equipo | Integer(10) | Relación con la tabla t_nomenclador_equipo. |
| id_unidad_medida | Integer(10) | Relación con la tabla t_unidad_medida. |
| descripción | Varchar(255) | Almacena la descripción de la pieza. |
| ref_fab | Varchar(255) | Almacena la referencia del fabricante. |
| gastable | Boolean | Almacena si es gastable o no la pieza. |
| precio_unitario | Varchar(9) | Almacena el precio de la pieza. |
| codigo | Varchar(50) | Almacena el código de la pieza. |
| id_producto | Integer(10) | Relación con la tabla t_producto. |
| Nombre: t_orden_servicio_pieza | | |
| Descripción: Es la tabla que almacena las piezas que se utilizan en las ordenes de servicio y el estado de las mismas. | | |
| Atributo | Tipo | Descripción |
| id_orden_servicio | Integer(10) | Relación con la tabla t_orden_servicio. |
| id_pieza | Integer(10) | Relación con la tabla t_pieza. |
| cantidad | Integer(10) | Almacena la cantidad. |
| existencia | Integer(10) | Almacena la cantidad de piezas existentes en el momento solicitado. |

| | | |
|---|--------------|--|
| estado | Varchar(100) | Describe el estado de gestión en el que se encuentra la pieza. |
| Nombre: t_solicitud_pieza | | |
| Descripción: Es la tabla que almacena todas las solicitudes de piezas que se han realizado. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_pieza | Integer(10) | Identificador de la tabla. |
| id_pieza | Integer(11) | Relación con la tabla t_pieza. |
| id_especialista | Integer(11) | Relación con la tabla t_especialista. |
| fecha_solicitud | Date | Almacena la fecha de solicitud. |
| id_orden_servicio | Integer(11) | Relación con la tabla t_orden_servicio. |
| cant_necesaria | Integer(11) | Almacena la cantidad necesaria. |
| disponible | Boolean | Indica si la pieza se encuentra disponible para realizar una solicitud de compra |
| Nombre: t_reporte_pieza_os | | |
| Descripción: Es la tabla que almacena todos los reportes de piezas que se han realizado para satisfacer una orden de servicio. | | |
| Atributo | Tipo | Descripción |
| id_reporte_pieza_os | Integer(11) | Identificador de la tabla. |
| id_orden_servicio | Integer(10) | Relación con la tabla t_orden_servicio. |
| id_pieza | Integer(10) | Relación con la tabla t_pieza. |
| cantidad | Integer(11) | Almacena la cantidad de las piezas. |
| Nombre: t_pieza_nomenclador_equipo | | |
| Descripción: Es la tabla que almacena las piezas para un equipo determinado. | | |
| Atributo | Tipo | Descripción |
| id_nomenclador_equipo | Integer(10) | Relación con la tabla t_nomenclador_equipo. |
| id_pieza | Integer(10) | Relación con la tabla t_pieza. |
| cantidad | Integer(10) | Almacena la cantidad. |
| def_fabricante | Boolean | Indica si la pieza está definida por el fabricante. |
| Nombre: t_solicitud_movimiento_pieza | | |
| Descripción: Es la tabla que almacena las piezas que se utilizan en una solicitud de movimiento de almacén. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_movimiento_pieza | Integer(10) | Identificador de la pieza. |

| | | |
|--|-------------|---|
| id_pieza | Integer(11) | Relación con la tabla t_pieza. |
| cantidad | Integer(11) | Almacena la cantidad de piezas. |
| Nombre: t_solicitud_movimiento_almacen | | |
| Descripción: Se utiliza para representar una solicitud de movimiento de almacén, para así pedir piezas a otros almacenes. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_movimiento_almacen | Integer(10) | Identificador de la tabla. |
| id_especialista_recibe | Integer(16) | Relación con la tabla t_especialista. |
| id_especialista_solicita | Integer(16) | Relación con la tabla t_especialista. |
| estado | Varchar(20) | Almacena el estado. |
| vale_entrega | Integer(10) | Almacena el vale de entrega. |
| numero_lote | Integer(10) | Almacena el número de lote. |
| fecha | Date | Almacena la fecha. |
| numero_solicitud | Integer(10) | Almacena el número de solicitud. |
| organismo | Varchar(50) | Almacena el organismo. |
| empresa | Varchar(50) | Almacena la empresa. |
| id_unidad_salud | Integer(16) | Relación con la tabla t_unidad_salud. |
| id_almacen | Integer(10) | Relación con la tabla t_almacen. |
| id_almacen_solicita | Integer(10) | Relación con la tabla t_almacen. |
| centro_costo | Varchar(50) | Almacena el centro de costo. |
| id_especialista_autoriza | Integer(16) | Relación con la tabla t_especialista. |
| id_orden_servicio | Integer(10) | Relación con la tabla t_orden_servicio. |
| Nombre: t_solicitud_material | | |
| Descripción: Es la tabla que almacena la información necesaria en el momento que se satisface una orden de servicio. | | |
| Atributo | Tipo | Descripción |
| id_solicitud_material | Integer(10) | Identificador de la tabla. |
| id_especialista_recibe | Integer(16) | Relación con la tabla t_especialista. |
| id_especialista_solicita | Integer(16) | Relación con la tabla t_especialista. |
| vale_entrega | Integer(16) | Almacena el vale de entrega. |
| codigo_unidad | Varchar(45) | Almacena el código de unidad. |
| numero_lote | Integer(10) | Almacena el número de lote. |

| | | |
|--------------------------|-------------|---|
| fecha | Date | Almacena la fecha. |
| numero_solicitud | Integer(10) | Almacena el número de solicitud. |
| organismo | Varchar(50) | Almacena el organismo. |
| empresa | Varchar(50) | Almacena el nombre de la empresa. |
| id_unidad_salud | Integer(16) | Relación con la tabla t_unidad_salud. |
| id_almacen | Integer(10) | Relación con la tabla t_almacen. |
| id_orden_servicio | Integer(10) | Relación con la tabla t_orden_servicio. |
| centro_costo | Varchar(50) | Almacena el centro de costo. |
| Id_especialista_autoriza | Integer(16) | Relación con la tabla t_especialista. |

4.2 Modelo de Implementación

El flujo de trabajo de implementación describe cómo los elementos del modelo del diseño se implementan en términos de componentes y cómo estos se organizan de acuerdo a los nodos específicos en el modelo de despliegue (21). La implementación tiene como propósito definir la organización del código e implementar el diseño de elementos en términos de componentes; es decir, ficheros de código fuente, binarios y ejecutables, entre otros. Además, permite probar los componentes desarrollados como unidades e integrar los resultados individuales en un sistema ejecutable.

Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y sus relaciones. “Un componente es una unidad de código fuente que sirve como un bloque de construcción para la estructura física de un sistema. Las clases que se agrupan en un componente son aquellas que o bien tienen funciones cooperativas o las que necesitan estar en una proximidad cercana por eficiencia de implementación”. (31)

Figura 29. Diagrama de componentes del módulo de Gestión Tecnológica v2.0

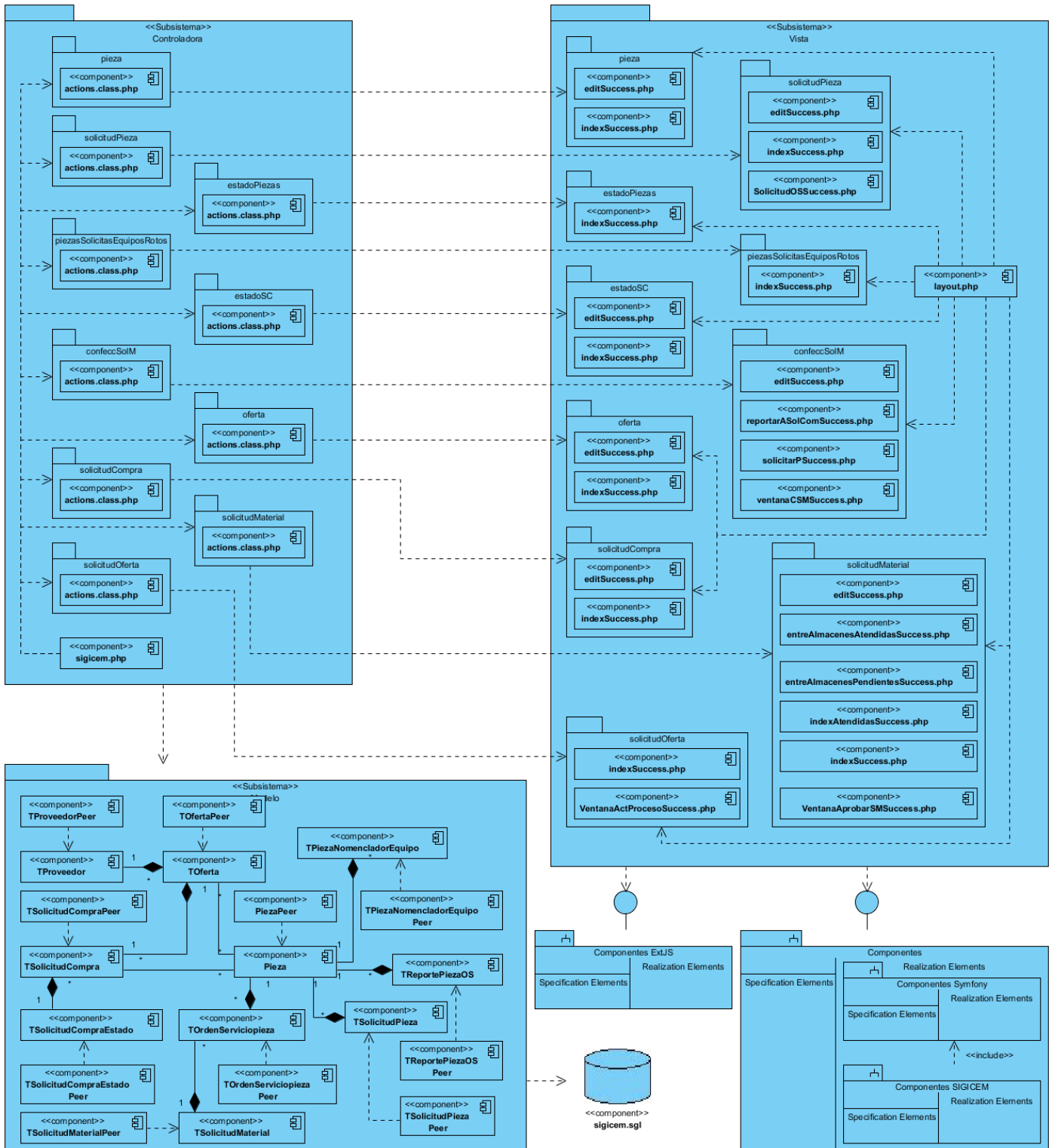
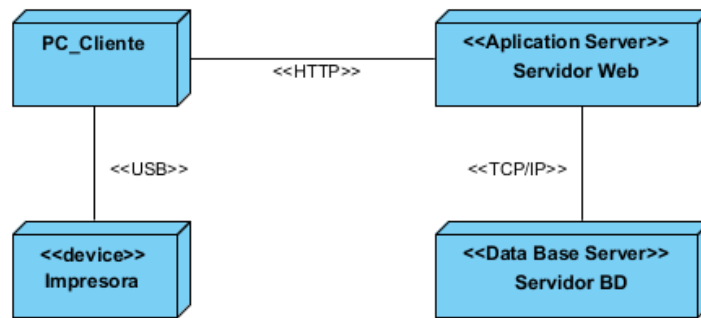


Diagrama de despliegue

El diagrama de despliegue se utiliza para modelar los aspectos físicos sobre los que se ejecutarán los componentes del sistema de software; encargándose de mostrar las conexiones existentes entre procesadores y dispositivos. Para ello, cuenta con varios nodos utilizados para modelar la topología del hardware sobre el que se ejecuta un sistema. Un nodo representa un dispositivo o procesador sobre el cual se pueden desplegar los componentes.

Figura 30. Diagrama de despliegue del módulo de Gestión Tecnológica v2.0



4.3 Estándares y estilos de codificación

Los estándares y estilos de codificación, son una forma de normalizar la programación de forma tal, que al trabajar en un proyecto, cualquier persona involucrada en el mismo tenga acceso y comprenda el código, permitiendo definir la escritura y organización del código fuente de un sistema; así como, la forma en que deben ser declaradas las variables y comentarios, entre otros. Facilita a los programadores el mantenimiento del código y logra que los mismos trabajen de forma coordinada.

A continuación, se presentan los estándares y estilos de codificación utilizados en la implementación del módulo de Gestión Tecnológica.

Tabla 12. Estándares y estilos de codificación

| Identación | |
|---|--|
| Objetivo: lograr una estructura uniforme para los bloques de código así como para los diferentes niveles de anidamiento. | |
| Inicio y fin de bloque | Se recomienda dejar dos espacios en blanco desde la instrucción anterior para el inicio y fin de bloque {}. Lo mismo sucede para el caso de las instrucciones if, else, for, while, do while, switch, foreach. |

| | | |
|--|---|---|
| Aspectos Generales | <p>El indentado debe ser de dos espacios por bloque de código. No se debe usar el tabulador; ya que este puede variar según la PC o la configuración de dicha tecla.</p> <p>Los inicios ({) y cierre (}) de ámbito deber estar alineados debajo de la declaración a la que pertenecen y deben evitarse si hay sólo una instrucción.</p> <p>Nunca colocar ({) en la línea de un código cualquiera, esto requiere una línea propia.</p> | |
| Comentarios, separadores, líneas, espacios en blanco y márgenes. | | |
| Objetivo: Establecer un modo común para comentar el código de forma tal que sea comprensible con sólo leerlo una vez. | | |
| Ubicación de comentarios | Al inicio de cada clase o función y al final de cada bloque de código. | Se recomienda comentar al inicio de la clase o función especificando el objetivo de la misma así como los parámetros que usa (especificar tipos de dato, y objetivo del parámetro) entre otras cosas. |
| Líneas en blanco | Se emplean antes y después de métodos, clases y estructuras. | Se recomienda dejar una línea en blanco antes y después de la declaración de una clase o de una estructura y de la implementación de una función. |
| Espacios en blanco | Entre operadores lógicos y aritméticos. | Se recomienda usar espacios en blanco entre estos operadores para lograr una mayor legibilidad en el código. Ejemplo: producto = nomproducto |
| Aspectos generales | Sobre el comentario. | Se debe evitar comentar cada línea de código. Cuando el comentario se aplica a un grupo de instrucciones debe estar seguido de una línea en blanco. En caso de que se necesite comentar una sola instrucción se suprime la línea en blanco o se escribe a continuación de la instrucción. |
| | Sobre los espacios en blanco. | No se debe usar espacio en blanco: <ul style="list-style-type: none"> ▪ Después del corchete abierto y antes del cerrado de un arreglo. ▪ Después del paréntesis abierto y antes del cerrado. ▪ Antes de un punto y coma. |
| Variables y constantes | | |
| Apariencia de variables | Las variables tendrán un | El nombre que se le da a las variables debe comenzar |

| | | |
|---|---|--|
| | prefijo para el tipo de datos en minúscula. | con la primera letra en minúscula, en caso de que sea un nombre compuesto se empleará notación lowerCamelCase*. Ejemplo: sNombrePaciente |
| Apariencia de constantes | Todas sus letras en mayúscula. | Se deben declarar las constantes con todas sus letras en mayúscula. |
| Aspectos generales | Nombres de las variables y constantes. | El nombre empleado, debe permitir que con sólo leerlo se conozca el propósito de la misma. |
| Clases y Objeto | | |
| Objetivo: Nombrar las clases e instancias de forma estándar para toda la aplicación. | | |
| Apariencia de clases y objetos | Primera letra en mayúscula. | Los nombres de las clases deben comenzar con la primera letra en mayúscula y el resto en minúscula, en caso de que sea un nombre compuesto se empleará notación PascalCasing*. Ejemplo: MiClase (). |
| Apariencia de atributos | Primera letra en minúscula. | El nombre que se le da a los atributos de las clases debe estar escrito con letras minúsculas, en caso de que sea un nombre compuesto se utilizará underscore (_) para separarlo. Ejemplo: mi_atributo. |
| Apariencia de las funciones | Primera letra en mayúscula | Para nombrar las funciones se debe tratar de utilizar verbos que denoten la acción que hace la función. Se empleará notación PascalCasing. Ejemplo: function BuscarUnidad(). Si son funciones que obtienen un dato se emplea el prefijo get y si fijan algún valor se emplea el prefijo set. |
| Aspectos generales | Sobre las clases, los objetos, los atributos y las funciones. | El nombre empleado para las clases, objetos, atributos y funciones debe permitir que con sólo leerlo se conozca el propósito de los mismos. |
| Bases de Datos, Tablas, Esquemas y Campo | | |
| Apariencia de las tablas | La primera letra | El nombre a emplear para las tablas debe comenzar |

* Los identificadores y nombres de variables, métodos y funciones están compuestos por múltiples palabras juntas iniciando cada palabra con letra mayúscula excepto la primera palabra que debe iniciar con minúscula.

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

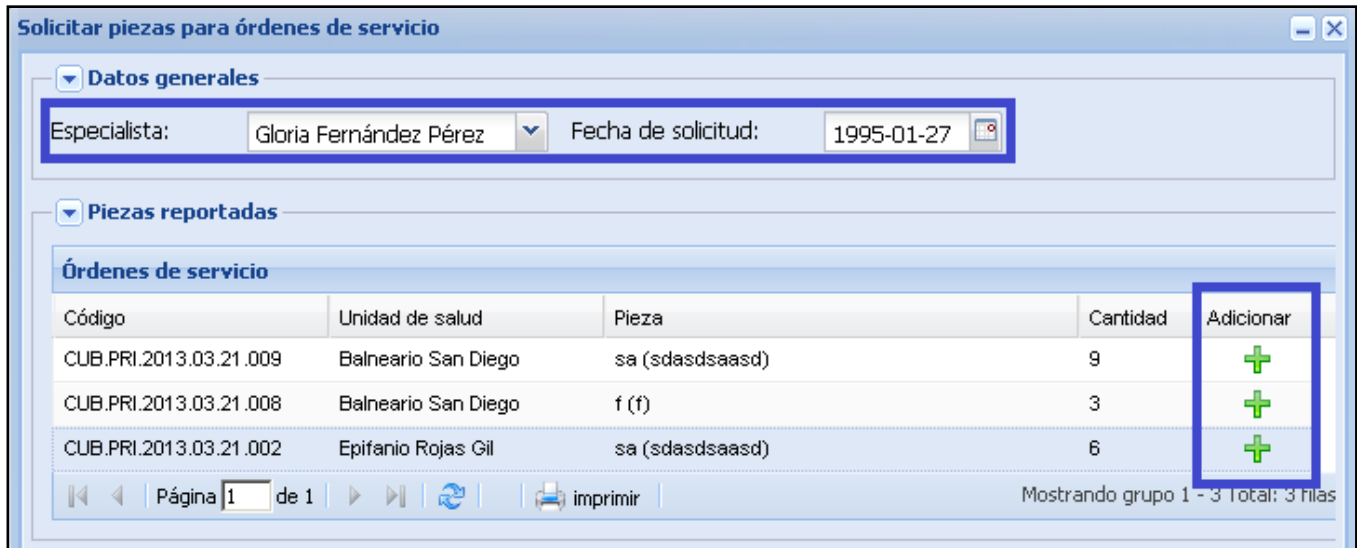
| | | |
|--------------------------|--|---|
| | representa el tipo. Todas las letras en minúscula. | con el prefijo “t” seguido de underscore y luego debe escribirse todas las letras en minúscula, en caso de que sea un nombre compuesto se utilizara underscore para separarlo. Ejemplo: t_pieza. |
| Apariencia de los campos | Todas las letras en minúscula. | El nombre a emplear para los campos debe escribirse con todas las letras en minúscula para evitar problemas con el Case Sensitive del gestor, en caso de que sea un nombre compuesto se utilizará underscore (_) para separarlo <i>Ejemplo: precio_pieza.</i> |
| Nombre de los campos | En caso de identificadores. | Todos los campos identificadores van a comenzar con el identificador id seguido de underscore (_) y posteriormente el nombre del campo Ejemplo: id_pieza. |
| Aspectos generales | Sobre las BD, vistas, tablas atributos y procedimientos. | El nombre empleado para las tablas y los campos de la base de datos, deben permitir que con sólo leerlos se conozca el propósito de los mismos. |

4.4 Tratamiento de errores

En la implementación de un sistema es de gran importancia el tratamiento de errores, ya que este garantiza la confidencialidad de la información que dicho sistema gestiona. En el módulo de Gestión Tecnológica, con el fin de evitar situaciones indeseadas que se conviertan en problemas que puedan afectar el funcionamiento del sistema, se adoptaron varias medidas.

Entre ellas se encuentra la validación de los datos introducidos por el usuario; las cuales son realizadas, en el lado del cliente mediante expresiones regulares y en el lado del servidor aprovechando las potencialidades de validación de formularios que posee el *framework* Symfony. Igualmente, se manifiestan mediante la invocación del método *isValid* presente en todos los formularios del *framework*, mediante el cual se validan los datos entrados por el usuario e informa los problemas encontrados en cada campo. Este método comprueba que cada uno de los campos que entra el usuario que interactúa con la aplicación se corresponda al tipo de datos con los que fueron declarados en la base de datos. Además, se cuenta con cuadros de opción y menú de selección, los cuales facilitan la entrada de datos y evitan que el usuario introduzca información errónea.

Figura 31. Ejemplo de tratamiento de errores



4.5 Seguridad

En el módulo de Gestión Tecnológica hay múltiples usuarios que estarán interactuando con el mismo, por lo tanto es de vital importancia que cada uno solo acceda a las funcionalidades que le han sido destinadas según el rol que desempeña en el centro. Para garantizar dicho objetivo se utilizará el *plugin sfGuardPlugin*, utilizado en *Symfony* para manejar la seguridad en cuanto a autenticación y autorización. Mediante la utilización de este *plugin* los usuarios tendrán que iniciar sesión para acceder al sistema y en dependencia de los privilegios que tenga cada uno asignado, accederán a las funcionalidades correspondientes del módulo.

Con la confección del presente capítulo, se logró la comprensión de los elementos relacionados con la etapa de implementación del sistema en cuestión. Se obtuvo el diagrama de componentes y la distribución física de los nodos que componen el sistema a través del diagrama de despliegue. Igualmente, se identificaron los estándares y estilos de codificación a utilizar en la implementación de los componentes definidos, así como las pautas a tener en cuenta durante el tratamiento de errores.

CONCLUSIONES

Una vez finalizada la investigación para el desarrollo de la solución propuesta, se arribaron a las siguientes conclusiones:

- El análisis de la primera versión del módulo de Gestión Tecnológica, evidenció que este no se ajusta a las necesidades actuales del Centro de Ingeniería Clínica y Electromedicina.
- Con el objetivo de destacar la necesidad de realizar la investigación propuesta, se efectuó un estudio de los sistemas de gestión tecnológica existentes a nivel internacional y nacional, determinándose que no constituyen una solución factible para aplicarlos en el Sistema de Gestión para Ingeniería Clínica y Electromedicina; debido a que la mayoría presentan características propias de la institución para la cual fueron diseñados o son aplicaciones de escritorio que no están soportadas para trabajar en redes.
- A través de entrevistas realizadas al cliente, se caracterizó el proceso de gestión tecnológica llevado a cabo en el Centro de Ingeniería Clínica y Electromedicina, lo que posibilitó obtener una guía para el diseño e implementación de la solución propuesta.
- Mediante la confección de los artefactos pertenecientes a las Disciplinas de Trabajo definidas y guiados por el Proceso Unificado de Desarrollo, se logró obtener la documentación necesaria de la investigación.
- La utilización de las herramientas y tecnologías establecidas por el proyecto Sistema de Gestión para Ingeniería Clínica y Electromedicina permitió una correcta implementación del módulo de Gestión Tecnológica.

RECOMENDACIONES

Luego de haber concluido el presente trabajo el equipo de desarrollo recomienda:

- Integrar el módulo de Gestión Tecnológica con el módulo Predicción del stock de piezas de repuesto para equipos médicos, con el fin de obtener una predicción de las piezas que deben ser compradas por plan anual.
- Implementar funcionalidades que permitan el envío de archivos adjuntos en correos electrónicos, con el objetivo de agilizar a través de estos, la obtención de ofertas para satisfacer las solicitudes de compra.

REFERENCIAS BIBLIOGRÁFICAS

1. **Encinosa, Lázaro J. Blanco.** La informática en la dirección de empresas. La Habana : Félix Varela, 2011. pág. 1. ISBN 978-959-07-1629-4.
2. **Hernández, Dennys.** Dirección de Información. SIGICEM: Sistema de gestión para Ingeniería Clínica y Electromedicina. [En línea] 9 de Marzo de 2012. [Citado el: 10 de Junio de 2013.] http://repositorio_institucional.uci.cu/jspui/handle/ident/3990. ISBN 978-959-286-019-3.
3. **Nuchera, Antonio Hidalgo.** La gestión de la tecnología como factor estratégico de la competitividad industrial. Madrid : Economía Industrial, 1999. 330 VI.
4. **Real Academia Española.** DICCIONARIO DE LA LENGUA ESPAÑOLA - Vigésima segunda edición. [En línea] 2013. [Citado el: 5 de Junio de 2013.] <http://lema.rae.es/drae/?val=tecnología>.
5. **Real Academia Española.** DICCIONARIO DE LA LENGUA ESPAÑOLA - Vigésima segunda edición. [En línea] 2013. [Citado el: 5 de Junio de 2013.] <http://lema.rae.es/drae/?val=gestión>.
6. **Ochoa, Migdely.** Biblioteca Virtual en Salud. Innovación, tecnología y gestión tecnológica. [En línea] Centro Nacional de Información de Ciencias Médicas-Infomed, 9 de Agosto de 2007. [Citado el: 3 de Abril de 2013.] http://bvs.sld.cu/revistas/aci/vol16_4_07/aci081007.html.
7. **Cordua, S. Joaquín.** Tecnología y desarrollo tecnológico. Santiago de Chile : CINDA, 1994.
8. **¿Qué son los sistemas de gestión?** BSI. [En línea] 2012. [Citado el: 1 de Octubre de 2012.] <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>.
9. **Ruiz, Lucas M. Palavecino.** Scribd. Sistema Informático de Gestión de Departamentos de Ingeniería Clínica. [En línea] 4 de Abril de 2012. [Citado el: 6 de Marzo de 2013.] <http://es.scribd.com/search?query=sistema+inform%C3%A1tico+de+gesti%C3%B3n+de+departamentos+de+ingenier%C3%ADa+cl%C3%ADnica>.
10. **EasyMaint.** [En línea] aBESofT Technologies, 2013. [Citado el: 8 de Marzo de 2013.] www.abe-soft.com/EasyMaint.
11. **CWORKS.** Computerized Maintenance Management System (CMMS). [En línea] CWORKS SYSTEMS BERHAD, 2002-2013. [Citado el: 18 de Febrero de 2013.] <http://www.cworks.com.my/>.
12. **Ranniel Rivero, Velmour Muñoz, Zoila E. Morales, Dennys J. Hernández.** SACCEM: SISTEMA AUTOMATIZADO CUBANO PARA EL CONTROL DE EQUIPOS MÉDICOS. Dirección de Información. [En línea] 9 de Marzo de 2012. [Citado el: 15 de Noviembre de 2012.] http://repositorio_institucional.uci.cu/jspui/handle/ident/3978. ISBN 978-959-286-019-3.

13. **José A. Senra, Elier Broche.** SISTEMA DE GESTIÓN TECNOLÓGICA HOSPITALARIA V 1.0. Google Académico. [En línea] 2000. [Citado el: 4 de Noviembre de 2012.] http://scholar.googleusercontent.com/scholar?q=cache:RypmlUVvw_cJ:scholar.google.com/+%22sistemas+de+gestion+tecnologica%22&hl=es&as_sdt=0.
14. **Cabrera, O.** ReporTech: Gestión de Tecnología Médica. La Habana: Sociedad Cubana de Bioingeniería, 2007. ISBN 978-959-212-236-9.
15. **Luis Vilcahuamán, Rossana Rivas.** Ingeniería Clínica y Gestión de Tecnología en Salud: Avances y Propuestas. Lima : GRAMBS Corporación Gráfica S.A.C, 2006. ISBN 9972-2885-0-1.
16. **B. Mahadevan, David F. Pyke, Moritz Fleischmann.** "Periodic review, push inventory policies for remanufacturing". [En línea] [Citado el: 20 de Noviembre de 2012.] <http://www.sciencedirect.com/science>.
17. **McClure, Carma.** Introducción a Herramientas CASE y System Architect. MTP. [En línea] 2002. [Citado el: 20 de Noviembre de 2012.] http://users.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf.
18. **Eugene Levner, Yael Perlman, T.C.E. Cheng, Ilya Levner.** "A network approach to modeling the multi-echelon spare-part inventory system with backorders and interval-valued demand". [En línea] [Citado el: 22 de Noviembre de 2012.] <http://www.sciencedirect.com/science>.
19. **Fabien Potencier, François Zaninotto.** Symphony 1.2, la guía definitiva. 2008.
20. **George E. P. Box, William G. Hunter, J. Stuart Hunter.** Statistics for Experimenters An Introduction to Design, Data Analysis, and Model Building. s.l. : Library of Congress Cataloging in Publication Data, 1978.
21. **I. Jacobson, G. Booch, J. Rumbaugh.** El Proceso Unificado de Desarrollo de Software. Madrid : Pearson Educación ,S.A, 2000. ISBN: 84-7829-036-2.
22. **Business Rules Group.** Manifiesto de Reglas de Negocio. [En línea] Noviembre de 2005. [Citado el: 1 de Junio de 2013.] <http://www.businessrulesgroup.org/brmanifesto/>.
23. **Patricia López, Francisco Ruiz.** INGENIERÍA DEL SOFTWARE I, Lenguaje Unificado de Modelado - UML. Universidad Cantabria : Facultad de Ciencias, 2011.
24. **Torres, Manolo.** INFORMATION SYSTEMS GROUP TIC-194. [En línea] [Citado el: 3 de Abril de 2013.] <http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf>.
25. **Argulo, Iván.** SGM. Diseño web y programación. Modelo Vista Controlador. [En línea] [Citado el: 4 de Abril de 2013.] <http://www.sgmweb.es/modelo.asp>.

26. **Tedeschi, Nicolás.** msdn. ¿Qué es un Patrón de Diseño? [En línea] 2013. [Citado el: 1 de Junio de 2013.] <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.
27. **Delgado, José Miguel Ruis.** slidesShare. Patrones de diseño. [En línea] 24 de Julio de 2009. [Citado el: 2 de Mayo de 2013.] <http://www.slideshare.net/jmruizforem/patrones-de-diseo-en-e-learning>.
28. **Iarman, Craig.** UML y Patrones. Mexico : Down Speth White, 2000. ISBN 970-17-0261-1.
29. **Visconti, Marcello.** Departamento de Informática. Fundamentos de Ingeniería de Software. [En línea] 2013. [Citado el: 2 de Junio de 2013.] www.inf.utfsm.cl/~visconti/ili236/Documentos/08-Patrones.pdf.
30. **Pressman, Roger.** Ingeniería de Software. Un enfoque práctico. Madrid : Biblioteca especializada, 2001.
31. **Scribd.** Fase de Implementación. [En línea] 3 de Febrero de 2010. [Citado el: 6 de Abril de 2013.] <http://es.scribd.com/doc/27701099/Explicacion-de-la-fase-de-IMPLEMENTACION-de-la-metodologia-RUP>.

BIBLIOGRAFÍA

¿Qué son los sistemas de gestión?BSI. [En línea] 2012. [Citado el: 1 de Octubre de 2012.] <http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>

Argulo, Iván. SGM. Diseño web y programación. Modelo Vista Controlador. [En línea] [Citado el: 4 de Abril de 2013.] <http://www.sgmweb.es/modelo.asp>.

B. Mahadevan, David F. Pyke, Moritz Fleischmann. “Periodic review, push inventory policies for remanufacturing”. [En línea] [Citado el: 20 de Noviembre de 2012.] <http://www.sciencedirect.com/science>.

Business Rules Group. Manifiesto de Reglas de Negocio. [En línea] Noviembre de 2005. [Citado el: 1 de Junio de 2013.] <http://www.businessrulesgroup.org/brmanifesto/>.

Cabrera, O. ReporTech: Gestión de Tecnología Médica. La Habana : Sociedad Cubana de Bioingeniería, 2007. ISBN 978-959-212-236-9.

Cerda, Felipe. NetBeans: El único IDE que necesitas. Sun. [En línea] [Citado el: 25 de Noviembre de 2012.] <http://blogs.sun.com/concepcion>.

Concepto de Gestión.Definición de. [En línea] 2008. [Citado el: 20 de Octubre de 2012.] <http://definicion.de/gestion/>.

Cordua, S. Joaquín. Tecnología y desarrollo tecnológico. Santiago de Chile : CINDA, 1994

Cruz, Antonio Miguel.Gestion Tecnologica Hospitalaria. [ed.] Universidad del Rosario. ISBN 9587380940, 9789587380941.

Cruz, Elizabeth De la. Propuesta de herramienta CASE para los proyectos del Centro de Desarrollo de Informatica Industrial (CEDIN). Dirección de Información. [En línea] 2010. [Citado el: 20 de Noviembre de 2012.] http://repositorio_institucional.uci.cu/jspui/handle/ident/TD_03818_10.pdf.

CWORKS. Computerized Maintenance Management System (CMMS). [En línea] CWORKS SYSTEMS BERHAD, 2002-2013. [Citado el: 18 de Febrero de 2013.] <http://www.cworks.com.my/>.

Delgado, José Miguel Ruis. slidesShare. Patrones de diseño. [En línea] 24 de Julio de 2009. [Citado el: 2 de Mayo de 2013.] <http://www.slideshare.net/jmruizforem/patrones-de-diseo-en-e-learning>.

Dennis J. Hernández, Ranniel River, Zoila E. Morales, Velmour Muñoz.SIGICEM: SISTEMA DE GESTIÓN PARA LA INGENIERÍA CLÍNICA Y ELECTROMEDICINA. Dirección de Información. [En línea] 9 de Marzo de 2012. [Citado el: 18 de Noviembre de 2012.] http://repositorio_institucional.uci.cu/jspui/handle/ident/3990. ISBN 978-959-286-019-3.

Diccionario de la lengua española. WordReference.com. [En línea] 2005. [Citado el: 2 de Noviembre de 2012.] <http://www.wordreference.com/definicion/despiece>.

Documentos. Biblioteca Nacional de Maestros. [En línea] [Citado el: 20 de Octubre de 2012.] <http://www.bnm.me.gov.ar/>.

Dr. Diego, Lz. de Ipiña, Gz. de Artaza. Introducción a Ruby on Rails. Scribd. [En línea] Julio de 2006. [Citado el: 29 de Noviembre de 2012.] <http://es.scribd.com/doc/54805894/71/Caracteristicas-LAMP>.

EasyMaint. [En línea] aBESoft Technologies, 2013. [Citado el: 8 de Marzo de 2013.] www.abesoft.com/EasyMaint.

EcuRed. Flujo de Trabajo de Implementación. [En línea] [Citado el: 5 de Abril de 2013.] http://www.ecured.cu/index.php/Flujo_de_Trabajo_de_Implementaci%C3%B3n.

Encinosa, Lázaro J. Blanco. La informática en la dirección de empresas. La Habana : Félix Varela, 2011. pág. 1. ISBN 978-959-07-1629-4.

Eugene Levner, Yael Perlman, T.C.E. Cheng, Ilya Levner. "A network approach to modeling the multi-echelon spare-part inventory system with backorders and interval-valued demand". [En línea] [Citado el: 22 de Noviembre de 2012.] <http://www.sciencedirect.com/science>.

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

García, Ing. Ruddy Gómez. Diseño de un sistema de gestión de ciencia e innovación tecnológica empresarial. monografias.com. [En línea] Julio de 2006. [Citado el: 13 de Noviembre de 2012.] Tesis para la Maestría en Gerencia de la Ciencia y la Innovación. <http://www.monografias.com/trabajos38/sistema-gestion-empresarial/sistema-gestion-empresarial.shtml>.

George E. P. Box, William G. Hunter, J. Stuart Hunter. Statistics for Experimenters An Introduction to Design, Data Analysis, and Model Building. s.l. : Library of Congress Cataloging in Publication Data, 1978.

Gestión Tecnológica.issuu. [En línea] 2012. [Citado el: 20 de Octubre de 2012.] http://issuu.com/harenas/docs/gestion_tecnologica_y_competitividad_-_102025121_-.

Hernández, Dennys. Dirección de Información. SIGICEM: Sistema de gestión para Ingeniería Clínica y Electromedicina. [En línea] 9 de Marzo de 2012. [Citado el: 10 de Junio de 2013.] http://repositorio_institucional.uci.cu/jspui/handle/ident/3990. ISBN 978-959-286-019-3.

Herramientas para trabajar con base de datos. Mixprogramas.com. [En línea] [Citado el: 23 de Noviembre de 2012.] <http://www.mixprogramas.com/db-designer-fork-1-3a/>.

<http://www.bsigroup.com.mx/es-mx/Auditoria-y-Certificacion/Sistemas-de-Gestion/De-un-vistazo/Que-son-los-sistemas-de-gestion/>.

I. Jacobson, G. Booch, J. Rumbaugh. El Proceso Unificado de Desarrollo de Software. Madrid : Pearson Educación ,S.A, 2000. ISBN: 84-7829-036-2.

Ingeniería de requisitos. Wikipedia. La enciclopedia libre. [En línea] 2013 de Marzo de 15. [Citado el: 2013 de Marzo de 16.] http://es.wikipedia.org/wiki/Ingenier%C3%ADa_de_requisitos.

José A. Senra, Elier Broche. SISTEMA DE GESTIÓN TECNOLÓGICA HOSPITALARIA V 1.0. Google Académico. [En línea] 2000. [Citado el: 4 de Noviembre de 2012.] http://scholar.googleusercontent.com/scholar?q=cache:RypmlUVvw_cJ:scholar.google.com/+%22sistemas+de+gestion+tecnologica%22&hl=es&as_sdt=0.

La enciclopedia libre.Wikipedia. [En línea] 31 de Octubre de 2012. [Citado el: 27 de Noviembre de 2012.] <http://es.wikipedia.org/wiki/Framework>.

LAMP.helpdesk. [En línea] 2010. [Citado el: 29 de Noviembre de 2012.] <http://www.helpdesk-software.ws/es/it/17052004.htm>.

Iarman, Craig. UML y Patrones. Mexico : Down Speth White, 2000. ISBN 970-17-0261-1.

Luis Vilcahuamán, Rossana Rivas. Ingeniería Clínica y Gestión de Tecnología en Salud: Avances y Propuestas. Lima : GRAMBS Corporación Gráfica S.A.C, 2006. ISBN 9972-2885-0-1.

McClure, Carma. Introducción a Herramientas CASE y System Architect. MTP. [En línea] 2002. [Citado el: 20 de Noviembre de 2012.]< http://users.dsic.upv.es/asignaturas/eui/mtp/doc-practicas/intro_case_SA.pdf

Ministerio de Protección Social - Colombia. Empresa Social del Estado. Instituto Nacional de Cancerología. [En línea] 2008. [Citado el: 13 de Noviembre de 2012.] <http://www.cancer.gov.co/contenido/contenido.aspx?catID=532&conID=1210>.

Modelos y Sistemas De Gestión Tecnológica. BuenasTareas.com. [En línea] febrero de 2011. [Citado el: 13 de Noviembre de 2012.] <http://www.buenastareas.com/ensayos/Modelos-y-Sistemas-De-Gestion-Tecnologica/1630944.html>.

Mora, Francisco. PRÁCTICAS UML: Lenguaje Unificado de Modelado. [En línea] 2002. [Citado el: 1 de Diciembre de 2012.] <http://www.dccia.ua.es/dccia/inf/asignaturas/GPS/archivos/Uml.PDF>.

Nuchera, Antonio Hidalgo. La gestión de la tecnología como factor estratégico de la competitividad industrial. Madrid : Economía Industrial, 1999. 330 VI.

Ochoa, Migdely. Biblioteca Virtual en Salud. Innovación, tecnología y gestión tecnológica. [En línea] Centro Nacional de Información de Ciencias Médicas-Infomed, 9 de Agosto de 2007. [Citado el: 3 de Abril de 2013.] http://bvs.sld.cu/revistas/aci/vol16_4_07/aci081007.html

Patricia López, Francisco Ruiz. INGENIERÍA DEL SOFTWARE I, Lenguaje Unificado de Modelado - UML. Universidad Cantabria: Facultad de Ciencias, 2011.

Pressman, Roger. Ingeniería de Software. Un enfoque práctico. Madrid : Biblioteca especializada, 2001.

Ranniel Rivero, Velmour Muñoz, Zoila E. Morales, Dennys J. Hernández. SACCEM: SISTEMA AUTOMATIZADO CUBANO PARA EL CONTROL DE EQUIPOS MÉDICOS. Dirección de Información. [En línea] 9 de Marzo de 2012. [Citado el: 15 de Noviembre de 2012.] http://repositorio_institucional.uci.cu/jspui/handle/ident/3978. ISBN 978-959-286-019-3

Real Academia Española. DICCIONARIO DE LA LENGUA ESPAÑOLA - Vigésima segunda edición. [En línea] 2013. [Citado el: 5 de Junio de 2013.] <http://lema.rae.es/drae/?val=gestión>

Ricardo, Ing. Febe Ángel Ciudad. monografias.com. Utilización del patrón Modelo-Vista-Controlador en el diseño de software educativo. [En línea] Profesor Instructor de Ingeniería y Gestión de Software – UCI, Abril de 2006. [Citado el: 3 de Abril de 2013.] <http://www.monografias.com/trabajos43/patron-modelo-vista/patron-modelo-vista2.shtml>.

Rodríguez, Denis E. Ingeniería Clínica. Scribd. [En línea] 2003. [Citado el: 15 de Noviembre de 2012.] <http://es.scribd.com/doc/80047589/Manual-Ing-Clin>.

Ruiz, Lucas M. Palavecino. Scribd. Sistema Informático de Gestión de Departamentos de Ingeniería Clínica. [En línea] 4 de Abril de 2012. [Citado el: 6 de Marzo de 2013.] <http://es.scribd.com/search?query=sistema+inform%C3%A1tico+de+gesti%C3%B3n+de+departamentos+de+ingenier%C3%ADa+cl%C3%ADnica>.

Scribd. Fase de Implementación. [En línea] 3 de Febrero de 2010. [Citado el: 6 de Abril de 2013.] <http://es.scribd.com/doc/27701099/Explicacion-de-la-fase-de-IMPLEMENTACION-de-la-metogologia-RUP>.

Scribd. Fase de Implementación. [En línea] 3 de Febrero de 2010. [Citado el: 6 de Abril de 2013.] <http://es.scribd.com/doc/27701099/Explicacion-de-la-fase-de-IMPLEMENTACION-de-la-metogologia-RUP>.

Sebastián, Juan. Comusoft.com. Modelo Vista Controlador-definición y características. [En línea] 13 de Noviembre de 2010. [Citado el: 4 de Abril de 2013.] <http://www.comusoft.com/modelo-vista-controlador-definicion-y-caracteristicas>.

SIMI. Sistema Integrado de Mantenimiento Industrial. [En línea] República Bolivariana de Venezuela. [Citado el: 18 de Febrero de 2013.] <http://www.simimantenimiento.com/psimi/site.htm>.

Tedeschi, Nicolás. msdn. ¿Qué es un Patrón de Diseño? [En línea] 2013. [Citado el: 1 de Junio de 2013.] <http://msdn.microsoft.com/es-es/library/bb972240.aspx>.

Torossi, Gustavo. Diseño de Sistemas. El Proceso Unificado de Desarrollo de Software. [En línea] [Citado el: 10 de Abril de 2013.] <http://www.chaco.gov.ar/utn/disenodesistemas/apuntes/oo/ApunteRUP.pdf>.

Torres, Manolo. INFORMATION SYSTEMS GROUP TIC-194. [En línea] [Citado el: 3 de Abril de 2013.] <http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf>.

Unified Modeling Language (UML).IBM. [En línea] [Citado el: 9 de Noviembre de 2012.] <http://www-01.ibm.com/software/rational/uml/>.

Visconti, Marcello. Departamento de Informática. Fundamentos de Ingeniería de Software. [En línea] 2013. [Citado el: 2 de Junio de 2013.] www.inf.utfsm.cl/~visconti/ili236/Documentos/08-Patrones.pdf.

Visual Paradingm. Boost Productivity with Innovative and Intuitive Technologies. [En línea] [Citado el: 21 de Noviembre de 2012.] <http://www.visual-paradigm.com>.

XperimentoS.[En línea] 4 de Septiembre de 2007. [Citado el: 28 de Noviembre de 2012.] <http://www.xperimentos.com/2007/09/04/los-mejores-12-frameworks-javascript/>.

ANEXOS

Anexo 1. Diagrama entidad-relación del módulo de Gestión Tecnológica v2.0

