

Universidad de las Ciencias Informáticas

Facultad 7



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

Balance y Planificación de Insumos Médicos

Módulo alas **BAP**Planificación v1.2

Autores: Arianna Suárez Torres

Mario Manuel Pérez Naranjo

Tutores: Ing. Annia Arencibia Morales

Ing. Rotceh Domínguez López

Co-tutora: Ing. Katia Hurtado Duvergel

Ciudad de La Habana, Junio de 2009

“Año del 50 aniversario del triunfo de la Revolución”

Declaración de Autoría

DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmamos la presente a los 29 días del mes de Junio del año 2009.

Arianna Suárez Torres

Mario Manuel Pérez Naranjo

Ing. Annia Arencibia Morales

Ing. Rotceh Domínguez López

Datos de Contacto

Datos de Contacto

Ing. Annia Arencibia Morales (aarencibia@.uci.cu): Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Profesor de la Universidad de las Ciencias Informáticas, pertenece al Departamento de Práctica Profesional, posee la categoría docente de profesor Instructor. Ha impartido la Asignatura de Investigación de Operaciones, Probabilidad y Estadística; actualmente imparte la asignatura de Segundo Perfil. Se desempeña como líder del Proyecto Balance Material.

Ing. Rotceh Dominguez López (rdominguez@.uci.cu): Graduado de Ingeniería en Ciencias Informáticas en la Universidad de las Ciencias Informáticas. Profesor de la Universidad de las Ciencias Informáticas, pertenece al Departamento de Práctica Profesional, profesor adiestrado recién graduado. Imparte la asignatura de Segundo Perfil. Se desempeña como Arquitecto del AT-SAS y pertenece al grupo de Negocio del AT-SAS.

No conozco a nadie que haya llegado a la cima sin trabajo duro. Esa es la receta. No siempre te llevará a la cima, pero te dejará bastante cerca.

Margaret Thatcher.

Agradecimientos

De Arianna:

Le agradezco a toda mi familia por haberme formado incondicionalmente, convirtiéndome en la persona que soy, especialmente: a mi mamá que ha confiado en mí cada día, me ha apoyado siempre y me ha dado fuerzas para seguir adelante. A mis padres por su preocupación y comprensión. A mis abuelas que son tan maravillosas. A Tivi por un ser mi ejemplo profesional. A mis hermanos y primos, por quienes me tracé la meta de darles un buen ejemplo. A mis vecinos que han estado al tanto de mis pasos y a mi Plutt por brindarme su apoyo y su cariño.

Además, a mis compañeros de todos estos años de mi carrera con los que he compartido alegrías y tristezas, tanto en las aulas como en el apartamento. A todos los profesores que me han brindado conocimientos y apoyo. A mis tutores y co-tutora que han sido guías durante el desarrollo de este trabajo y muchas veces quienes coloreaban el documento. A mi compañero de tesis que me brindó mucho apoyo y fue protagonista de la realización de este trabajo, sin olvidar que discutíamos cuando no estábamos de acuerdo.

A la Revolución y Fidel Castro que han permitido que me convierta en una profesional.

De Mario:

Para la realización de esta tesis fue de gran ayuda contar con el apoyo de muchas personas a las cuales quisiera agradecer. Primeramente a mis padres, hermana y abuelos por ser mis ejemplos ante la vida. Me han brindado confianza y amor incondicional, proporcionándome fuerzas para seguir adelante ante buenas y malas circunstancias. Agradecer a Pompa, Dannier, Bolmey y Luis Ángel, por su gran ayuda para realizar este trabajo. A todo aquel que ha aportado su granito de arena cuando lo he necesitado, a Jorge del Toro, Walfrido, René y Frank David. A mis tutores por todo el trabajo que les ocasioné y en especial a Katia, quien aceptó guiarnos como co-tutora de esta investigación.

Agradecer a mis amigos: Yusi, Aramis, Mary, Juan, Nay, Under, Dana, Daylén, Humberto, Yay, Yunieski y a todos los demás con los cuales compartí buenos y malos momentos a lo largo de la carrera. A Mercedes Urrutia, Ariel Garcés, Américo Eugenio y a Sarah Inés por sus consejos, cariño y amistad. A la flaca más alegre que he conocido, la cual siempre me ha dado ánimos para trabajar, inspirándome confianza y elevando mi autoestima en los momentos difíciles, con la que he tenido el privilegio de compartir este trabajo, mi súper CT Arianna Suárez.

Dedicatoria

Dedico este trabajo a todos aquellos que han contribuido a mi educación y me han brindado su apoyo. A quienes me han llenado de alegría en momentos de tristeza, en especial, a mi mamita por todo su amor.

Arianna Suárez Torres.

Existen personas que buscan una razón para vivir, yo he tenido suerte, la mía me estaba esperando al llegar a este mundo, mi familia. Me siento orgulloso y complacido de contar con tanta gente que me quiere, y sobre todo, que se quieren entre ellos. A toda mi querida y sencilla familia que se preocupa por mí, en el seno de la cual he crecido rodeado de amor, compañía y comprensión va dedicado este trabajo de diploma.

Mario Manuel Pérez Naranjo.

Resumen

Resumen

Actualmente la planificación y el balance de insumos médicos, en las entidades de salud del país se realizan de forma manual. La información que se maneja para llevar a cabo este proceso está expuesta a diversos problemas tales como: la duplicidad, pérdida o la introducción de errores humanos. Además, para realizar la planificación no se tienen en cuenta las opiniones de los especialistas, quienes realmente conocen las actividades planificadas para cierto período de tiempo. Este trabajo tiene como objetivo desarrollar una aplicación web para la planificación de materiales médicos que esté en correspondencia con las nuevas necesidades de planificación en las unidades de salud.

Para el desarrollo del mismo, se utiliza el Enterprise Architect 7.0 como herramienta de modelado y UML 2.0 como lenguaje. Para el diseño de las páginas se emplea el Dreamweaver 8, el cual soporta los CCS, HTML y JavaScript. Para el desarrollo con PHP 5.2 y el Framework Symfony 1.2 se usa el ZendStudioForEclipse 6.0 y para la administración de la Base de Datos el EMS Manager for PostgreSQL 4.1.

El Módulo de Planificación agilizará y facilitará el proceso de planificación de materiales médicos en las unidades de salud. Permitirá que la misma se lleve a cabo estableciendo diferentes períodos para planificar, según las necesidades de la unidad de salud y posibilitará crear la demanda de los materiales teniendo en cuenta su existencia en el almacén.

Palabras Claves: *Planificación, demanda, unidad de salud, insumos médicos.*

Tabla de Contenidos

TABLA DE CONTENIDOS

INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	4
1.1 Sistema Nacional de Salud	4
1.2 Sistemas automatizados vinculados al campo de acción	5
1.3 Técnicas, tecnologías, metodologías y software usados para la solución del problema ..	10
1.3.1 Internet.....	10
1.3.2 Aplicaciones Web.....	11
1.3.3 Servidor Web.....	12
1.3.4 Servidor de web Apache 2.2.....	12
1.3.5 Arquitectura.....	12
1.3.6 Modelo Vista Controlador (MVC).....	13
1.3.7 Cliente Servidor.....	13
1.3.8 Javascript 1.1.....	15
1.3.9 Ajax.....	15
1.3.10 HTML.....	16
1.3.11 XML.....	16
1.3.12 CSS 2.....	16
1.3.13 PHP 5.2.....	17
1.3.14 PostgreSQL 8.3.....	18
1.3.15 Framework Yahoo User Interface (YUI) 2.5.....	19
1.3.16 Symfony 1.2.....	19
1.3.17 UML 2.0.....	21
1.3.18 Proceso Unificado Racional (RUP).....	22
1.3.19 Enterprise Architect 7.0.....	23
1.4 Herramientas	24
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA	25
2.1 Objeto de Automatización	25
2.2 Modelo de Dominio	26
2.2.1 Conceptos Fundamentales.....	27
2.2.2 Diagrama del Modelo de Dominio.....	29
2.3 Propuesta del sistema	30

Tabla de Contenidos

2.4 Especificación de Requisitos Software	30
2.4.1 Dependencias y relaciones con sistemas externos	30
2.4.2 Requerimientos Funcionales	31
2.4.3 Requerimientos No Funcionales.....	34
2.5 Modelo del sistema	37
2.5.1 Definición de los actores	37
2.5.1.1 Vista global de los actores.....	39
2.5.2 Diagrama de Casos de Uso del Sistema	39
2.5.3. Descripción textual de Casos de Uso del Sistema.....	41
CAPÍTULO 3: DISEÑO DEL SISTEMA	45
3.1. Modelo de diseño	45
3.1.1. Estructura del diseño.....	45
3.1.2. Descripción de elementos del diseño	46
3.1.3. Diagramas de Clases del Diseño.....	47
3.1.4. Diagramas de Secuencia	52
3.1.5. Descripción de las clases	57
3.1.5.1. Descripción de páginas Clientes.....	57
3.1.5.2. Descripción de páginas Servidoras	59
3.2. Modelo de Datos	59
3.2.1. Descripción de las tablas de la Base de Datos	61
CAPÍTULO 4: IMPLEMENTACIÓN	66
4.1 Modelo de Implementación	66
4.1.1. Diagrama de Componentes.....	66
4.1.2. Diagrama de Despliegue	75
CONCLUSIONES	77
RECOMENDACIONES	78
REFERENCIAS BIBLIOGRÁFICAS	79
BIBLIOGRAFÍA	82
ANEXOS	87

Introducción

INTRODUCCIÓN

Con el proceso de informatización de la sociedad cubana se han destinado múltiples recursos a los diferentes sectores del país. Uno de los que ha sido priorizado es el Sistema Nacional de Salud (SNS). El proceso de informatización en esta rama se define por el conjunto de métodos, técnicas, procedimientos y actividades gerenciales dirigidas al manejo de la información en salud, la cual comprende el estado de salud de la población, la información sobre el conocimiento de las ciencias de la salud y la información en general para la toma de decisiones clínico-epidemiológicas, operativas y estratégicas.

Dentro del Ministerio de Salud Pública (MINSAP), se encuentra el Departamento de Planificación y Economía (DPE), cuya función fundamental es la realización de los planes económicos anuales y su control, que abarca, entre otros aspectos, la planificación de los materiales gastables de uso médico y las inversiones en el sector de la salud. [1]

Entre todos los procesos realizados por el DPE, uno de los de mayor interés para la presente investigación es el de planificación de materiales médicos, el cual constituye la base fundamental para decidir qué y cuánto comprar, con la finalidad de alcanzar una eficiente asignación presupuestaria para estos medios. De esta forma se logra el uso óptimo de los recursos que se asignan al SNS, a fin de que sus resultados en la salud de la población y el grado de satisfacción de la misma sean los más elevados.

Enfrascadas en el proceso de la informatización de la salud cubana, existen diferentes empresas de software como Softel y la Universidad de las Ciencias Informáticas (UCI).

La UCI está conformada por varias facultades, cada una de ellas trabaja un perfil definido para el desarrollo de software. La facultad 7 es la encargada de desarrollar software para la salud, para ello consta de diferentes Áreas Temáticas entre las que se encuentran: Atención Primaria de la Salud (APS), Atención Secundaria de la Salud (Hospitales), Atención Terciaria de la Salud (Especializado) y existe una de apoyo a las anteriores, Sistema de Apoyo a la Salud (SAS). Perteneciente a esta última, se encuentra el proyecto Balance Material, que desarrolló en el año 2007 la aplicación "Sistema para la planificación de materiales gastables de uso médico (Módulo de Planificación)", cuya principal funcionalidad es desarrollar las planificaciones de materiales gastables en las unidades de salud.

En el año 2008, se necesitó mejorar la versión anterior, al superar algunas de sus deficiencias que se analizan a continuación. Entre estas se encuentran que: en primer lugar, la unidad de salud planificaba especialidades que no brindaba, por tanto, eran datos que no se necesitaban en la aplicación.

Introducción

Además, en segundo lugar, la planificación en las entidades policlínicos se realizaba por especialidades y procederes. En tercer lugar, se gestionaban las unidades de salud, los usuarios que iban a interactuar con el sistema y sus permisos, existiendo otro sistema que brindaba estos servicios y que le podían brindar esa información.

La segunda versión fue el “Sistema de Gestión de Información en el Proceso de Planificación de Materiales Gastables de Uso Médico del Ministerio de Salud Pública (MINSAP)”. En esta se hicieron una serie de mejoras tales como: la planificación en las unidades de salud policlínicos se basa en la nueva estrategia de planificar, de acuerdo a los servicios que se brindan en estas unidades y los módulos de materiales asignados a estos servicios. Se eliminó de las especialidades el nivel de subespecialidades. Se hizo la integración con sistemas externos para obtener la información de las unidades de salud, los usuarios y sus permisos. Se introdujo la funcionalidad de llevar los registros históricos lo cual permitía tener una referencia de años anteriores.

Este sistema superó las expectativas del sistema anterior, sin embargo, aún presenta ciertas deficiencias. Esta versión no brinda la posibilidad de efectuar planificaciones en diferentes períodos de tiempo para que la planificación anual sea más eficaz y exacta; maneja información que no le pertenece a ese negocio como es la gestión de materiales, de servicios y la existencia de los materiales en los almacenes. En el despliegue de este software se encontró que todas las unidades de salud no cuentan con conectividad, lo que dificulta la planificación, pues estaba concebido para que los niveles superiores planificaran una vez que tuvieran, mediante el sistema, la solicitud de todos los niveles inferiores.

Además, este sistema está desarrollado sobre una arquitectura en la cual el proceso de mantenimiento del mismo se torna engorroso, pues un cambio que se desee hacer provoca la reprogramación de gran parte del mismo.

A partir de lo antes expuesto, se identifica como **problema científico**: El sistema existente para la planificación de materiales médicos no cumple con todas las necesidades actuales de las unidades de salud.

Lo que conlleva al **objeto de estudio**: Proceso de planificación de materiales en las unidades de salud. Centrándose en el **campo de acción**: Gestión de la información en el proceso de planificación de materiales en las unidades de salud.

Para dar solución al problema antes mencionado se propone como **objetivo general**: Desarrollar la versión 1.2 para la planificación de materiales médicos que esté en correspondencia con las nuevas necesidades de planificación en las unidades de salud.

Introducción

Para dar cumplimiento al objetivo general se proponen las siguientes tareas de investigación:

1. Analizar cómo se planifica en las unidades de salud y en el sistema anterior.
2. Elaborar toda la documentación según el ciclo de desarrollo, basándose en la metodología RUP.
3. Diseñar el prototipo no funcional de todo el sistema.
4. Construir la capa de presentación con las opciones que brinda el Symfony y la Yahoo User Interface (YUI).
5. Diseñar la base de datos a partir de las nuevas funcionalidades identificadas.
6. Implementar las funcionalidades relacionadas con el proceso de planificación, teniendo en cuenta el uso de las interfaces y el consumo de servicios que brindan los restantes módulos.

El trabajo está estructurado en cuatro capítulos, los cuales se describen a continuación:

En el **capítulo 1 Fundamentación Teórica** se refleja el estado del arte del tema en cuestión tanto a nivel nacional como internacional. Se analizan las tecnologías, metodologías y herramientas utilizadas para la propuesta de solución al problema.

En el **capítulo 2 Características del sistema** se presenta una breve descripción de los elementos del negocio relacionados con el procesamiento de la información que se maneja en la planificación de materiales médicos. Se especifican los principales artefactos que se generan en los flujos de trabajo de Negocio y Requerimientos como son: el modelo de dominio, los requerimientos funcionales y no funcionales, a partir de los cuales se representan los casos de uso del sistema y la descripción de los mismos.

En el **capítulo 3 Diseño del sistema** se realizan los diagramas de clases del diseño y los diagramas de secuencia, además, se describen una serie de tablas para una mejor comprensión del Modelo de Datos.

En el **capítulo 4 Implementación** se muestra la implementación del sistema propuesto a través de la descripción de algunos componentes importantes del sistema. Están reflejados el diagrama de componentes y subsistemas de implementación.

Capítulo 1: Fundamentación Teórica

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

En este capítulo se plantean conceptos relacionados con el Sistema Nacional de Salud y los procesos de balance material que forman parte del dominio del problema a resolver en el presente Trabajo de Diploma. Contiene una investigación de los sistemas informáticos existentes para la planificación de materiales, lo cual es una base para la realización de este trabajo. Además se analizan las tecnologías, metodologías y herramientas a utilizar en el desarrollo de la investigación.

1.1 Sistema Nacional de Salud

El Sistema Nacional de Salud (SNS) cubano depende del Ministerio de Salud Pública de Cuba, que es el organismo encargado de dirigir, ejecutar y controlar la aplicación de la política del Estado referente a la salud pública.

Entre los principios fundamentales de este ministerio se encuentran: el carácter estatal de la salud pública, la accesibilidad y gratuidad de la salud, la unidad de la ciencia, la docencia y la práctica médica, integridad y desarrollo planificado, la participación organizada y activa de la población en las tareas de salud y el internacionalismo y colaboración.[2]

Entre los componentes del Sistema Nacional de Salud se encuentran:

- Aseguramiento de suministros no médicos y mantenimiento del sistema.
- Aseguramiento tecnológico, médico y electromedicina.
- Producción, distribución y comercialización de medicamentos y equipos.[3]

El SNS cubano para cumplir sus metas de elevar la eficiencia sobre la base de un mejoramiento constante de la salud de la población, cuenta con la Dirección de Planificación y Economía (DPE). Esta se encarga de la planificación económica en la salud, por lo que dirige la planificación de las importaciones de materiales de uso médico, las inversiones en el sector de la salud, la planificación de salud que incluye la planificación de los portadores energéticos (electricidad, combustibles), de los alimentos y productos de uso no médico.

La planificación de materiales de uso médico es una actividad fundamental para llevar a cabo el control de la economía en el país, así como para satisfacer la necesidad de garantizar el abastecimiento en las unidades de salud con los materiales suficientes.

La planificación se realiza en todas las unidades de salud pero no de la misma manera. En los hospitales hay que tener en cuenta las especialidades que se practican en él, así como los procedimientos que implican esas especialidades. En los policlínicos, se realiza a partir de los servicios que allí se

Capítulo 1: Fundamentación Teórica

prestan. Para llevar a cabo este proceso, es importante el conocimiento de algunos aspectos que se abordan a continuación:

Procederes médicos: Métodos, procedimientos, técnicas o acciones que se ejecutan sobre un paciente. Están comprendidos dentro una especialidad médica, por ejemplo, dentro de la especialidad médica Cirugía Cardiovascular se realiza el proceder médico Cirugía Aórtica, la especialidad Anatomía Patológica tiene entre sus procederes la Necropsia, la Biopsia por congelación, entre otros. Para cada proceder se establecen índices de consumo de cada recurso o renglón que se utilizan en el mismo. [4]

Índices de consumo: Cantidad de cada tipo materiales de que se utilizan en un proceder específico, por ejemplo, en el proceder Cirugía Aórtica se utilizan 4 (índice de consumo) bisturís, 10(índice de consumo) pinzas, 10(índice de consumo) gasas, entre otros. Los índices de consumo transmiten eficiencia en los cálculos. [5]

Niveles de Actividad: Cantidad de veces que se va a realizar un proceder específico en un período de tiempo determinado, en este caso en cierto período de tiempo. Deben ser planificados a nivel de la institución asistencial, a partir de las regularidades de su comportamiento y sus necesidades de atención médica. [6]

1.2 Sistemas automatizados vinculados al campo de acción

La planificación de materiales, debido a la necesidad de hacer más fácil y efectiva su organización en las diferentes empresas, ha sido objetivo de informatización. A continuación se muestran algunos ejemplos de los sistemas existentes para llevar a cabo el proceso que constituye el campo de acción.

1.2.1 Soluciones Informáticas existentes en el ámbito nacional

Sistema para la planificación de materiales gastables de uso médico. (Módulo de Planificación)

Sistema informático concebido en la Universidad de las Ciencias Informáticas en el año 2007, utilizado para perfeccionar la planificación de los materiales gastables de uso médico en las entidades de salud del país. Permite que la información fluya de una forma más rápida y organizada entre los diferentes niveles de dirección en que está organizado el Sistema Nacional de Salud. Facilita el conocimiento de la cantidad de materiales a utilizar en cada entidad de salud del país. Permite también realizar los cálculos pertinentes que posibilitan conocer el importe de la cantidad de materiales que el MINSAP

Capítulo 1: Fundamentación Teórica

debe garantizar, para el buen funcionamiento del sistema de atención médica en el país. La planificación se realiza teniendo en cuenta una organización que responde a las especialidades y sus procedimientos en cada entidad de salud.

Sistema Automatizado de Balance Material

Aplicación que hoy en día se utiliza en los Centros de Diagnóstico Integral (CDI) en Venezuela. Es un sistema automatizado integral de abastecimiento médico, elaborado por especialistas cubanos del Ministerio de Industria Básica (MINBAS) que laboran en Venezuela.

La aplicación permite realizar las siguientes actividades: captar pedidos de las diferentes instituciones de salud; realizar análisis de los pedidos; conocer los inventarios en cada una de las instituciones, almacenes, estados y a nivel nacional; realizar balances semanales, mensuales, anuales. Así como, conocer los productos en falta o con baja cobertura en cada una de las instituciones de salud; conocer los días de coberturas de cada producto en cada una de las instituciones de salud; llevar registros históricos y realizar facturaciones (despachos).

El sistema incluye: materiales gastables, reactivos, medicamentos y los fondos fijos (material no gastable). La planificación se realiza sobre la base de módulos que tienen confeccionados para cada institución de salud, en correspondencia con sus características. Estos módulos se van modificando y completando teniendo en cuenta los consumos reales que van teniendo, además de utilizar índices de consumo.

El sistema no incluye Estomatología, ni Oftalmología. Internamente, cada CDI lleva el registro detallado de sus niveles de actividad, aunque en sus informes los realizan de forma general: cantidad de consultas, operaciones realizadas, estudios radiológicos realizados, pruebas de laboratorio, etc. En el sistema están incluidos alrededor de 500 productos diferentes, mucho menor que los que se planifican en Cuba. Productos tales como: papel para ultrasonido, papel para electrocardiograma, gasa, medicamentos y reactivos. El sistema es efectivo y actualmente está en un proceso de perfeccionamiento.

Las características presentadas por este sistema son similares a las del que se desea desarrollar, sin embargo no cumple con todas necesidades actuales de las unidades de salud como son:

- No incluye las especialidades de Estomatología ni Oftalmología.
- Está concebido para el control de la planificación de los CDI y no para todas las unidades de salud como hospitales y clínicas estomatológicas.

Capítulo 1: Fundamentación Teórica

Sistema de Gestión de Información en el Proceso de Planificación de Materiales Gastables de Uso Médico del Ministerio de Salud Pública (MINSAP)

Sistema informático concebido en la Universidad de las Ciencias Informáticas en el año 2008, el cual incluye las funcionalidades que necesitó el cliente para realizar la planificación de materiales gastables de uso médico de forma exitosa, teniendo en cuenta las políticas definidas por el MINSAP respecto a la integración con sistemas externos. Este sistema presenta una nueva organización en la planificación de los policlínicos, teniendo en cuenta servicios médicos y módulos de materiales asignados a los mismos. Los planificadores de las provincias y la nación asignarán, a cada entidad, las especialidades y procedimientos médicos que se practiquen en ellas. El sistema facilita registros de planificaciones almacenadas de años anteriores y permite conocer la existencia de materiales en las unidades de salud del país. [7]

Este sistema constituye una visión para el sistema a desarrollar pero no se puede adoptar como solución debido a que:

- Maneja información como: gestión de materiales, gestión de servicios y la existencia de los materiales en los almacenes, las cuales no pertenecen al negocio de planificación.
- Presenta un rígido sistema de planificación debido a que las planificaciones solo se pueden hacer anuales.
- Está desarrollada teniendo en cuenta los cuatro niveles que conforman el Sistema Nacional de Salud en Cuba: Nivel Nacional, Nivel Provincial, Nivel Municipal y Nivel de Entidad, necesitando cada nivel superior de la planificación de los inferiores para realizar su planificación, sin tener en cuenta que muchas unidades de salud no tienen conectividad para enviar las planificaciones haciendo uso de la red a los niveles superiores.

ERP Cubano "CEDRUX: Sistema Integral de Gestión"

El CEDRUX es un Sistema de Planificación de Recursos de una Empresa que constituye un proyecto del país, el cual se está desarrollando en la Universidad de Ciencias Informáticas con el apoyo de varias entidades que hasta el momento llevaban este tema, como Desoft y TEICO Casa del software Villa Clara. Incluye tecnologías como: PHP, PostgreSQL versión 8.2, ZendFramework, Apache HTTP Server versión 2 y otras. [8]

El Módulo de Planificación cuenta con funcionalidades como:

- Gestión de etapas.
- Gestión de versiones de formatos para cada etapa.

Capítulo 1: Fundamentación Teórica

- Gestión de indicadores.
 - Propios de la planificación.
 - De relaciones con otras tablas de la Base de Datos.
- Gestión de formatos. (Puede ser talón, modelo)
- Gestión de contenidos. (Datos entrados en los formatos)
- Cálculo de necesidades materiales.
- Gestión de notificaciones.
- Gestión de modificaciones.
- Gestión de indicadores físicos no asociados en este momento a ningún subsistema.
- Interoperabilidad con otros sistemas.

Este sistema hasta el momento no constituye una solución al problema, pues aún se encuentra en desarrollo, por tanto el cliente no conoce si este será capaz de solucionar sus necesidades. Además, el tipo de planificación que este propone realizar es económica, la misma consiste en planificar las necesidades de productos libremente, sin tener en cuenta un presupuesto asignado por el cual regirse, no siendo así en el MINSAP, donde se realiza una planificación de tipo financiera. Esta se basa en un presupuesto asignado a partir del cual se realizará la planificación tratando de suplir todas las necesidades sin superar el mismo.

1.2.2 Soluciones Informáticas existentes en el ámbito internacional

Sistemas ERP

Los Sistemas Enterprise Resource Planning o de Planificación de Recursos de la Empresa (ERP) son sistemas de gestión de información. Estos integran y automatizan muchas de las prácticas de negocio asociadas con los aspectos operativos o productivos de una empresa.

Se caracterizan por estar compuestos por diferentes partes integradas en una única aplicación. Estas partes son de diferente uso, por ejemplo: producción, ventas, compras, logística, contabilidad (de varios tipos), gestión de proyectos, GIS (sistema de información geográfica), inventarios y control de almacenes, pedidos, nóminas, entre otras. Sólo se puede definir un ERP como la integración de todas estas partes.

Los objetivos principales de los sistemas ERP son:

Capítulo 1: Fundamentación Teórica

- Optimización de los procesos empresariales.
- Acceso a toda la información de forma confiable, precisa y oportuna (integridad de datos).
- La posibilidad de compartir información entre todos los componentes de la organización.
- Eliminación de datos y operaciones innecesarias (o redundantes).
- Reducción de tiempos y de los costes de los procesos (mediante procesos de reingeniería).

[9]

Existen ejemplos de ERP libres y sin embargo usan tecnologías que pueden ser riesgosas en la situación actual de bloqueo que sufre el país; por ejemplo OpenBravo y OpenXpertya que están basados en la plataforma de Java: J2EE, cuya máquina virtual es propiedad de SUN que es una empresa norteamericana; aunque haya comenzado a liberar el código de esta máquina virtual, sigue bajo las leyes de su gobierno que bloquea por todos los medios el acceso a tecnología informática. Como segunda desventaja se señala que estos sistemas han sido diseñados para empresas capitalistas que tienen un modelo de gestión y de procesos, muy diferente a las entidades cubanas, donde la economía es centralizada y operan otros mecanismos. No quiere decir que estos ERP libres sean totalmente inadecuados, pero hacerle las adaptaciones necesarias sería tan engorroso que es preferible diseñar uno específico para satisfacer las necesidades nacionales. [10]

Material Requirement Planning (MRP)

El Sistema Material Requirement Planning o Planificador de las Necesidades de Material (MRP), es un sistema de planificación de materiales y gestión de existencias que responde a las preguntas de, cuánto y cuándo aprovisionarse de materiales. La utilización del sistema MRP conlleva una forma de planificar la producción caracterizada por la anticipación, tratándose de establecer qué se quiere hacer en el futuro y con qué materiales se cuenta, o en su caso, se necesitarán para poder realizar todas las tareas de producción. [11]

Sistema SAP

El sistema Sistemas, Aplicaciones y Productos (SAP), está compuesto por una serie de áreas funcionales o módulos que responden de forma completa y en tiempo real a los procesos operativos de las compañías. Las academias de SAP orientadas a los consultores autónomos de negocios de las grandes empresas son: Módulo Contabilidad Financiera, Módulo Control, Módulo Gestión de Materiales, Módulo de Ventas y Distribución y Módulo Planificación de Producción. [12]

Capítulo 1: Fundamentación Teórica

El Módulo Gestión de materiales (MM), está completamente integrado a otras áreas funcionales de SAP y brinda soporte a todas las fases de gestión de materiales: planificación de necesidades y control, compras, entrada de mercaderías, gestión de existencias y verificación de facturas.

Los componentes más importantes son:

- Planificación de las necesidades sobre consumo: La función principal es la de supervisar existencias y crear automáticamente propuestas de pedidos para el departamento de compras y fabricación.
- Compras: Las tareas incluyen aprovisionamiento externo de materiales y servicios, determinación de posibles fuentes para provisión, supervisión de entregas y pago a proveedores. [13]

En el caso de estos sistemas tratados en el ámbito internacional, no constituyen una solución para el proceso de planificación de materiales médicos en las unidades de salud, debido a que están concebidos para gestionar materiales independientemente del tipo que sea, es decir, no se centran en la gestión de materiales médicos. Además, Cuba está sometida al bloqueo, que crea grandes limitaciones económicas, por lo que el sistema a desarrollar debe permitir una planificación variable según los materiales que se pueden obtener, y ajustándose a las necesidades de las unidades de salud. Ello, debido a que estos sistemas son muy costosos y necesitan de soporte.

1.3 Técnicas, tecnologías, metodologías y software usados para la solución del problema.

Las tecnologías a utilizar para la construcción del Sistema de Planificación son aquellas que han sido definidas por el Área Temática Sistema de Apoyo a la Salud (SAS), las cuales en su mayoría son libres. A continuación se muestran aspectos respecto a las mismas.

1.3.1 Internet

Internet es una enorme red que conecta redes y computadoras distribuidas por todo el mundo, permitiendo la comunicación, transmisión y búsqueda de información sin grandes recursos tecnológicos ni económicos. Internet surge a razón de la necesidad del Ministerio de Defensa del Gobierno de los Estados Unidos de mantener sus sistemas de cómputos distantes conectados en una red llamada ARPANET. La primera razón de su desarrollo fueron motivos militares, años más tarde, el proyecto sirvió para conectar científicos que fueron desarrollando y ampliando las potencialidades de la

Capítulo 1: Fundamentación Teórica

misma, llevando su desarrollo al punto de que en el año 1972 fueran conectadas todas las agencias y proyectos del departamento de defensa y alrededor de 50 universidades.

Eventualmente la Fundación Nacional de Ciencia, entidad gubernamental de los Estados Unidos, se hizo cargo de la red, conectando las redes que luego darían lugar a lo que hoy, se conoce como Internet. [14]

Con el nacimiento de Internet, nacen las denominadas aplicaciones Web, las cuales dieron un impulso definitivo a la red y constituyen una forma clara y visual de presentar información. Una página Web es un documento que contiene información sobre un tema específico y que es almacenado en un servidor remoto conectado a la red mundial de información, más conocida como Internet. Este documento podrá ser consultado por cualquier persona que se pueda conectar a esta red. [15]

1.3.2 Aplicaciones Web

Las aplicaciones Web son sistemas informáticos, usados por los usuarios para acceder a un servidor Web, a través de Internet o Intranet, y obtener así la información que se encuentra almacenada en él. Las aplicaciones Web son muy utilizadas hoy en día debido a las ventajas que las mismas ofrecen. Algunas de estas se ven reflejadas a continuación:

- **Compatibilidad multiplataforma.** Tienen un camino mucho más sencillo para la compatibilidad con múltiples plataformas que las aplicaciones de escritorio. Varias tecnologías incluyendo Java, Flash, ASP y Ajax permiten un desarrollo efectivo de programas soportando todos los sistemas operativos principales.
- **Actualización.** Están siempre actualizadas con la última versión, sin necesidad de llamar la atención del usuario o interferir con sus hábitos de trabajo, con la posibilidad de que va a iniciar nuevas descargas y procedimientos de instalación.
- **Inmediatez de acceso.** No necesitan ser descargadas, instaladas y configuradas.
- **Menos Bugs (errores).** Con aplicaciones basadas en Web, todos los usuarios utilizan la misma versión y los errores pueden ser corregidos tan pronto como son descubiertos.
- **Múltiples usuarios concurrentes.** Las aplicaciones Web pueden ser utilizadas por múltiples usuarios al mismo tiempo. [16]

Para que la información de las Aplicaciones Web viaje de forma segura por la red, existen medios que permiten la encriptación u ocultamiento de la información de forma que no exista ningún tercero que sea capaz de obtenerla y utilizarla con fines lesivos que perjudiquen a los involucrados. Este medio se llama Protocolos de Comunicación.

Capítulo 1: Fundamentación Teórica

1.3.3 Servidor Web

Es un programa que se ejecuta de forma continua en un servidor, manteniéndose a la espera de peticiones hechas por parte de un cliente (un navegador de Internet), luego contesta estas peticiones devolviendo el contenido que el cliente solicita. Además utiliza el protocolo Hypertext Transfer Protocol (HTTP) el cual está diseñado para transferir los hipertextos, páginas web o páginas Hypertext Markup Language (HTML): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música. Actualmente a nivel mundial existen disímiles servidores web, pero el más utilizado, por su estabilidad y su rendimiento, es el Servidor Web Apache. [17]

1.3.4 Servidor web Apache 2.2

Comenzó a desarrollarse en 2005 pero hoy en día es el servidor Web más utilizado del mundo, encontrándose por encima de todos sus competidores, tanto gratuitos como comerciales. Representa el complemento perfecto para el desarrollo de páginas dinámicas con PHP, pues comparte con este muchas de sus características, como gratuidad, popularidad, su sencillez de manejo y versatilidad, ya que puede ser instalado sobre Linux o Windows [18].

Es considerado el servidor web por excelencia pues desde su surgimiento ha demostrado que es estable y que tiene mejor rendimiento que sus competidores. Además brinda algunas ventajas que son fundamentales para el desarrollo Web como son:

- **Fiabilidad:** Más del 90% de los servidores con más alta disponibilidad funcionan bajo un servidor Apache.
- **Software libre:** El servidor Apache es totalmente gratuito y es distribuido bajo la licencia de Apache en la cual se permite realizarle cambios al código fuente.
- **Extensibilidad:** Se pueden añadir módulos para ampliar aún más las capacidades del servidor.

[19]

1.3.5 Arquitectura

La arquitectura, según Roger S. Pressman, es la estructura de las estructuras del sistema, la cual comprende los componentes de software, las propiedades de esos componentes visiblemente externos y las relaciones entre ellos. [20] Esta no es más que la organización de los componentes del sistema de forma que quede registrado la manera en la que colaboran y se relacionan entre ellos. Es una vía en la cual el sistema queda modelado desde distintas perspectivas con el objetivo de lograr y establecer como deberá ser construido el futuro sistema.

Capítulo 1: Fundamentación Teórica

1.3.6 Modelo Vista Controlador (MVC)

Este patrón de arquitectura de software permite separar los datos de una aplicación, la interfaz de usuario y la lógica de negocio en tres componentes distintos, proporcionando múltiples vistas sobre un mismo modelo de datos. El patrón MVC se usa frecuentemente en aplicaciones Web donde se utilicen diferentes interfaces de usuario y el código que provee los datos a la página es dinámico. Los tres elementos esenciales de este patrón son los siguientes: [21]

- **Modelo:** Administra el comportamiento y los datos del dominio de la aplicación, responde a requerimientos de información sobre su estado, usualmente formulados desde la vista, respondiendo a instrucciones de cambio para cambiar el estado de estos datos, habitualmente desde el controlador.
- **Vista:** Este presenta el modelo en un formato adecuado para interactuar, usualmente un elemento de interfaz de usuario.
- **Controlador:** Este responde a eventos, usualmente acciones del usuario e invoca cambios en el modelo y probablemente en la vista. (**Ver Anexo 1**)

Tanto la vista como el controlador dependen del modelo, el cual no depende de otros conceptos o clases. Esta separación permite construir y probar el modelo independientemente de la representación visual. En aplicaciones Web por ejemplo, la separación entre la vista (navegador) y el controlador (componentes del lado del servidor que manejan los requerimientos a través de HTTP) está muy claramente definida.

Entre las ventajas del estilo señaladas por Microsoft están las siguientes:

- **Soporte de vistas múltiples:** Dado que la vista se encuentra separada del modelo y no hay dependencia directa del modelo con respecto a la vista, la interfaz de usuario puede mostrar múltiples vistas de los datos de manera simultánea.
- **Adaptación al cambio:** Los requerimientos no funcionales de interfaz de usuario tienden a cambiar con mayor rapidez que las reglas del negocio. Los clientes pueden preferir distintas opciones de representación pero dado que el modelo no depende de la vista, agregar nuevas opciones o modificar las ya existentes generalmente no afecta al modelo.

1.3.7 Cliente Servidor

Este tipo de arquitectura es un modelo que las aplicaciones clientes utilizan para comunicarse con el servidor. Donde un servidor es una aplicación que ofrece servicios a usuarios en Internet, el servidor recibe una solicitud de un servicio por parte de un cliente y devuelve los resultados en forma de una respuesta a la solicitud realizada. (**Ver Anexo 2**)

Capítulo 1: Fundamentación Teórica

Los usuarios son los que inician la invocación del servicio en la parte cliente de la aplicación en la cual se construye la solicitud del servicio y es enviada al servidor utilizando el protocolo TCP/IP como medio de transporte. [22]

Con la implantación de este modelo cada usuario, llamado cliente, tiene la posibilidad de trabajar desde cualquier lugar solamente haciendo una solicitud al servidor y manipulando la información que este le devuelva en forma de respuesta, en dependencia de las necesidades del cliente.

Bajo esta forma de comunicación, la información se puede mantener de forma centralizada, evitando así la pérdida de la misma por problemas de hardware u otros motivos relacionados con situaciones técnicas que puedan presentarse con los clientes. Este tipo de modelo es aplicable bajo cualquier circunstancia, el mismo puede implantarse tanto en Internet como en una red local (Intranet); además el procedimiento no es muy costoso ya que solo se requiere una PC y un navegador web como cliente de la aplicación. Igualmente cualquier tipo de mantenimiento en la aplicación sería transparente para el usuario, lo que posibilitaría reducir aún más, los costos en su implantación.

Características que presenta la arquitectura Cliente Servidor:

- Las funciones de Cliente y Servidor pueden estar en plataformas separadas, o en la misma plataforma.
- Un servidor brinda servicios a múltiples clientes en forma concurrente.
- Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.
- Las funciones cliente-servidor pueden ser dinámicas. Ejemplo, un servidor puede convertirse en cliente cuando realiza la solicitud de servicios a otras plataformas dentro de la red.
- El servidor presenta a todos sus clientes una interfaz única y bien definida.
- El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo. [23]

Ventajas de la arquitectura Cliente Servidor:

- Permite un mejor aprovechamiento de los sistemas existentes, protegiendo la inversión. Por ejemplo, la compartición de servidores y dispositivos periféricos (como impresoras) entre máquinas clientes.

Capítulo 1: Fundamentación Teórica

- Se pueden utilizar componentes, tanto de hardware como de software de distintos fabricantes, lo cual contribuye considerablemente a la reducción de costos y favorece la flexibilidad en la implantación y actualización de soluciones.
- Proporcionan un mejor acceso a los datos. La interfaz de usuario ofrece una forma homogénea de ver el sistema, independientemente de los cambios o actualizaciones que se produzcan en él y de la ubicación de la información.
- La arquitectura Cliente Servidor elimina la necesidad de mover grandes bloques de información por la red hacia los ordenadores personales o estaciones de trabajo para su proceso. Los servidores controlan los datos, procesan peticiones y después transfieren sólo los datos requeridos a la máquina cliente. Finalmente la máquina cliente presenta los datos al usuario mediante interfaces amigables. Todo esto reduce el tráfico de la red, lo que facilita que pueda soportar un mayor número de usuarios.
- En una arquitectura como esta, los clientes y los servidores son independientes los unos de los otros, con lo que pueden renovarse para aumentar sus funciones y capacidad de forma independiente, sin afectar al resto del sistema. [24]

1.3.8 Javascript 1.1

Es un lenguaje de programación que proporciona dinamismo a las páginas web, facilitando la aparición y desaparición del texto, animación, y acciones que se activan al oprimir en botones y ventanas que aparecen con mensajes. [25] Es un lenguaje interpretado, por lo que no es necesario compilar el código para ejecutarlo, es decir, es posible probarlos en cualquier navegador sin necesidad de procesos intermedios. Está diseñado para controlar la apariencia y eventos de los objetos de un formulario, siendo soportado en múltiples navegadores, compartiendo elementos con otros lenguajes de alto nivel como son el C, Java o PHP, no solo en su sintaxis sino en su formato. [26]

1.3.9 Ajax

El término Ajax significa “Asynchronous JavaScript + XML” o lo que es lo mismo “JavaScript Asíncrono + XML”. En sí, no es una tecnología, sino la combinación de varias tecnologías que se desarrollan de forma autónoma y se unen de forma sorprendente. Las tecnologías que conforman Ajax son XHTML + CSS para crear una presentación basada en estándares, DOM para la creación y manipulación dinámica de la presentación. Además XML, XSLT y JSON para el intercambio y la manipulación de la información que será mostrada en la presentación, XMLHttpRequest, para el intercambio asíncrono de información y por último JavaScript para combinar todas las tecnologías anteriores. Surge con motivo

Capítulo 1: Fundamentación Teórica

de mejorar la interacción con las aplicaciones web tradicionales, en el aspecto en que, en el modelo tradicional, por cada acción en la presentación desencadena una petición al servidor, por lo cual una vez procesada dicha llamada, el servidor vuelve a construir una nueva página cliente al navegador del usuario.

Las aplicaciones desarrolladas usando Ajax posibilitan la reducción de la recarga constante de las páginas con la utilización de un objeto intermedio entre el usuario y el servidor. Ello permite una mejora en la respuesta de la aplicación, y que el usuario nunca se encuentre frente a una ventana del navegador vacía esperando una respuesta tardía del servidor. [27]

1.3.10 HTML

HTML es el acrónimo de Hypertext Markup Language (Lenguaje de Marcado de Hipertexto), creado en 1989 por Tim Berners-Lee. Es originalmente un subconjunto de SGML (Standard Generalized Markup Language), especializado en la descripción de documentos en pantalla. El proyecto inicial se basaba en una colección de etiquetas que permitían describir documentos de texto y vínculos de hipertexto que hacían posible el desplazamiento en forma jerárquica entre diferentes documentos.

La facilidad de su uso y la particularidad de no ser propiedad de nadie, hizo de HTML el sistema idóneo para compartir información a través de Internet. Inicialmente su intención era que las etiquetas fueran capaces de marcar la información de acuerdo a su significado, pero por diversos motivos los creadores de los navegadores Web fueron añadiendo más etiquetas HTML, dirigidas a controlar la representación de la información contenida en el documento. [28]

1.3.11 XML

El eXtensible Markup Language (XML) es el lenguaje de marcado extensible establecido en febrero 1998, no es más que un metalenguaje de marcado descriptivo, que posee un grupo de reglas que le permite crear sus propios elementos de marcado, los cuales pueden usarse después para describir su contenido. XML se desarrolló porque HTML no estaba diseñado para describir algunos tipos de datos que las personas querían enviar a través de la red, pues este último tenía sus etiquetas predefinidas. XML proporciona la flexibilidad y consistencia que no se podía alcanzar con HTML, pues no se tiene que forzar al contenido para que se ajuste al grupo limitado de elementos proporcionados por este. [29]

1.3.12 CSS 2

Cascading Style Sheets (Hojas de Estilo en Cascada), es un mecanismo que describe cómo se va a mostrar un documento en la web, o incluso cómo se va a imprimir. Este mecanismo ofrece a los

Capítulo 1: Fundamentación Teórica

desarrolladores el control sobre el estilo y el formato de sus documentos. Se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la presentación. Estos estilos permiten a los desarrolladores Web controlar el formato de múltiples páginas Web al mismo tiempo, posibilitando ante cualquier cambio en el estilo marcado para un elemento en la CSS, que sean afectadas todas las páginas vinculadas a dicha CSS en las que aparezca ese elemento. [30]

1.3.13 PHP 5.2

PHP (Hypertext Pre-processor) es un lenguaje de programación interpretado, creado en 1994 por Rasmus Lerdof. Es utilizado habitualmente para la creación de sitios, contenido dinámico para aplicaciones Web y aplicaciones para servidores. Con frecuencia los scripts PHP se embeben en otros códigos como HTML ampliando las posibilidades del diseñador de páginas Web. La interpretación y ejecución de estos scripts se hacen en el servidor, el cliente (solicitud realizada desde un navegador Web) sólo recibe el resultado y jamás ve el código PHP.

Permite conexión con todo tipo de bases de datos como MySQL, PostgreSQL, Oracle, DB2, Microsoft SQL Server, Firebird y SQLite. PHP se puede ejecutar sobre siete plataformas, funciona en once tipos de servidores, ofrece soporte para varios Sistemas Gestores de Bases de Datos (SGBD) y contiene unas 40 extensiones estables, actualmente se encuentra en la versión 5.

Algunas de las más importantes capacidades de PHP son: [31]

- Integración con varias bibliotecas externas, permitiendo generar documentos Portable Document Format (PDF) y Microsoft Office Excel (XLS).
- Ofrece una solución simple y universal para las paginaciones dinámicas de fácil programación.
- Soportado por una gran comunidad de desarrolladores, como producto de código abierto, permitiendo que los fallos de funcionamiento se encuentren y reparen rápidamente, implicando menos costos.
- Gran número de funciones predefinidas. A diferencia de otros lenguajes de programación, PHP fue diseñado especialmente para el desarrollo de páginas Web dinámicas. Por ello, está dotado de un gran número de funciones que simplificará enormemente tareas habituales como descargar documentos, envío de correo electrónico, creación dinámica de imágenes y gráficos en el servidor, procesamiento de información en formularios,

Capítulo 1: Fundamentación Teórica

manipulación de cookies y sesiones, transporte de información mediante HTTP y análisis de documentos XML.

- Análisis léxico para reconocer el tipo de dato almacenado en una variable, se ejecuta automáticamente, permitiéndole al usuario no tener que separar las variables de sus valores.
- Posee un conjunto de funciones de seguridad que previenen la inserción de órdenes dentro de una solicitud de datos desde el cliente evitando por ejemplo, la ocurrencia de la conocida inyección de código SQL.

Debe tenerse en cuenta que este potente lenguaje de programación también posee algunas desventajas entre ellas se pueden mencionar que todo el trabajo se realiza en el lado del servidor, no delegando responsabilidades al cliente, puede que en un momento determinado la capacidad de respuesta sea ineficiente en la medida en que las solicitudes al servidor aumenten considerablemente. Además si el código PHP se incluye en código HTML, puede que la legibilidad de este se vea afectada. La Programación Orientada a Objetos (POO) en PHP es aún deficiente para aplicaciones de gran envergadura influyendo sobre el rendimiento de estos sistemas.

1.3.14 PostgreSQL 8.3

El Gestor de Base de Datos PostgreSQL 8.3 es un sistema de base de datos muy potente de código abierto. Soporta gran parte del estándar SQL y, en algunos aspectos, está diseñado para que sea extensible por los usuarios. Se caracteriza por posibilitar transacciones ACID, claves foráneas, vistas, secuencias, subpeticiones, lanzadores, tipos y funciones definidas por el usuario, control de concurrencia multiversión. También posee interfaces gráficas de usuario y enlazadores para algunos lenguajes de programación. [32]

Principales características:

- Bloqueos consultivos: permiten el control de objetos de bases de datos a nivel de aplicación usando el motor rápido de bloqueos de PostgreSQL.
- Sentencias preparadas: tiene nuevas interfaces administrativas y mejoras de rendimiento en sentencias preparadas.
- Criptografía: el módulo pgcrypto, soporta criptografía dentro de la base de datos, el cual fue actualizado con los mejores algoritmos de criptografía.[33]

Capítulo 1: Fundamentación Teórica

1.3.15 Framework Yahoo User Interface (YUI) 2.5

El framework o librería YUI (Yahoo User Interface) es un conjunto de utilidades y controles escritos en JavaScript que se utilizan para crear aplicaciones web dinámicas complejas. Además, la librería YUI incluye varias utilidades relacionadas con CSS, por lo que también se considera un framework CSS. La empresa Yahoo distribuye gratuitamente la librería YUI en forma de software libre y bajo licencia BSD, que permite utilizar la librería para proyectos de cualquier tipo, incluso comercial. Lo mejor de YUI es que cuenta con el respaldo de Yahoo, que utiliza la librería en sus páginas que son consultadas a diario. Además, YUI cuenta con una gran documentación que incluye cientos de ejemplos de uso. [34]

Principales Ventajas:

- Una documentación muy buena con unas páginas bien trabajadas.
- El respaldo de una gran empresa que soporta el proyecto.
- Las utilidades de YUI simplifican el desarrollo para la compatibilidad entre navegadores basados en técnicas DOM, DHTML y AJAX.
- Los controles de YUI proporcionan elementos visuales altamente interactivos del diseño para sus aplicaciones Web. Estos elementos se crean y se manejan íntegramente del lado del cliente y nunca requieren de una recarga de página.[35]
- Tiene la hoja de estilo Reset.css que elimina los estilos predeterminado de los navegadores web (Internet Explorer, Mozilla Firefox, Safari, Opera, entre muchos otros) para la presentación de un sitio Web.
- Todo el desarrollo de YUI, es de código abierto. Permitiendo que cualquier desarrollador web del mundo lo pueda usar libremente.
- La hoja de estilos Reset.css se adapta a casi cualquier proyecto web competente.
- La empresa Yahoo tiene gran experiencia en el desarrollo web, por lo que representan una fuente confiable. Además cuentan con un amplio equipo de profesionales dedicados a investigar periódicamente todos los temas antes mencionados.[36]

1.3.16 Symfony 1.2

Symfony es un completo framework diseñado para optimizar el desarrollo de las aplicaciones web. 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

Capítulo 1: Fundamentación Teórica

compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación.

Symfony está desarrollado completamente con PHP 5. Ha sido probado en numerosos proyectos reales y se utiliza en sitios web de comercio electrónico de primer nivel. Symfony es compatible con la mayoría de gestores de bases de datos, como MySQL, PostgreSQL, Oracle y SQL Server de Microsoft. Se puede ejecutar tanto en plataformas *nix (Unix, Linux, etc.) como en plataformas Windows.

Principales Ventajas:

- Fácil de instalar y configurar en la mayoría de plataformas (y con la garantía de que funciona correctamente en los sistemas Windows y *nix estándares).
- Independiente del sistema gestor de bases de datos.
- Sencillo de usar en la mayoría de casos, pero lo suficientemente flexible como para adaptarse a los casos más complejos.
- Basado en la premisa de "convenir en vez de configurar", en la que el desarrollador solo debe configurar aquello que no es convencional.
- Sigue la mayoría de mejores prácticas y patrones de diseño para la web.
- Preparado para aplicaciones empresariales y adaptables a las políticas y arquitecturas propias de cada empresa, además de ser lo suficientemente estable como para desarrollar aplicaciones a largo plazo.
- Código fácil de leer que incluye comentarios de phpDocumentor y que permite un mantenimiento muy sencillo.
- Fácil de extender, lo que permite su integración con librerías desarrolladas por terceros.
- La capa de internacionalización que incluye Symfony permite la traducción de los datos y de la interfaz, así como la adaptación local de los contenidos.
- La capa de presentación utiliza plantillas y layouts que pueden ser creados por diseñadores HTML sin ningún tipo de conocimiento del framework. Los helpers incluidos permiten minimizar el código utilizado en la presentación, ya que encapsulan grandes bloques de código en llamadas simples a funciones.
- Los formularios incluyen validación automatizada y relleno automático de datos ("repopulation"), lo que asegura la obtención de datos correctos y mejora la experiencia de usuario.
- Los datos incluyen mecanismos de escape que permiten una mejor protección contra los ataques producidos por datos corruptos.

Capítulo 1: Fundamentación Teórica

- La gestión de la caché reduce el ancho de banda utilizado y la carga del servidor.
- La autenticación y la gestión de credenciales simplifican la creación de secciones restringidas y la gestión de la seguridad de usuario.
- El sistema de enrutamiento y las URL limpias permiten considerar a las direcciones de las páginas como parte de la interfaz, además de estar optimizadas para los buscadores.
- El soporte de e-mail incluido y la gestión de APIs permiten a las aplicaciones web interactuar más allá de los navegadores.
- Los listados son más fáciles de utilizar debido a la paginación automatizada, el filtrado y la ordenación de datos.
- Los plugins, las factorías (patrón de diseño "Factory") y los "mixin" permiten realizar extensiones a medida de Symfony.
- Las interacciones con Ajax son muy fáciles de implementar mediante los helpers que permiten encapsular los efectos JavaScript compatibles con todos los navegadores en una única línea de código. [37]

1.3.17 UML 2.0

El Lenguaje de Modelado Unificado (UML - Unified Modeling Language) es un lenguaje gráfico para visualizar, especificar y documentar cada una de las partes que comprende el desarrollo de software. UML entrega una forma de modelar cosas conceptuales como lo son procesos de negocio y funciones de sistema, además de actividades concretas como lo son escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables.

En las versiones previas del UML, se hacía un fuerte hincapié en que este no era un lenguaje de programación. Un modelo creado mediante UML no podía ejecutarse. En el UML 2.0, esta asunción cambió de manera drástica y se modificó el lenguaje, de manera tal que permitiera capturar mucho más comportamientos. De esta forma, se permitió la creación de herramientas que soporten la automatización y generación de código ejecutable, a partir de modelos UML.

Ventajas de UML 2.0:

- Produce un aumento en la calidad del desarrollo.
- Reduce los costos del proyecto.
- Mejora en un 50% o más los tiempos totales de desarrollo.

Capítulo 1: Fundamentación Teórica

- Permite especificar la estructura y el comportamiento del sistema y comunicarlo a todos los integrantes del proyecto.
- Brinda la posibilidad de obtener un "plano" del sistema.
- Permite dimensionar mejor los riesgos de un proyecto, tener un mejor rendimiento antes de construir el sistema.
- Facilita la documentación de las decisiones de la arquitectura del proyecto.
- Ofrece un mejor soporte a la planificación y control del proyecto.
- Ofrece mayor rigurosidad en la especificación.
- Permite realizar una verificación y validación del modelo realizado.
- Se pueden automatizar determinados procesos y permite generar código a partir de los modelos y viceversa.[38]

1.3.18 Proceso Unificado Racional (RUP)

Un proceso de desarrollo de software es el conjunto de actividades necesarias para transformar los requisitos de un usuario en un software. Sin embargo, es más que un simple proceso; es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de software, para diferentes áreas de aplicación, tipos de organizaciones, niveles de aptitud y tamaños de proyecto. El Proceso Unificado está basado en componentes, lo cual quiere decir que el sistema o software en construcción está formado por componentes interconectados a través de interfaces.

RUP posee tres características fundamentales que marcan el desarrollo y adaptación de la metodología en un proyecto:

- Centrado en la arquitectura, lo que posibilita que se defina la estructura fundamental del sistema y que la misma nunca se pierda.
- Guiado por los casos de uso, brindando la característica de que el desarrollo se vea marcado por una serie de construcciones organizadas por los casos de uso, posibilitando una mejor organización y avance en el desarrollo.
- Iterativo e incremental, posibilitando establecer las construcciones del sistema por partes y que a medida que sean terminadas dichas partes se vayan obteniendo incrementos en el desarrollo del software.

RUP divide el proceso en cuatro fases, y a su vez divide estas en iteraciones a través de las cuales se van obteniendo productos que posibilitan incrementos en el software. Las fases definidas son: Fase de

Capítulo 1: Fundamentación Teórica

Inicio, Fase de Elaboración, Fase de Construcción y Fase de Transición. (**Ver Anexo 3**) Brinda una serie de flujos de trabajos que son realizados durante las fases anteriormente mencionadas y que garantizan el orden en que deben ser ejecutadas todas las tareas que establece esta metodología, además de lograr que la realización de la aplicación sea de alta calidad, que resuelva las necesidades del usuario dentro de un cronograma y que el producto sea obtenido con el menor costo posible. RUP utiliza el UML (Lenguaje Unificado de Modelado) para representar todos los diagramas en la concepción y diseño del Sistema de Planificación a desarrollar.

Es muy útil para eliminar carga de trabajo al servidor, siendo idóneo para las validaciones de datos y acciones que son posibles hacer del lado del cliente, con el objetivo de agilizar el proceso que puede desencadenar el servidor como respuesta a una petición del cliente. [39]

1.3.19 Enterprise Architect 7.0

Es la herramienta más potente y flexible para la plataforma de Windows. Una herramienta de análisis de negocio y UML orientada a objetos para el desarrollo completo del ciclo de vida del software. Enterprise Architect provee el límite competitivo para el desarrollo de software, administración de proyecto, administración de requerimientos y análisis de negocio. Entre sus características más importantes figuran:

- Última especificación UML 2.1.
- Importación/Exportación XMI 2.1.
- Nuevo motor de Reporte HTML.
- Transformaciones MDA.
- Perfiles y soporte de tecnologías.
- Pruebas, rastreo de recursos, mantenimiento.

Otras de sus características son:

- Modelado Basado en el Equipo.
 - Archivos compartibles o modelos basados del repositorio.
 - Control de versiones con cualquier herramienta SCC.
 - Seguridad incorporada y administración de permisos.
- Modelado visual por excelencia.

Capítulo 1: Fundamentación Teórica

- Ingeniería reversa de código fuente en 10 lenguajes.
- Importar esquema de base de datos.
- Importar fuente XSD y WSDL.
- Importar binarios .NET y Java.
- Integración con Visual Studio y Eclipse.[40]

1.4 Herramientas

Basándose en los lineamientos arquitectónicos del Área Temática SAS, para dar solución al problema planteado se seleccionan a utilizar el Enterprise Architect 7.0 como herramienta de modelado, la cual utiliza el UML 2.0 como lenguaje y posibilita crear los diferentes diagramas. Para el diseño de las páginas se utilizará el Dreamweaver 8, el cual soporta los CCS, HTML y JavaScript.

Para el desarrollo con PHP 5.2 y el Framework Symfony 1.2, se utilizará el ZendStudioForEclipse 6.0 y para la administración de la Base de Datos el EMS Manager for PostgreSQL 4.1.

Conclusiones

En este capítulo se analizaron varios sistemas que se han desarrollado tanto en Cuba como en el resto del mundo, y que fueron concebidos para la gestión de materiales. Sin embargo, ninguno se ajusta a las necesidades, debido a las peculiaridades que presentan las unidades de salud y el Sistema Nacional de Salud cubano, además de las limitaciones económicas con las que cuenta país.

Además, se realizó un estudio acerca de las tecnologías y herramientas que han sido definidas por el Área Temática SAS, arribando a que las mismas son idóneas a utilizar, debido a estudios que se han realizado por el grupo de arquitectura. La mayoría de estos software son libres, los cuales constituyen una salida para los países en subdesarrollo y pequeñas empresas que no tendrían que someterse a las limitaciones impuestas por los software propietarios.

Capítulo 2: Características del Sistema

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

En este capítulo se definen los procesos a automatizar y los conceptos asociados al dominio del sistema que se han obtenido luego de una investigación, centrada en el objeto de estudio. Se describen las características del sistema a desarrollar y se enumeran los requerimientos funcionales y no funcionales. Se identifican los Casos de Uso del Sistema a partir de los requerimientos funcionales.

2.1 Objeto de Automatización

Actualmente en las unidades de salud se cuenta con un departamento de planificación encargado de elaborar la planificación y obtener la demanda de cada una de ellas, este proceso ocurre de forma diferente según el tipo de unidad de salud que sea y se realiza en diferentes períodos de tiempo para obtener una planificación más eficiente.

En el caso de los hospitales el planificador consulta a los especialistas encargados de los diferentes departamentos de la unidad de salud para obtener las planificaciones que estos proponen. Las mismas se realizan de forma aproximada, basándose en la experiencia de períodos anteriores. Una vez que el planificador obtiene y revisa todas las planificaciones, contacta al almacenero para conocer la existencia de materiales en el almacén y a partir de ahí, procede a crear la demanda.

En el caso de los policlínicos, se asignan módulos fijos de materiales según los servicios que se prestan en él. En cada período de planificación, el planificador verifica la cantidad de estos materiales que han sido utilizados, luego se dirige al almacén con el objetivo de conocer la existencia de los materiales, y según esta, establecer la demanda. Con este proceso, se realiza nuevamente el completamiento de los materiales que tiene asignado cada servicio del policlínico. De esta forma, en dependencia de la unidad de salud, se realiza la planificación por especialidades y procederes (en hospitales) o por servicios y módulos de materiales (en policlínicos).

➤ Planificación por especialidades y procederes

En los hospitales se practican especialidades médicas, las cuales presentan procederes médicos. Un proceder tiene asociada la cantidad de cada material que se necesita para realizar ese proceder, lo cual se refleja como índice de consumo (IC).

Para realizar la planificación en un hospital, es necesario que el especialista determine la cantidad de veces que se va a realizar cada proceder médico en el período de tiempo para el que se esté planificando, lo cual constituiría el nivel de actividad (NA). Una vez que se conozca el nivel de actividad para cada proceder y el índice de consumo de cada material que se utiliza en ese proceder, se puede conocer la cantidad de cada material (CM) que se utilizará para cada proceder en ese período de

Capítulo 2: Características del Sistema

tiempo. Por tanto, para cada proceder se puede calcular la cantidad de materiales que utiliza, de la siguiente manera: $NA * IC = CM$.

Para determinar la cantidad total de cada material (CTM) que se va a utilizar en la unidad de salud durante ese período de tiempo, es necesario sumar la cantidad de materiales por tipo de todos los procederes. A esa cantidad total de materiales, se le restaría la Existencia (E), que no es más que la cantidad de ese material, con que se cuenta en el almacén. Por tanto, para obtener la necesidad real de un tipo de material (CRM) es necesario calcularlo de la siguiente manera: $CTM - E = NRM$.

Realizando todas estas operaciones para cada material de cada proceder es que se obtiene la demanda de materiales en la unidad de salud.

- Planificación por servicios y módulo de materiales.

En los policlínicos se prestan servicios, que tienen un módulo de materiales asignado. Por tanto, es preciso conocer los servicios que presta ese policlínico y el módulo de materiales que tiene asignado. Para obtener la cantidad de cada material utilizado por estos servicios y la necesidad real de cada material, sólo restaría consultar la existencia de esos materiales en los almacenes y calcular de la siguiente manera:

$CM - E = NRM$.

2.2 Modelo de Dominio

Un Modelo del Dominio captura los tipos más importantes de objetos que existen o los eventos que suceden en el entorno donde estará el sistema.

En el desarrollo de este trabajo se van a hacer mejoras a un sistema ya existente, donde el negocio se mantiene. El sistema anterior está concebido para los cuatro niveles de Dirección de Salud, el propuesto se adaptará a las necesidades de una unidad de salud, por lo que se realiza un Modelo de Dominio donde se documentan los conceptos ya existentes en el negocio y se incorporan los que surgen en la propuesta del nuevo sistema, como son: el período de tiempo y el administrador. Entre las mejoras que se van a realizar, se encuentran:

- Brindar la posibilidad de que el período de tiempo para la planificación se defina según las necesidades de la unidad de salud.
- Eliminar el manejo de información no perteneciente al negocio de planificación. Este sistema no será el encargado de gestionar materiales, gestionar servicios y la existencia de los materiales en los almacenes, sino que obtendrá esa información de otros módulos como el Módulo Nomenclador y el Módulo de Almacén.

Capítulo 2: Características del Sistema

- Las planificaciones podrán realizarse en un ambiente desconectado, por lo que no será necesaria la conectividad permanente para llevar a cabo el proceso de planificación.

Por lo antes expuesto, es más efectivo realizar una descripción que represente los principales conceptos del sistema en cuestión.

2.2.1 Conceptos Fundamentales

A continuación se muestran las definiciones de los conceptos fundamentales para la comprensión del Diagrama de Dominio.

Administrador: Usuario del sistema encargado de todas las funciones administrativas en el mismo.

Especialista: Usuario del sistema encargado de realizar la planificación de materiales para su departamento.

Planificador: Usuario del sistema encargado de realizar la planificación de la Unidad de Salud, a partir de las planificaciones de todos los Departamentos.

Período de tiempo: Intervalo de tiempo establecido para la realización de la planificación.

Planificación: Estimación de todos los materiales de uso médico que se consideran necesarios durante cierto período de tiempo, con el objetivo de dar cumplimiento a los servicios médicos brindados a los pacientes.

Estado: Condiciones en que puede encontrarse una planificación en dependencia de su realización o no en el período de tiempo establecido, las cuales pueden ser: planificada, no planificada, modificada, cancelada, aceptada, atrasada o rechazada.

Especialidad médica: Conformada por un grupo de procedimientos médicos. Ejemplo de Especialidades: Cardiología, Cirugía, Estomatología, Dermatología, entre muchas otras.

Procederes médicos: Diferentes métodos, procedimientos, técnicas o acciones que se ejecutan sobre un paciente.

Materiales: Materiales médicos que se utilizan para atender a los pacientes. Estos materiales pueden ser gasas, algodones, entre otros.

Servicios: Servicios que se brindan en los policlínicos para atender a los pacientes. Estos servicios pueden ser: Estomatología, Consulta Externa, Fisioterapia y Rehabilitación, Urgencia, Medios Diagnósticos tales como Rayos X, Drenaje biliar, Endoscopia, Ultrasonido entre otros.

Nivel de Actividad: Cantidad de veces que se va a realizar un proceder específico en un período de tiempo determinado.

Demanda: Pedido de materiales que se realiza según la necesidad que se tenga del mismo.

Capítulo 2: Características del Sistema

Demanda Departamento: Pedido de materiales que se realiza según la necesidad que se tenga del mismo en el departamento.

Demanda Unidad de Salud: Pedido de materiales que se realiza según la necesidad que se tenga del mismo en todos los departamentos de una Unidad de Salud.

Reportes: Información que se puede obtener a partir de las planificaciones.

Unidad de salud: Entidad médica, ya sea hospital o policlínico.

Capítulo 2: Características del Sistema

2.2.2 Diagrama del Modelo de Dominio

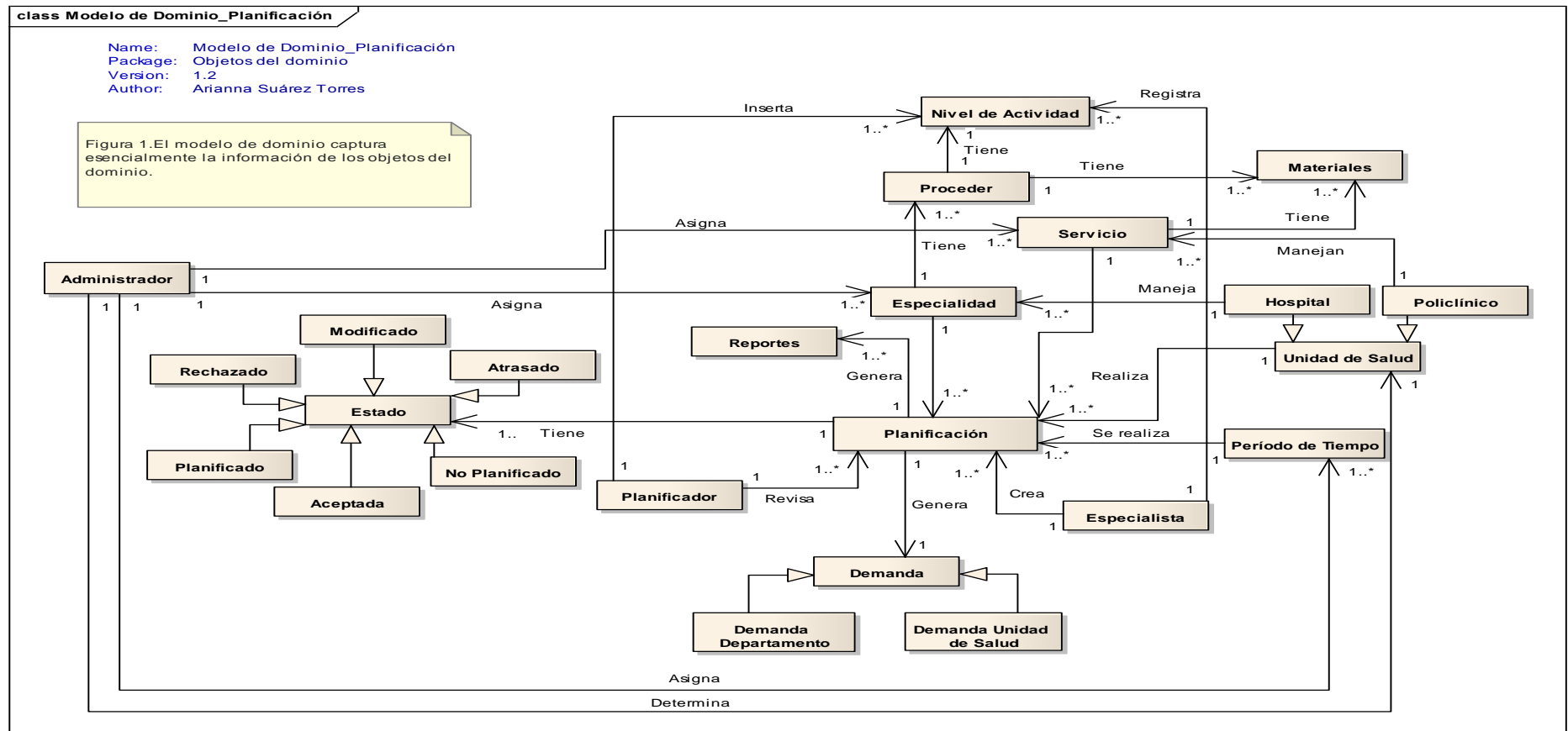


Figura 1. Diagrama del Modelo de Dominio.

Capítulo 2: Características del Sistema

2.3 Propuesta del sistema

Sobre la base del sistema anterior se propone una nueva versión para la planificación de materiales médicos, la cual cumpla con todas las necesidades actuales de las unidades de salud, por tanto este debe reunir las siguientes funcionalidades:

- Asignar especialidades a las unidades de salud.
- Definir períodos de tiempo para la planificación.
- Sistema de avisos por varios criterios.
- Registrar nivel de actividad.
- Revisar planificación.
- Crear demanda de unidad de salud.
- Conectarse con los módulos que le brindan información para realizar sus procesos.

Además de brindar reportes por:

- Especialidades médicas y su información.
- Materiales.
- Demanda.

Todas estas funcionalidades tienen como objetivo perfeccionar la Gestión de la información en el proceso de planificación de materiales en las unidades de salud.

2.4 Especificación de Requisitos Software

En este acápite se especificará el comportamiento del sistema que se va a desarrollar a partir de los requisitos funcionales y no funcionales, además se expondrán las dependencias que tiene el mismo con sistemas externos mostrando cómo ocurre el intercambio de información entre ellos.

2.4.1 Dependencias y relaciones con sistemas externos

Este módulo se integrará con otros módulos que le brinden información necesaria durante el proceso de planificación de materiales. Del Módulo Nomenclador se obtendrá información común y poco variable en el tiempo como los datos de: materiales, servicios, especialidades, estados de planificación y períodos de tiempo, sin necesidad de gestionarlos en el sistema. Del Módulo de Almacén se obtendrá la existencia de los materiales en los almacenes, lo cual posibilitará, a partir de la planificación de la unidad de salud, crear la demanda basándose en la cantidad real que existe de cada material. La accesibilidad al sistema será manejada desde el Componente de Seguridad desarrollado en el Área Temática SAS.

Capítulo 2: Características del Sistema

2.4.2 Requerimientos Funcionales

A continuación se enumeran las capacidades que debe cumplir el Módulo de Planificación, las cuales fueron identificadas a partir de las relaciones que existen entre los conceptos que se enmarcan en el dominio del sistema a resolver.

- RF1. Abrir planificación
- RF2. Actualizar estado de especialidades médicas
- RF3. Actualizar estado de procedimientos médicos
- RF4. Actualizar estado de servicios médicos
- RF5. Actualizar fecha de fin de los días para planificar
- RF6. Actualizar fecha de fin del período
- RF7. Actualizar listado de materiales médicos
- RF8. Actualizar listado de especialidades médicas
- RF9. Actualizar listado de procedimientos médicos
- RF10. Actualizar listado de servicios médicos
- RF11. Asignar especialidades médicas
- RF12. Asignar servicios médicos
- RF13. Autenticar
- RF14. Buscar demanda por especialidades médicas
- RF15. Buscar demanda por materiales médicos
- RF16. Buscar demanda por unidad de salud
- RF17. Buscar especialidades médicas
- RF18. Buscar materiales médicos
- RF19. Buscar planificaciones históricas
- RF20. Buscar planificaciones por estados
- RF21. Buscar procedimientos médicos
- RF22. Buscar servicios médicos
- RF23. Cancelar apertura de planificación
- RF24. Cancelar configuración de tipo de unidad de salud
- RF25. Cancelar especialidades médicas
- RF26. Cancelar período de planificación
- RF27. Cancelar servicios médicos
- RF28. Sistema de avisos
- RF29. Configurar tipo de unidad de salud

Capítulo 2: Características del Sistema

- RF30. Crear período de planificación
- RF31. Crear planificación por especialidad médica
- RF32. Crear planificación por servicios médicos
- RF33. Crear reporte de demanda por especialidades
- RF34. Crear reporte de demanda por materiales
- RF35. Crear reporte de demanda por procedimientos
- RF36. Crear reporte de demanda por servicios
- RF37. Crear reporte de demanda por unidad de salud
- RF38. Crear reporte de especialidades atrasadas
- RF39. Crear reporte de planificación histórica por materiales
- RF40. Crear reporte de planificación histórica por especialidades
- RF41. Crear reporte de planificación histórica por procedimientos
- RF42. Crear reporte de planificación histórica por servicios
- RF43. Crear reporte de planificación histórica por unidad de salud
- RF44. Desasignar especialidades médicas
- RF45. Desasignar procedimientos médicos
- RF46. Desasignar servicios médicos
- RF47. Enviar planificación
- RF48. Imprimir reporte de demanda por especialidades
- RF49. Imprimir reporte de demanda por materiales
- RF50. Imprimir reporte de demanda por procedimientos
- RF51. Imprimir reporte de demanda por servicios
- RF52. Imprimir reporte de demanda por unidad de salud
- RF53. Imprimir reporte de especialidades atrasadas
- RF54. Imprimir reporte de planificación histórica por especialidades
- RF55. Imprimir reporte de planificación histórica por materiales
- RF56. Imprimir reporte de planificación histórica por procedimientos
- RF57. Imprimir reporte de planificación histórica por servicios
- RF58. Imprimir reporte de planificación histórica por unidad de salud
- RF59. Insertar Nivel de actividad
- RF60. Insertar valor a recursos similares
- RF61. Listar especialidades médicas
- RF62. Listar especialidades médicas asignadas
- RF63. Listar estados de planificación

Capítulo 2: Características del Sistema

- RF64. Listar materiales médicos
- RF65. Listar período de planificación
- RF66. Listar procederes médicos
- RF67. Listar servicios médicos
- RF68. Listar servicios médicos asignados
- RF69. Modificar Nivel de actividad
- RF70. Modificar estado planificaciones enviadas
- RF71. Modificar período de planificación
- RF72. Mostrar estado de la planificación
- RF73. Seleccionar especialidades médicas
- RF74. Seleccionar estado
- RF75. Seleccionar material médico
- RF76. Seleccionar período de planificación
- RF77. Seleccionar planificación
- RF78. Seleccionar proceder médico
- RF79. Seleccionar servicios médicos
- RF80. Ver detalles de las especialidades
- RF81. Ver detalles de las planificaciones
- RF82. Ver detalles de los materiales
- RF83. Ver detalles de los procederes
- RF84. Ver detalles de los servicios médicos
- RF85. Configurar sistema de avisos
- RF86. Establecer similares
- RF87. Consultar existencia
- RF88. Buscar Avisos
- RF89. Crear avisos
- RF90. Eliminar avisos
- RF91. Modificar avisos
- RF92. Cancelar avisos
- RF93. Actualizar avisos
- RF94. Crear módulo de materiales
- RF95. Modificar módulo de materiales
- RF96. Listar módulo de materiales
- RF97. Cancelar módulo de materiales

Capítulo 2: Características del Sistema

RF98. Actualizar módulo de materiales

RF99. Buscar módulo de materiales

RF100. Seleccionar módulo de materiales

2.4.3 Requerimientos No Funcionales

Cuando se desarrolla un software este debe cumplir con un grupo de cualidades o propiedades, conocidas como requisitos no funcionales. A continuación se exponen los requisitos propios del Módulo de Planificación.

Usabilidad

Estos requerimientos describen los niveles apropiados de usabilidad, dados los usuarios finales del producto. Para ello debe revisarse la especificación de los perfiles de usuarios.

RNF1 El sistema solo podrá ser utilizado por los especialistas de cada departamento, el administrador y el planificador de la unidad de salud.

RNF2 El sistema debe presentar un acceso fácil y rápido, para facilitar el uso del mismo por usuarios con conocimientos informáticos básicos, tales como: manejo de la computadora, y específicamente de un navegador web.

RNF3 Las funcionalidades estarán organizadas en la parte superior del sistema, según las probabilidades de uso de cada una de ellas, de manera que proporcione una mayor organización de las mismas y permitiendo que sean más fáciles de encontrar por el usuario.

Seguridad

➤ Confidencialidad

RNF4 La información estará protegida contra accesos no autorizados utilizando mecanismos de autenticación y autorización, lográndose con la utilización de servicios web que brinda el Componente de Seguridad del AT.

RNF5 La autenticación será la primera acción del usuario en el sistema y consistirá en suministrar un nombre de usuario único y una contraseña que debe ser de conocimiento exclusivo de la persona que se autentica. Si el usuario autenticado no se encuentra registrado se debe reportar un error de acceso.

➤ Integridad

RNF6 La información podrá ser modificada solo por personal autorizado.

RNF7 Se realiza la validación de los formularios para evitar que se introduzcan datos incorrectos.

RNF8 Se harán validaciones de la información tanto en el cliente como en el servidor contra ataques de inyección HTML o SQL.

➤ Disponibilidad

Capítulo 2: Características del Sistema

RNF9 A los usuarios autorizados se les garantizará el acceso a la información en el momento que el mismo lo requiera.

Eficiencia

RNF10 El sistema deberá ser rápido ante las solicitudes de los planificadores y en el procesamiento de la información, el tiempo de respuesta deberá ser el menor posible.

Soporte

RNF11 Una vez terminado el sistema se realizarán procesos de despliegue, capacitación y mantenimiento de software.

RNF12 Para agilizar el tiempo de soporte se deben utilizar estándares de codificación permitiendo una asimilación rápida del código fuente.

RNF13 Se deberá realizar una documentación detallada de cada clase implementada en el código fuente.

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

Describen los requisitos que provee a los usuarios y clientes de una documentación para la manipulación del sistema, así como para entender el mismo.

RNF14 Se dispondrá de un Manual de Usuario que indicará cómo interactuar con las funcionalidades del sistema.

Interfaz

➤ **Interfaces de usuario**

RNF15 El sistema debe presentar una estructura y diseño de interfaz homogénea permitiendo manejo rápido y confiable de la información.

RNF16 Se deben utilizar tonalidades de colores claros y agradables a la vista, lo cual permite una mayor aceptación por parte del usuario final.

RNF17 El sistema emitirá diferentes mensajes, algunos informativos, otros de advertencia, los cuales servirán para informar al usuario acerca de la realización satisfactoria o no de determinadas acciones, y en otros casos, para indicar que se no se ha introducido algún dato obligatorio o que se introdujo de forma incorrecta.

RNF18 Todos los mensajes de error o de advertencia del sistema deben mostrarse de forma similar permitiendo esto una mayor organización y visualización de la información.

RNF19 La navegabilidad del sistema debe ser simple, permitiendo un manejo rápido y sencillo de la información a gestionar en la misma.

➤ **Interfaces Hardware común**

RNF20 Ordenador Pentium IV.

RNF21 Se requiere tarjeta de red.

Capítulo 2: Características del Sistema

➤ Interfaces Hardware para el cliente

RNF22 Se requiere que tenga al menos:

- 512 MB de memoria RAM
- 10 GB de disco duro
- CPU 3.00GHz

➤ Hardware para los servidores

• Aplicación:

RNF23 Se requiere que tenga al menos:

- 1 GB de memoria RAM
- 40 GB de disco duro como mínimo
- CPU 3.00GHz

• Base de Datos:

RNF24 Se requiere que tenga al menos:

- 1 GB de memoria RAM
- 80 GB de disco duro como mínimo
- CPU 3.00GHz

Interfaces Software

➤ Interfaces Software para el cliente

RNF25 Se requiere que tenga al menos:

- En las computadoras de los clientes se requiere del navegador Web Mozilla Firefox en versión 3.0 o superior.
- Sistema operativo Windows XP Service Pack 2 o GNU/Linux distribución Ubuntu 8.04.

➤ Interfaces Software para los servidores

Aplicación:

RNF26 Se requiere que tenga al menos:

- Servidor web Apache versión 2.2 o superior.
- PHP 5.2 y el Framework Symfony 1.2.
- Sistema operativo GNU/Linux en su distribución Debian 4 Etch preferentemente o Ubuntu 8.04.

Base de Datos:

RNF27 Se requiere que tenga al menos:

- Postgre SQL versión 8.3

Capítulo 2: Características del Sistema

- Sistema operativo instalado GNU/Linux en su distribución Debian 4 Etch preferentemente o Ubuntu 8.04.

Estándares aplicables

RNF28 Para la implementación del sistema se deberán seguir los estándares de codificación y diseño definidos por el AT.

2.5 Modelo del sistema

El Modelo de Casos de Uso del Sistema es un artefacto de Ingeniería de Software que se construye para comunicar ideas a otros, comprender mejor el sistema que se está desarrollando y tener una visión del sistema en su totalidad. Describe, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario, permitiendo que se establezca un acuerdo entre clientes y desarrolladores sobre las condiciones y requerimientos que el sistema debe cumplir. Este modelo está formado por actores, casos de uso y relaciones entre estos, lo cual representa gráficamente los procesos y su interacción con los actores.

2.5.1 Definición de los actores

Los actores del sistema son terceros que no forman parte del sistema sino que interactúan con él, los cuales pueden ser personas o sistemas externos. Los actores que interactúan con el Módulo de Planificación son los siguientes:

Actor	Descripción
Usuario	Persona o individuo que interactúa con el sistema. Constituye una generalización de los roles que se desempeñan dentro de la unidad de salud, ya sea Administrador, Especialista o Planificador, el cual se conoce a partir de la autenticación de dicho Usuario.
Administrador	Es el rol que representa al usuario encargado de realizar las funciones administrativas como: determinar el tipo de unidad de salud y abrir períodos de planificación.
Especialista	Es el rol que representa al usuario encargado de realizar las planificaciones de su departamento, con el objetivo de ser más específico en las necesidades

Capítulo 2: Características del Sistema

	reales de materiales en su especialidad.
Planificador	Es el rol que representa al usuario encargado de revisar las planificaciones realizadas por los especialistas, las cuales puede aceptar o rechazar. Además crea la demanda una vez que todas las planificaciones de los diferentes departamentos de la unidad de salud sean aprobadas. Es el encargado de asignar especialidades o servicios en dependencia del tipo de unidad de salud a la que pertenece.
Visualizador	Es el rol que surge a partir de funcionalidades que tienen en común el Especialista y el Planificador de buscar planificaciones históricas y la demanda de materiales.
Componente de Seguridad	Sistema externo encargado de la seguridad de la aplicación el cual gestiona la seguridad de los productos desarrollados por el AT, basado en el modelo de Autenticación, Autorización y Auditoría.
Módulo Nomenclador	Es el módulo externo del cual se obtendrá información común y poco variable en el tiempo como los datos de: materiales, servicios, especialidades, estados de planificación y períodos de tiempo, sin necesidad de gestionarlos en el sistema.
Módulo de Almacén	Es un módulo externo del cual se obtendrá la existencia de los materiales en los almacenes, lo cual posibilitará, a partir de la planificación de la unidad de salud, crear la demanda basándose en la cantidad real que existe de cada material.

Tabla 1. Descripción de los actores del sistema.

Capítulo 2: Características del Sistema

2.5.1.1 Vista global de los actores

En el siguiente diagrama se puede observar que como usuario del sistema se pueden encontrar: el Administrador, el Especialista y el Planificador, donde los dos últimos pueden actuar a su vez como Visualizador. Además, se representan los tres sistemas externos que interactúan con el Módulo de Planificación, los cuales están representados por el Componente de Seguridad, el Módulo Nomenclador y el Módulo de Almacén.

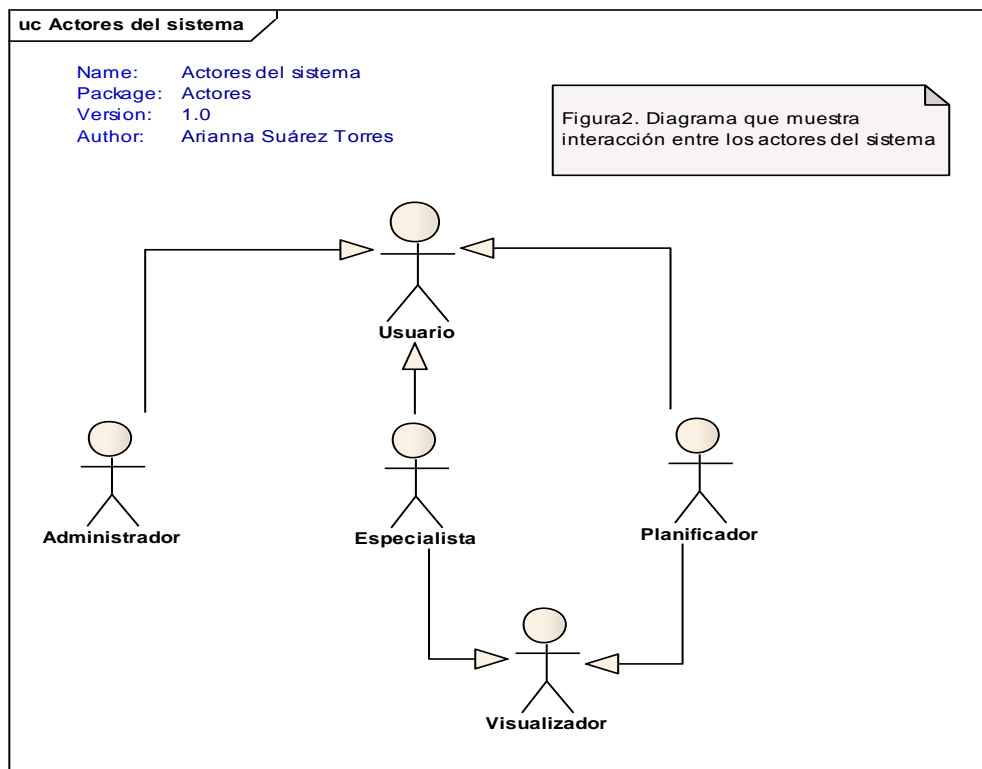


Figura 2. Vista global de los actores.

2.5.2 Diagrama de Casos de Uso del Sistema

A continuación se muestra el diagrama de casos de uso del sistema que muestra los distintos requisitos funcionales que se esperan de la aplicación y cómo se relaciona con su entorno (usuarios u otras aplicaciones).

Capítulo 2: Características del Sistema

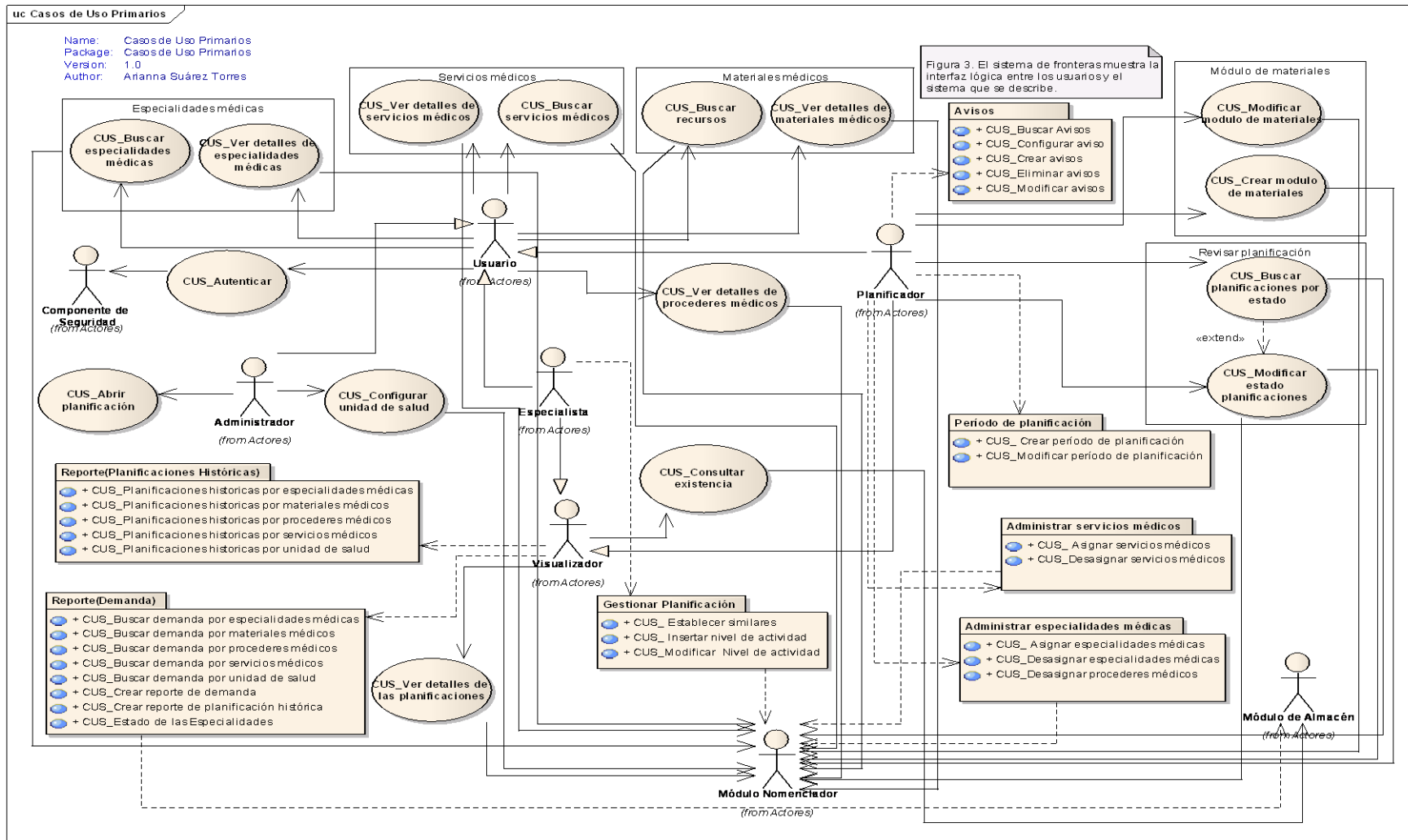


Figura 3. Diagrama de Casos de Uso del Sistema.

Capítulo 2: Características del Sistema

2.5.3. Descripción textual de Casos de Uso del Sistema

Un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica. A continuación se muestran los Casos de Uso del Sistema que serán implementados en esta iteración, los cuales fueron identificados a partir de los requisitos funcionales:

CUS_Ver detalles de las planificaciones.	CUS_Crear período de planificación.
CUS_Ver detalles de las especialidades médicas.	CUS_Modificar periodo de planificación.
CUS_Ver detalles de los materiales médicos.	CUS_Abrir planificación.
CUS_Ver detalles de los procedimientos médicos.	CUS_Buscar planificaciones por estados.
CUS_Crear módulo de materiales.	CUS_Modificar nivel de actividad.
CUS_Modificar módulo de materiales.	CUS_Insertar nivel de actividad.
CUS_Modificar estado planificaciones.	CUS_Buscar servicios médicos.
CUS_Autenticar.	CUS_Asignar servicios médicos.
CUS_Configurar unidad de salud.	CUS_Desasignar servicios médicos.
CUS_Buscar recursos.	CUS_Buscar especialidades médicas.
CUS_Estado de las especialidades.	CUS_Desasignar procedimientos médicos.
CUS_Buscar demanda por unidad de salud.	CUS_Desasignar especialidades médicas.
CUS_Buscar demanda.	CUS_Asignar especialidades médicas.

Tabla 2. Listado de Casos de Uso del Sistema para esta iteración.

También fueron identificados los siguientes Casos de Uso del Sistema, que de igual forma fueron documentados, sin embargo, serán implementados en la próxima iteración:

CUS_Establecer similares.	CUS_Consultar existencia.
CUS_Planificaciones históricas.	CUS_Crear avisos.
CUS_Buscar demanda por servicios médicos.	CUS_Buscar avisos.
CUS_Buscar demanda por materiales médicos.	CUS_Eliminar avisos.
CUS_Buscar demanda por procedimientos médicos.	CUS_Modificar avisos.
CUS_Buscar demanda por especialidades médicas.	CUS_Configurar avisos.
CUS_Planificaciones históricas por especialidades	CUS_Ver detalles de los servicios médicos.

Capítulo 2: Características del Sistema

médicas.	
CUS_Planificaciones históricas por materiales médicos.	CUS_Planificaciones históricas por unidad de salud.
CUS_Planificaciones históricas por procedimientos médicos.	CUS_Planificaciones históricas por servicios médicos.

Tabla 3. Listado de Casos de Uso del Sistema para la próxima iteración.

A continuación se muestra la descripción de algunos de los casos de uso arquitectónicamente significativos, las restantes descripciones se pueden encontrar en el Expediente de Proyecto.

Caso de Uso:	CUS_ Asignar especialidades médicas
Resumen:	El caso de uso inicia cuando el administrador selecciona la opción de Asignar->Especialidades . El sistema debe mostrar las especialidades médicas existentes y a partir de ahí, se procede a seleccionar las especialidades que se practicarán en la unidad de salud. El caso de uso termina una vez que la asignación se haya aceptado.
Complejidad:	Alta
Prioridad:	Crítico
Precondiciones:	Debe haberse definido el tipo de unidad de salud. (Debe ser un hospital.)
REFERENCIAS	
Requisitos:	RF11, RF17, RF25, RF 44, RF61, RF62, RF73, RF80.
Actores:	Planificador (Inicia)
Casos de Uso:	CUS_Desasignar especialidades médicas.(Extendido) CUS_Desasignar procedimientos médicos.(Extendido)
Poscondiciones:	Las especialidades fueron asignadas a la unidad de salud.

Tabla 4. Descripción textual del caso de uso: CUS_ Asignar especialidades médicas.

Caso de Uso:	CUS_ Insertar nivel de actividad
Resumen:	El caso de uso inicia cuando comienza el período de planificación y el especialista debe planificar cada uno de los procedimientos que se practican en la especialidad, insertando el índice de consumo de cada proceder. El caso de

Capítulo 2: Características del Sistema

	uso termina cuando se ha planificado para todos los procedimientos de la especialidad y la planificación sea enviada.
Complejidad:	Media
Prioridad:	Crítico
Precondiciones:	El sistema debe tener asignada la especialidad que se va a planificar.
REFERENCIAS	
Actores:	Especialista (Inicia)
Requisitos:	RF2, RF3, RF31, RF47, RF59, RF60, RF64, RF66, RF72, RF78, RF83.
Poscondiciones:	Queda planificada o no la especialidad médica. Queda planificado o no el proceder seleccionado con los materiales similares escogidos. El planificador puede revisar la planificación de esta especialidad.

Tabla 5. Descripción textual del caso de uso: CUS_ Insertar nivel de actividad.

Caso de Uso:	CUS_ Crear período de planificación
Resumen:	Este caso de uso se inicia cuando el planificador selecciona la opción Configuración->Crear período de planificación . Una vez realizada esta operación el planificador debe introducir los días con que cuenta el especialista para planificar y la fecha de inicio de esta planificación, así como la fecha de cierre de la planificación de la unidad de salud. Una vez realizadas estas operaciones el planificador debe escoger el período de planificación y fecha en que debe iniciar el mismo, una vez terminadas estas operaciones debe oprimir el botón aceptar, terminado de este modo el caso de uso.
Complejidad:	Baja
Prioridad:	Crítico
Precondiciones:	El usuario debe haberse autenticado satisfactoriamente.
REFERENCIAS	
Actores:	Planificador (Inicia)
Requisitos:	RF5, RF6, RF26, RF30, RF65.
Poscondiciones:	El sistema muestra en la interfaz la fecha en que deben comenzar a planificar el especialista.

Capítulo 2: Características del Sistema

Tabla 6. Descripción textual del caso de uso: CUS_ Crear período de planificación.

Caso de Uso:	CUS_ Modificar estado de planificaciones
Resumen:	Este caso de uso se inicia cuando el planificador revisa las planificaciones para aprobarlas. El caso de uso termina cuando acepta o rechaza la planificación.
Complejidad:	Baja
Prioridad:	Critico
Precondiciones:	Los especialistas deben haber enviado las planificaciones de sus departamentos.
REFERENCIAS	
Actores:	Planificador (Inicia)
Requisitos:	RF2, RF3, RF4, RF61, RF63, RF64, RF66, RF67, RF70, RF72, RF73, RF77, RF78, RF79, RF81
Casos de Uso:	CUS_Buscar planificaciones por estados (Extendido)
Poscondiciones:	Han sido rechazadas o aceptadas las planificaciones realizadas.

Tabla 7. Descripción textual del caso de uso: CUS_ Modificar estado de planificaciones.

Conclusiones

El desarrollo de este capítulo propició un mayor entendimiento de la propuesta realizada, teniendo en cuenta que se describieron las características del sistema en términos de requerimientos funcionales y no funcionales. Se logró identificar los actores y los casos de uso del sistema, los cuales fueron modelados gráficamente mediante el Diagrama de Casos de Uso del Sistema, así como la descripción de los mismos. Este capítulo sentó las bases para comenzar el flujo de diseño del Módulo de Planificación v1.2.

Capítulo3: Diseño del Sistema

CAPÍTULO 3: DISEÑO DEL SISTEMA

En este capítulo tiene como propósito decidir cómo se implementará el sistema. Durante el diseño se toman decisiones tácticas y estratégicas para cumplir los requisitos funcionales y no funcionales de un sistema, para obtener una arquitectura robusta para la implementación. Se muestran artefactos de este flujo entre los que se encuentran el Diagrama de Clases del Diseño y Diagrama de Secuencia.

3.1. Modelo de diseño

En el flujo de trabajo de diseño, se modela el sistema de manera que soporte todos los requerimientos, tanto los funcionales como los no funcionales. El propósito del diseño es especificar una solución que trabaje y pueda ser fácilmente convertida en código fuente, y construir una arquitectura simple y extensible.

3.1.1. Estructura del diseño

Para lograr una mayor organización y entendimiento el diseño se estructuró por subsistemas con sus interfaces y sus dependencias, basándose en el patrón de arquitectura MVC que propone el Framework Symfony, donde se separan los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.

En la Capa del Controlador se encuentra un único controlador frontal para todo el sistema y una clase acción para cada módulo definido. El controlador frontal actúa como el punto de entrada a la aplicación, es quien toma las peticiones del usuario y las redirecciona a la clase acción correspondiente. Esta clase contiene los métodos que permiten dar respuesta a la solicitud.

En la Capa Vista se encuentran los formularios e interfaces determinadas para las funcionalidades que debe cumplir el sistema y en la Capa del Modelo se encuentran las clases que permiten el acceso a datos representadas por las clasesPeer.php y las clases que permiten la abstracción de la base de datos representadas por las clases.php.

Una vez que el usuario realiza una petición en una página cliente, esta le envía la petición al controlador frontal y este la redirecciona a la acción correspondiente. La clase acción es la encargada de dar la respuesta a través del controlador, construye la página cliente que debe mostrarse, la cual contiene un formulario. Una vez que el cliente realice una entrada de datos, esta información es enviada al controlador frontal, que envía la información a la acción correspondiente y de ser necesario accede a la base de datos a través de los métodos implementados en la lógica del negocio que se encuentran en el modelo.

Capítulo 3: Diseño del Sistema

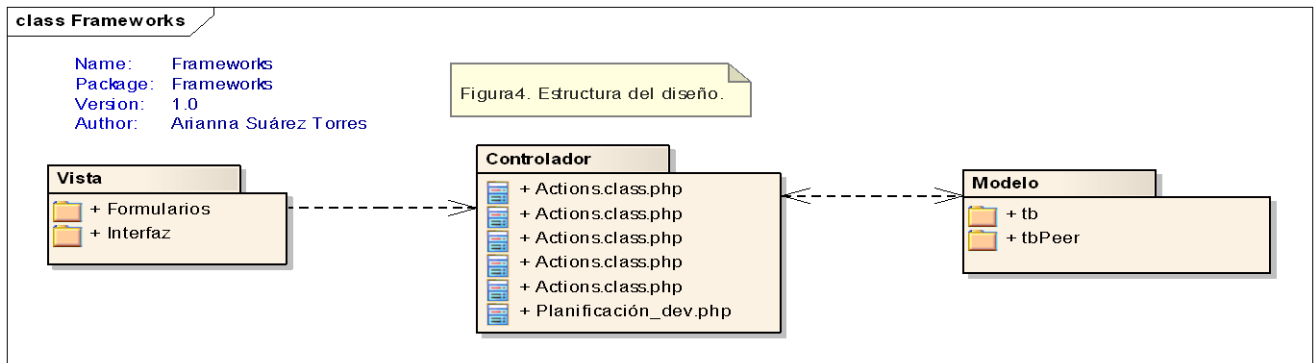


Figura 4. Estructura del diseño.

3.1.2. Descripción de elementos del diseño

Para la realización de los Diagramas de Clases del Diseño, las clases están representadas con estereotipos como: Server Page, Client Page y Form, debido a que son los elementos que se utilizan para la representación de una página web que es lo que constituye el Módulo de Planificación.

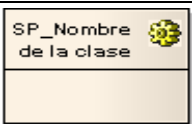
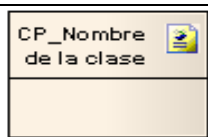
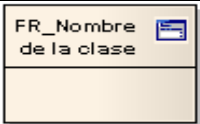
Estereotipos Web para las Clases del diseño	
Estereotipo	Descripción
 Server Page	Representa una página Web que tiene scripts ejecutados por el servidor. Estos scripts interactúan con los recursos que se encuentran al alcance del servidor. Sólo puede mantener relaciones con objetos que se encuentren en el servidor.
 Client Page	Representan páginas que son mostradas por el navegador web y pueden ser una combinación de algún o algunos lenguajes de marcado, scripts del lado del cliente.
 Form	Representa una colección de campos de entrada que forman parte de la página cliente (Client Page).

Tabla 8. Estereotipos Web para las Clases del Diseño.

Capítulo3: Diseño del Sistema

Estereotipos para las Relaciones entre las Clases	
Link	Representa un apuntador desde una “client page” hacia una “client page” o “server page”.
Submit	Esta relación siempre ocurre entre una “form” y una “server page”, la “server page” procesa los datos que la “form” le envía (submit).
Build	Sirve para la “server page” que crea la “client page”. Una “server page” puede crear varias “client page”, pero una “client page” sólo puede ser creada por una sola “server page”. Esta relación siempre es unidireccional.
Redirect	Esta es también una relación unidireccional que indica que una página Web redirige hacia otra.

Tabla 9. Estereotipos para las relaciones entre las Clases. [41]

3.1.3. Diagramas de Clases del Diseño

Uno de los artefactos más importantes a obtener del flujo de trabajo de diseño son los diagramas de clases de diseño, donde se exponen las clases que intervienen en las realizaciones de los casos de uso del sistema. Este modelo se puede utilizar para visualizar la implementación y para soportar las técnicas de programación gráfica de la aplicación debido a que en ellos se muestran los atributos y métodos de cada clase y se representa de una forma sencilla la colaboración y las responsabilidades de las distintas clases que forman el sistema. [42]

A continuación se muestran algunos de los Diagramas de Clases del Diseño de Casos de Uso Arquitectónicamente significativos; los restantes pueden encontrarse en el Expediente de Proyecto.

Capítulo 3: Diseño del Sistema

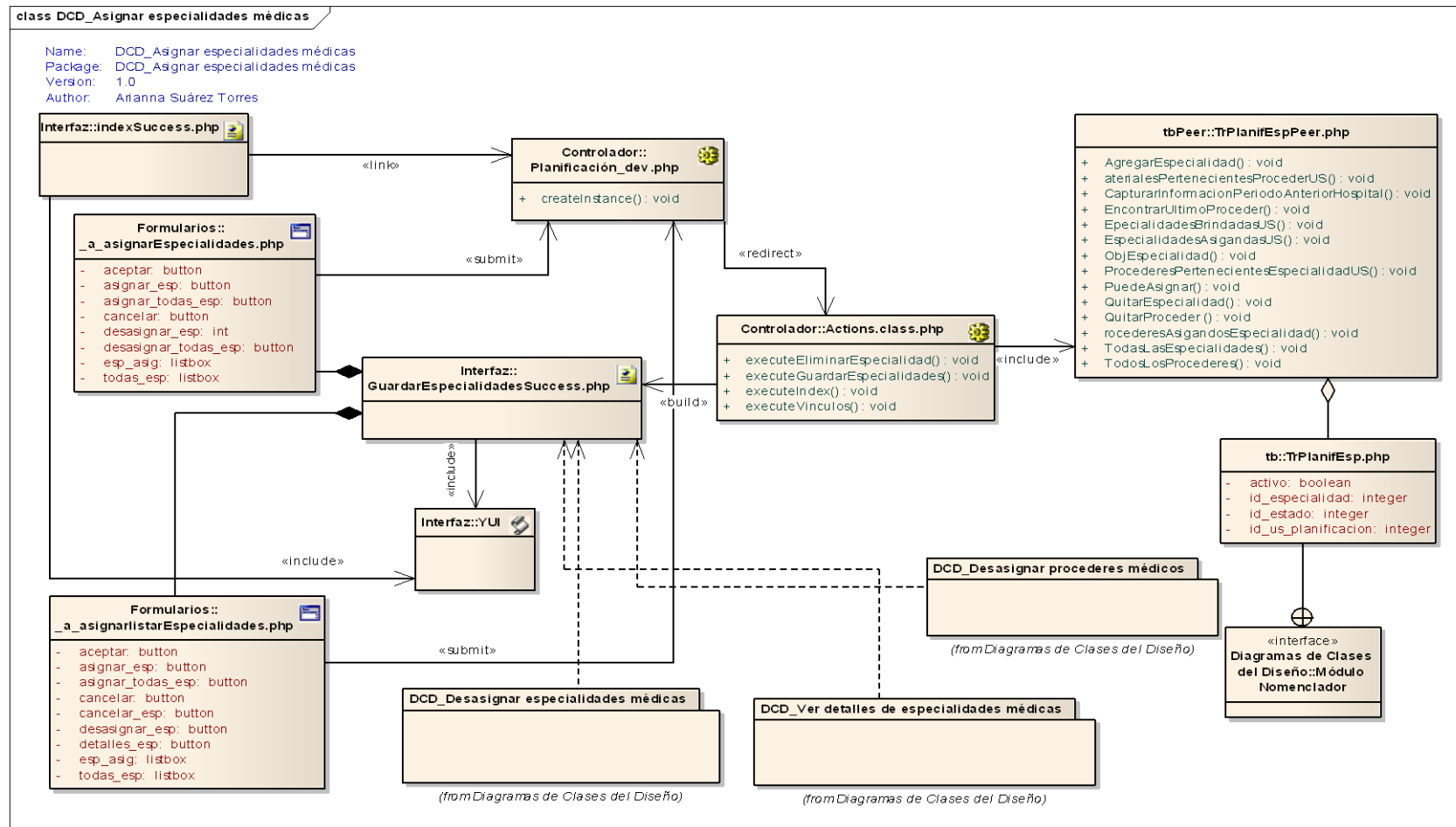


Figura 5. Diagrama de Clases del Diseño: DCD_Asignar especialidades médicas.

Capítulo 3: Diseño del Sistema

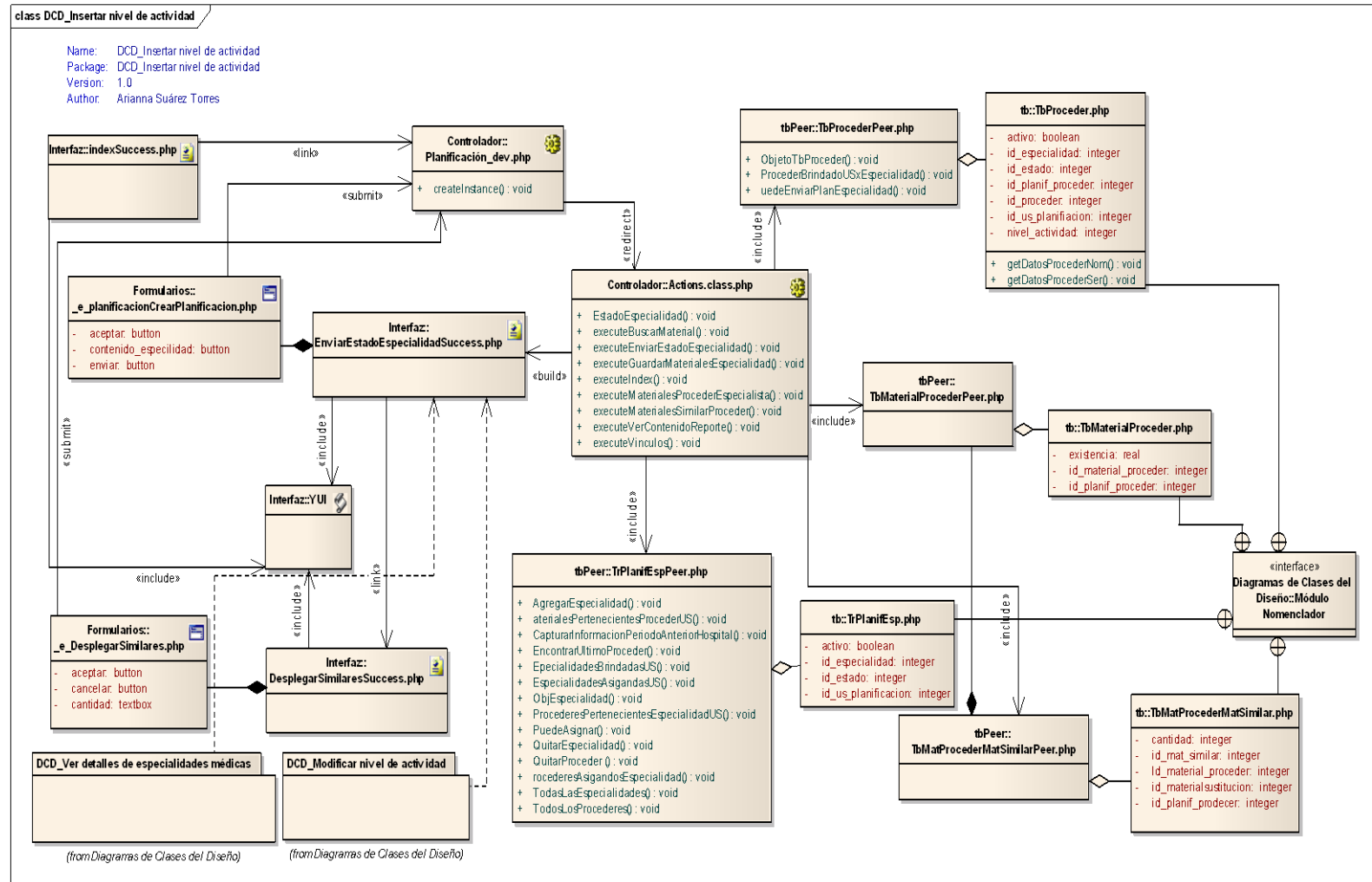


Figura 6. Diagrama de Clases del Diseño: DCD_Insertar nivel de actividad.

Capítulo 3: Diseño del Sistema

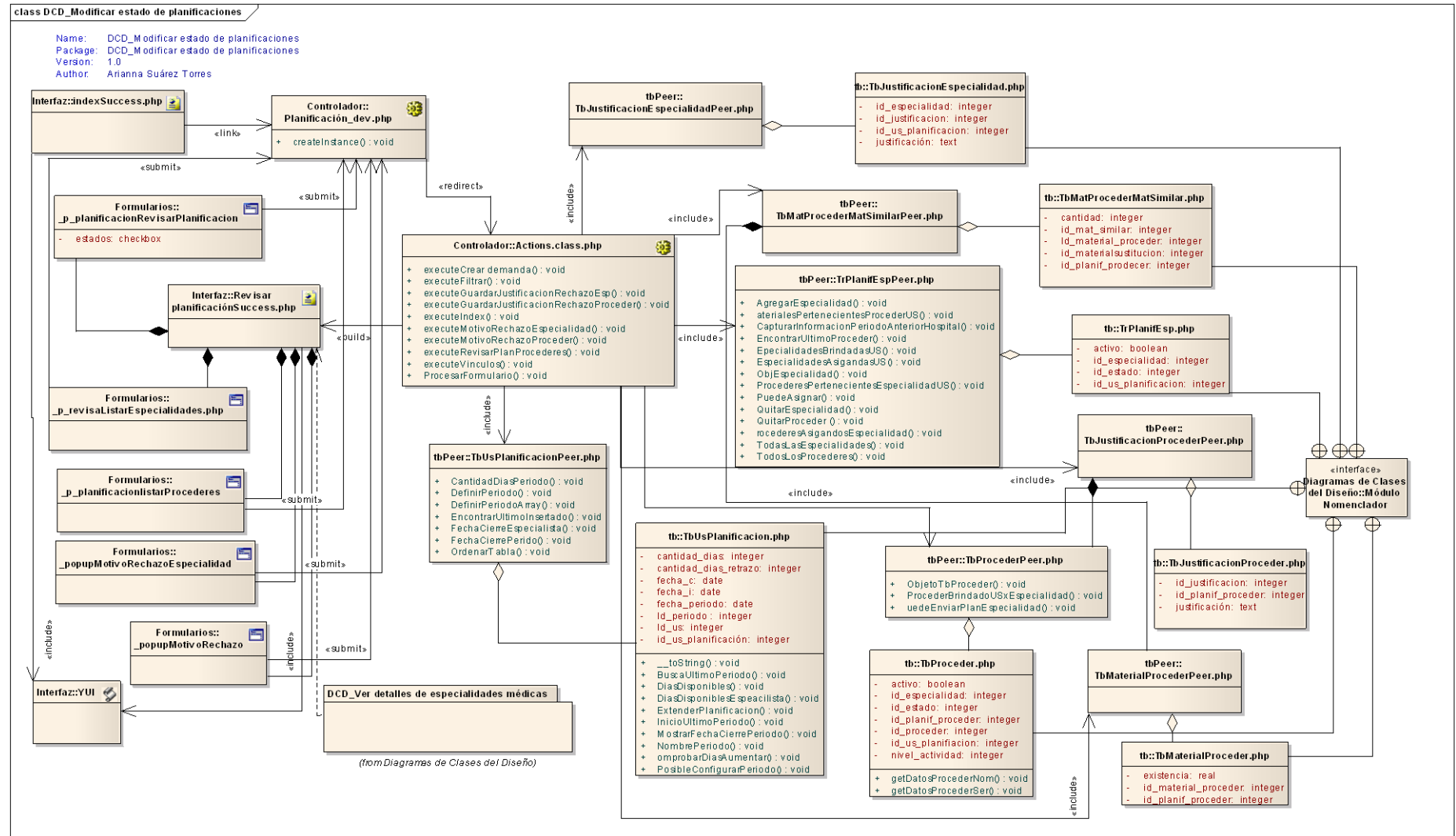


Figura 7. Diagrama de Clases del Diseño: DCD_Modificar estado de planificaciones.

Capítulo 3: Diseño del Sistema

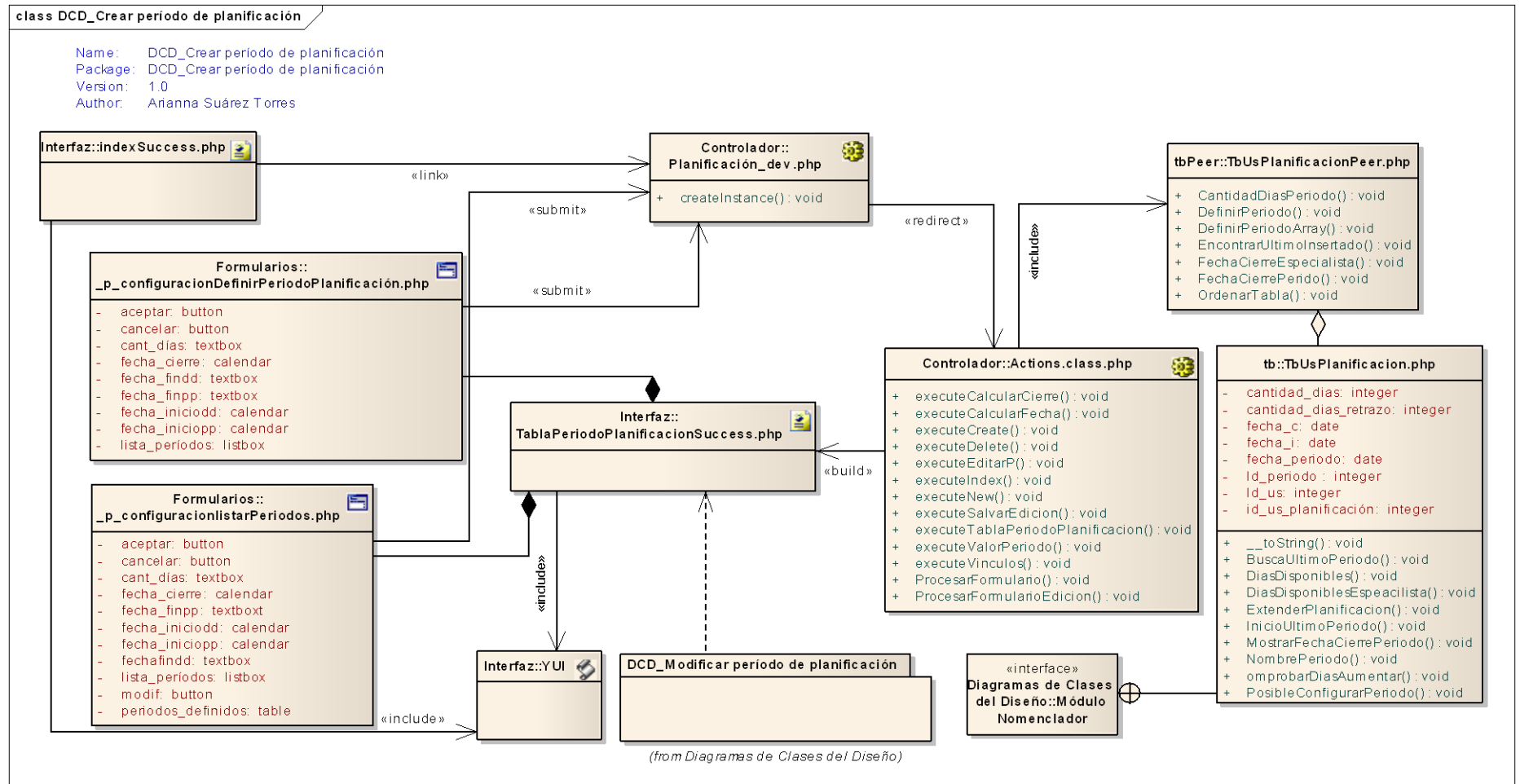


Figura 8. Diagrama de Clases del Diseño: DCD_Crear periodo de planificación.

Capítulo 3: Diseño del Sistema

3.1.4. Diagramas de Secuencia

Para mostrar las interacciones entre objetos ordenados en secuencia temporal, se realizaron los diagramas de secuencia para cada uno de los casos de uso. Estos diagramas contienen detalles de implementación del escenario, incluyendo los objetos y clases utilizadas, así como los mensajes enviados entre los objetos.

A continuación se muestran algunos de los diagramas de secuencia concebidos para el sistema a desarrollar; el resto se puede encontrar en el Expediente de Proyecto.

Capítulo 3: Diseño del Sistema

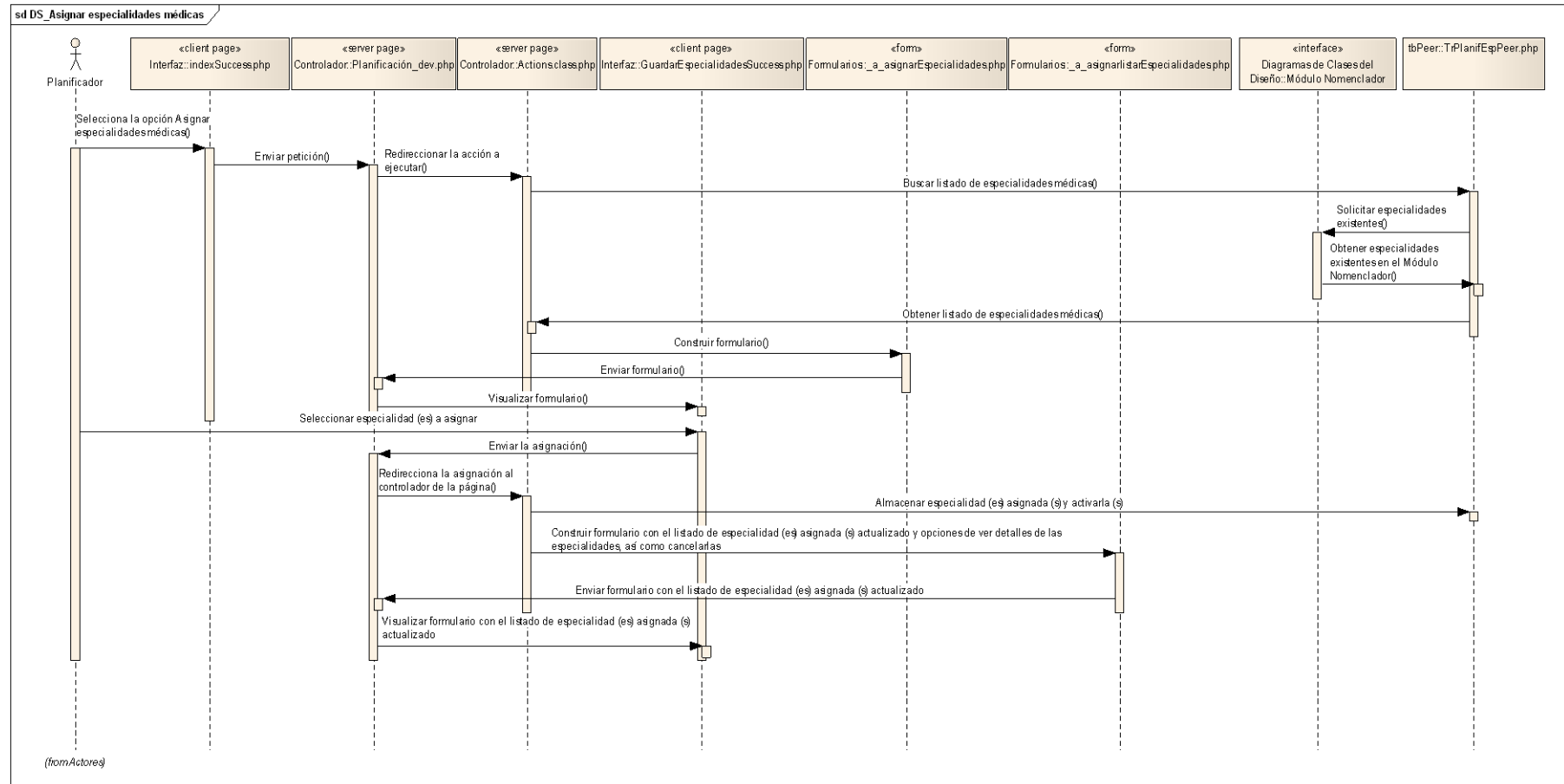


Figura 9. Diagrama de Secuencia: DS_Asignar especialidades médicas.

Capítulo 3: Diseño del Sistema

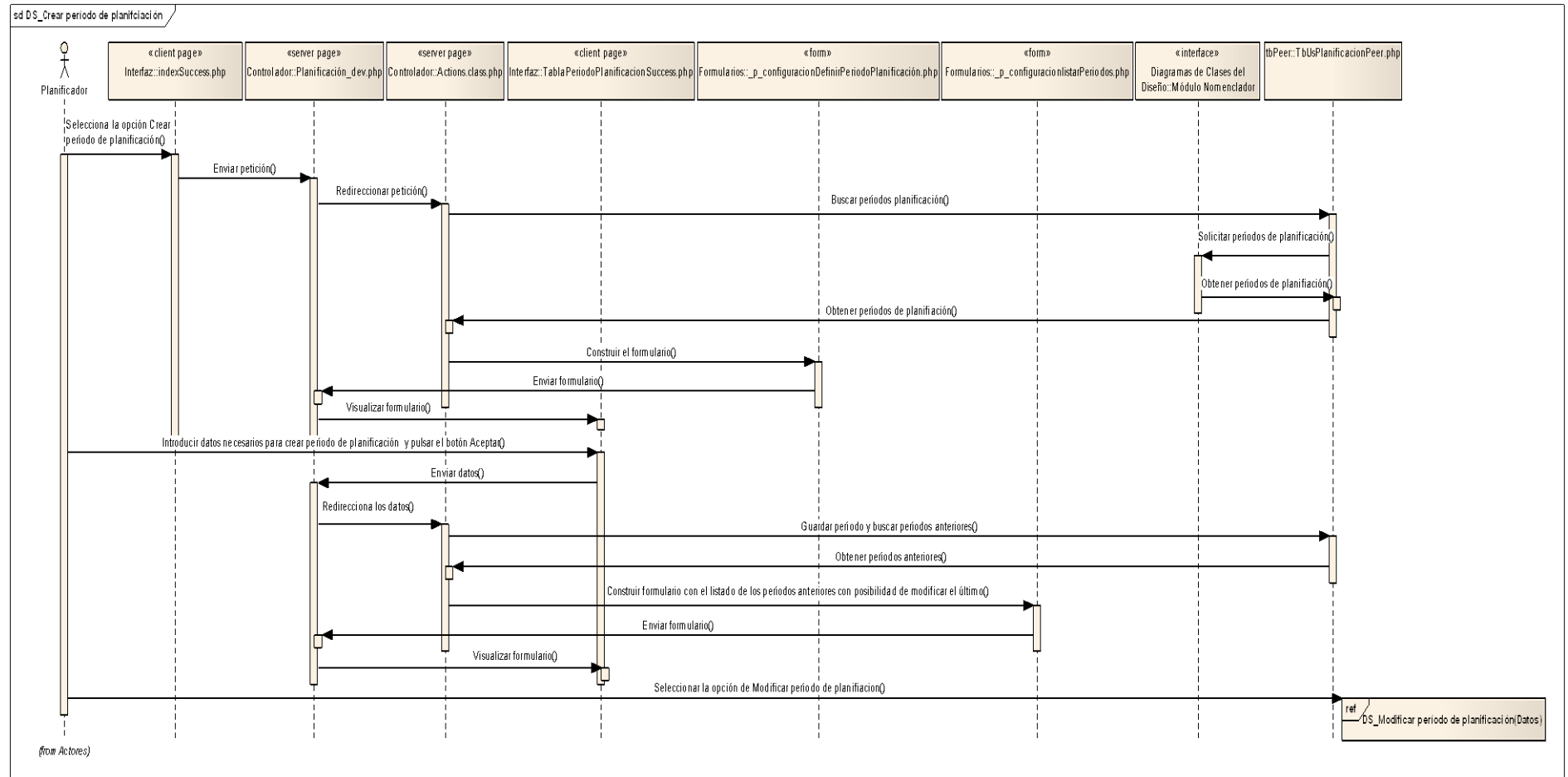


Figura 11. Diagrama de Secuencia: DS_Crear periodo de planificación.

Capítulo 3: Diseño del Sistema

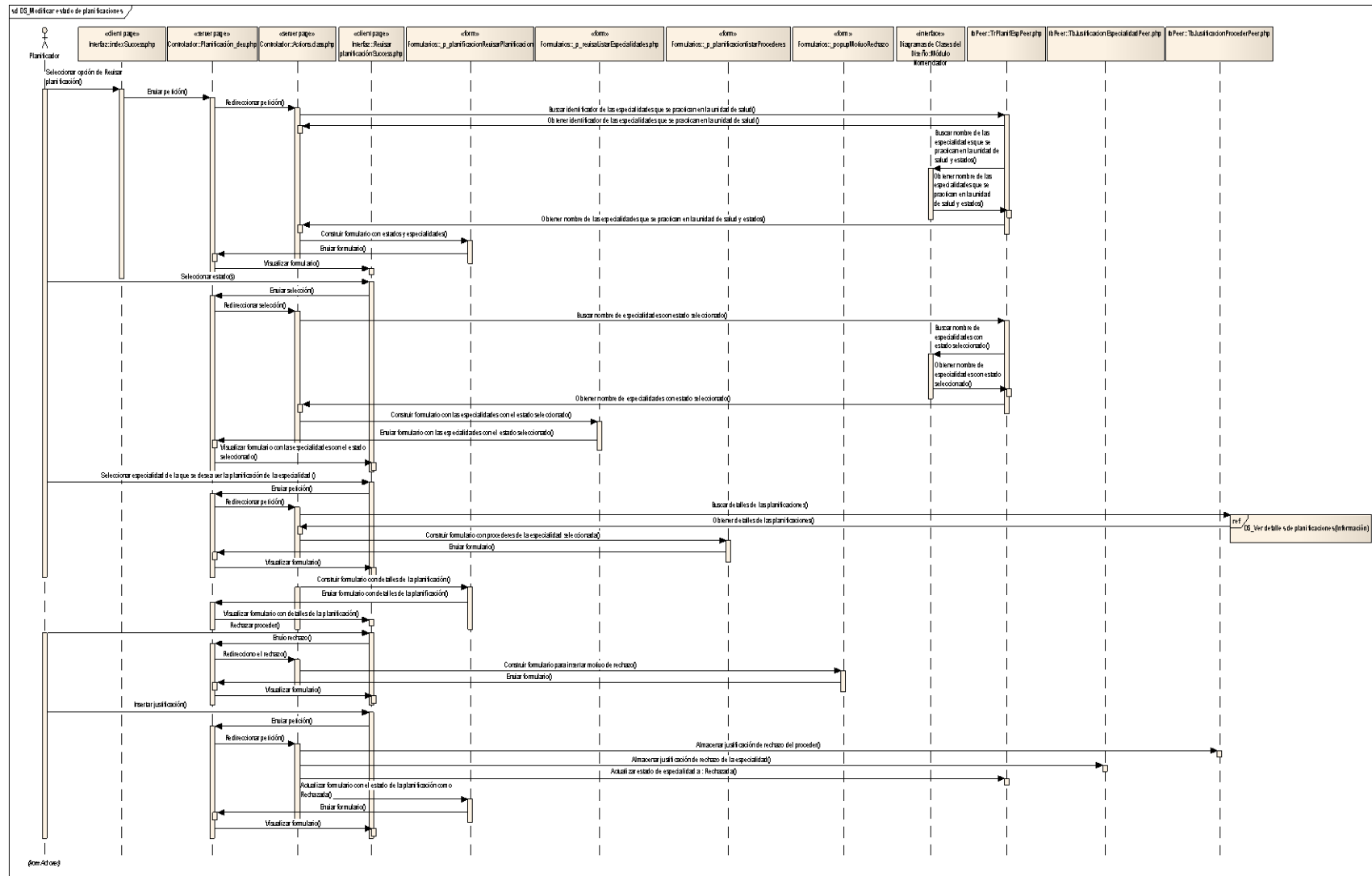


Figura 12. Diagrama de Secuencia: DS_Modificar estado de planificaciones.

Capítulo 3: Diseño del Sistema

3.1.5. Descripción de las clases

Las descripciones de las clases del diseño ayudan a comprender mejor las responsabilidades de las mismas y al mismo tiempo tener una visión más exacta del diseño del sistema.

3.1.5.1. Descripción de páginas Clientes

Nombre: indexSuccess.php
Tipo de clase: Client Page
Descripción general: Esta es la página de bienvenida que brinda al usuario la posibilidad de dirigirse a otras páginas.

Tabla 10. Descripción de la página cliente: indexSuccess.php.

Nombre: GuardarEspecialidadesSuccess.php
Tipo de clase: Client Page
Descripción general: Esta es la página que brinda al usuario la posibilidad de asignar, desasignar y cancelar las especialidades médicas. La misma consta de un listado de especialidades existentes para que se asignen a la unidad de salud las que se van a practicar en la misma y de esta forma se construye una lista de especialidades asignadas que pueden ser desasignadas. Una vez asignadas las especialidades, se construye una lista de las mismas con su descripción y además, brinda la opción de cancelarlas y de ver sus detalles. Es utilizada por los casos de uso: <ul style="list-style-type: none">➤ CUS_Asignar especialidades médicas.➤ CUS_Desasignar especialidades médicas.

Tabla 11. Descripción de la página cliente: GuardarEspecialidadesSuccess.php.

Nombre: EnviarEstadoEspecialidadSuccess.php
Tipo de clase: Client Page
Descripción general: Esta es la página que brinda al usuario la posibilidad de planificar una especialidad médica según sus procedimientos. La misma consta de un listado de los procedimientos de la especialidad y las opciones de ver sus detalles y de planificar según los materiales del mismo que el mismo utilice. Es utilizada por el caso de uso: <ul style="list-style-type: none">➤ CUS_Insertar nivel de actividad.

Capítulo 3: Diseño del Sistema

- CUS_Modificar nivel de actividad.

Tabla 12. Descripción de la página cliente: EnviarEstadoEspecialidadSuccess.php.

Nombre: Aumentar_Dias_RetrazoSuccess.php
Tipo de clase: Client Page
Descripción general: Esta es la página que permite al usuario aumentar días al período establecido para realizar la planificación. La misma consta de dos campos con el objetivo de que se introduzcan los días que serán aumentados al período y la justificación de la necesidad de abrir la planificación. Es utilizada por el caso de uso: <ul style="list-style-type: none">➤ CUS_Abrir planificación.

Tabla 13. Descripción de la página cliente: Aumentar_Dias_RetrazoSuccess.php.

Nombre: TablaPeriodoPlanificacionSuccess.php
Tipo de clase: Client Page
Descripción general: Esta es la página que permite al usuario crear el período de planificación para el cual se va a planificar y durante que tiempo. Una vez creado el período de tiempo, brinda un listado de los períodos de tiempo definidos para el mismo período, y a partir de ahí, permite la posibilidad de modificar el período definido. Es utilizada por los casos de uso: <ul style="list-style-type: none">➤ CUS_Crear período de planificación.➤ CUS_Modificar período de planificación.

Tabla 14. Descripción de la página cliente: TablaPeriodoPlanificacionSuccess.php.

Nombre: Revisar planificacionSuccess.php
Tipo de clase: Client Page
Descripción general: Esta es la página que brinda al usuario un listado de todas las especialidades y la opción de seleccionar el(los) estado(s) de las planificaciones que se desean revisar. Este listado se

Capítulo3: Diseño del Sistema

muestra a partir de un paginado definido. Brinda la posibilidad de ver detalles de las especialidades, así como de su planificación. Es utilizada por el caso de uso:

- CUS_Modificar estado de planificaciones.
- CUS_Buscar planificaciones por estado.

Tabla 15. Descripción de la página cliente: Revisar planificaciónSuccess.php.

3.1.5.2. Descripción de páginas Servidoras

Nombre: Planificación_dev.php
Tipo de clase: Server Page
Descripción general: Esta página es el controlador frontal del sistema y constituye el punto de entrada único de toda la aplicación en un entorno determinado, siendo la encargada de detectar la acción que se debe ejecutar.

Tabla 16. Descripción de la página servidora: Planificación_dev.php.

3.2. Modelo de Datos

A continuación se muestra el Modelo de Datos definido para describir la representación lógica y física de la información persistente manejada por el Módulo de Planificación. El mismo fue realizado a partir de la base de datos previamente creada.

Capítulo 3: Diseño del Sistema

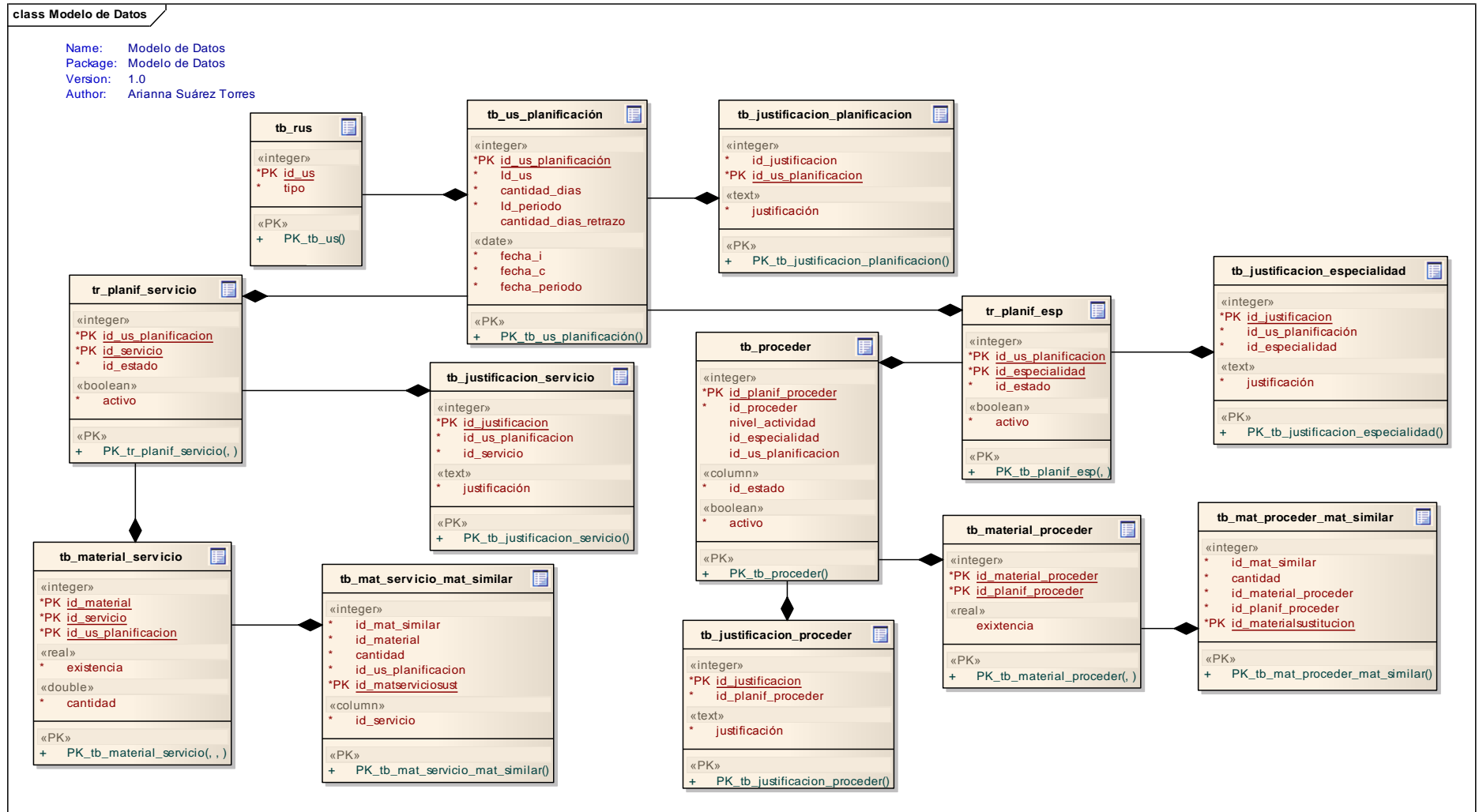


Figura 13. Modelo de Datos.

Capítulo 3: Diseño del Sistema

3.2.1. Descripción de las tablas de la Base de Datos

Las tablas utilizadas para el almacenamiento de los datos que se utilizarán, se describen a continuación:

Nombre: tb_us_planificación	
Descripción: Almacena información que permite la representación de una planificación en una unidad de salud, según los campos asociados, ejemplos: fecha de inicio, cierre y período, cantidad de días.	
Atributo	Tipo
id_us_planificación	integer
id_us	integer
fecha_i	date
fecha_c	date
cantidad_dias	integer
fecha_periodo	date
id_periodo	integer
cantidad_dias_retraso	integer

Tabla 17. Descripción de tb_us_planificación.

Nombre: tb_justificacion_planificacion	
Descripción: Almacena información referente a las justificaciones de una planificación en caso de haber un retraso.	
Atributo	Tipo
id_justificacion	integer
justificación	text
id_us_planificacion	integer

Tabla 18. Descripción de tb_justificacion_planificacion.

Nombre: tb_rus	
Descripción: Almacena información referente a los campos que se encuentran en tb_us que pueden identificar a una unidad de salud y definir su tipo.	

Capítulo 3: Diseño del Sistema

Atributo	Tipo
id_us	integer
tipo	integer

Tabla 19. Descripción de tb_rus.

Nombre: tb_planif_servicio	
Descripción: Almacena información referente la planificación del (los) servicio (s) prestado (s) de acuerdo al tipo de unidad de salud (policlínicos).	
Atributo	Tipo
id_us_planificacion	integer
id_servicio	integer
id_estado	integer
activo	boolean

Tabla 20. Descripción de tb_planif_servicio.

Nombre: tb_justificacion_servicio	
Descripción: Almacena información referente a las justificaciones de la planificación en caso de haber un retraso.	
Atributo	Tipo
id_justificacion	integer
justificación	text
id_us_planificacion	integer
id_servicio	integer

Tabla 21. Descripción de tb_justificacion_servicio.

Nombre: tb_material_servicio	
Descripción: Almacena información referente a los materiales a utilizar por el servicio.	
Atributo	Tipo
id_material	integer
id_us_planificacion	integer
id_servicio	integer

Capítulo 3: Diseño del Sistema

existencia	real
cantidad	double precision

Tabla 22. Descripción de tb_material_servicio.

Nombre: tb_mat_servicio_mat_similar	
Descripción: Almacena información referente a los materiales similares (materiales que pueden sustituir a otros materiales) del servicio.	
Atributo	Tipo
id_matserviciosust	integer
cantidad	integer
id_material	integer
id_mat_similar	integer
id_us_planificacion	integer
id_servicio	integer

Tabla 23. Descripción de tb_mat_servicio_mat_similar.

Nombre: tb_planif_esp	
Descripción: Almacena información referente la planificación de la (s) especialidad (es), presente (s) en algunas unidades de salud (hospitales y clínicas estomatológicas).	
Atributo	Tipo
id_us_planificacion	integer
id_especialidad	integer
id_estado	integer
activo	boolean

Tabla 24. Descripción de tb_planif_esp.

Nombre: tb_justificacion_especialidad	
Descripción: Almacena información referente a justificaciones de la planificación de la especialidad en la cual hubo un retraso en la planificación.	

Capítulo 3: Diseño del Sistema

Atributo	Tipo
id_justificacion	integer
justificación	text
id_us_planifiacion	integer
id_especialidad	integer

Tabla 25. Descripción de tb_justificacion_especialidad.

Nombre: tb_proceder	
Descripción: Almacena información referente a los procederes que puede realizar una especialidad.	
Atributo	Tipo
id_planif_proceder	integer
id_proceder	integer
nivel_actividad	integer
id_estado	integer
id_us_planificacion	integer
id_especialidad	integer
activo	boolean

Tabla 26. Descripción de tb_proceder.

Nombre: tb_justificacion_proceder	
Descripción: Almacena información referente a las justificaciones de un proceder en el cual hubo un retraso en la planificación.	
Atributo	Tipo
id_justificacion	integer
justificación	text
id_planif_proceder	integer

Tabla 27. Descripción de tb_justificacion_proceder.

Capítulo 3: Diseño del Sistema

Nombre: tb_material_proceder	
Descripción: Almacena información referente a los materiales que son empleados en un proceder.	
Atributo	Tipo
id_material_proceder	integer
id_planif_proceder	integer
existencia	real

Tabla 28. Descripción de tb_material_proceder.

Nombre: tb_mat_proceder_mat_similar	
Descripción: Almacena información referente a los materiales y los materiales similares relacionados con un proceder determinado.	
Atributo	Tipo
id_materialesustitucion	integer
cantidad	integer
id_mat_similar	integer
id_material_proceder	integer
id_planif_proceder	integer

Tabla 29. Descripción de tb_mat_proceder_mat_similar.

Conclusiones

Con el desarrollo de este capítulo se logró modelar el sistema de manera que soporte los requisitos funcionales y no funcionales. También se realizó una estructuración de los elementos del diseño creando una organización entre las clases que serán utilizadas durante la implementación del sistema a desarrollar. Finalmente se obtuvo el modelo de datos en el que se reflejan las tablas de la base de datos a utilizar para almacenar la información que será manejada en el sistema.

Capítulo4: Implementación

CAPÍTULO 4: IMPLEMENTACIÓN

En este capítulo, basándose en el diseño del sistema se va a realizar la implementación en términos de componentes. Se construyen los diagramas de Componente y de Despliegue.

4.1 Modelo de Implementación

El Modelo de Implementación es comprendido por un conjunto de componentes y subsistemas que constituyen la composición física de la implementación del sistema. Entre los componentes se encuentran: datos, archivos, ejecutables, código fuente y los directorios. Fundamentalmente, se describe la relación que existe desde los paquetes y clases del modelo de diseño a subsistemas y componentes físicos.

Este artefacto describe cómo se implementan los componentes, congregándolos en subsistemas organizados en capas y jerarquías, y señala las dependencias entre éstos. [43]

4.1.1. Diagrama de Componentes

Los diagramas de componentes son usados para estructurar el modelo de implementación en términos de subsistemas de implementación y mostrar las relaciones entre los elementos de implementación. [44]

El diagrama que se muestra a continuación, representa la estructura y organización del Módulo de Planificación en términos de componentes, y la aplicación mediante la arquitectura Modelo Vista Controlador que propone el framework Symfony.

Capítulo 4: Implementación

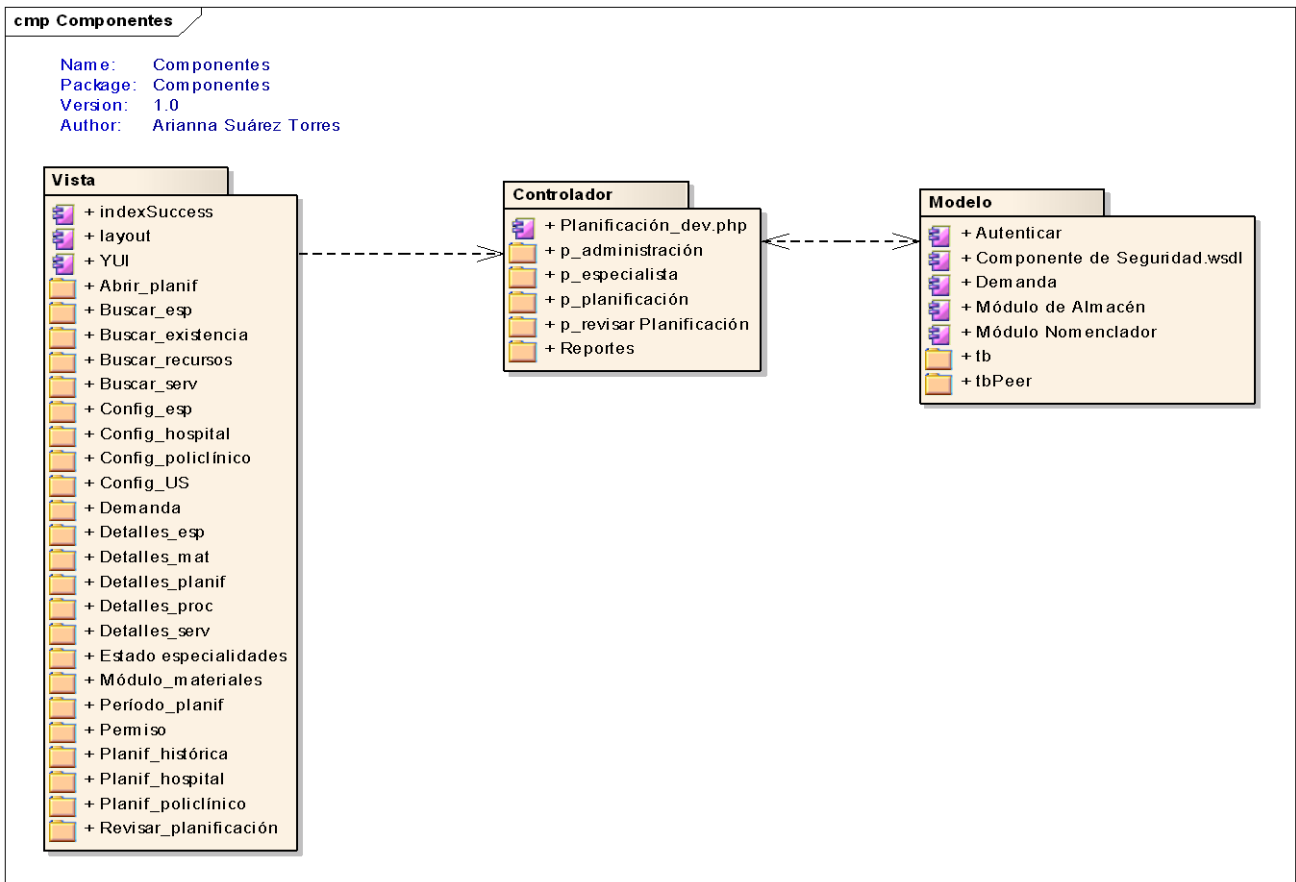


Figura 14. Diagrama de Componentes.

La Capa Vista permite mostrar al usuario una página web y le posibilita interactuar con la misma. Para ello cuenta con la página cliente y la librería YUI que la complementa proporcionándole dinamismo a la página.

Capítulo 4: Implementación

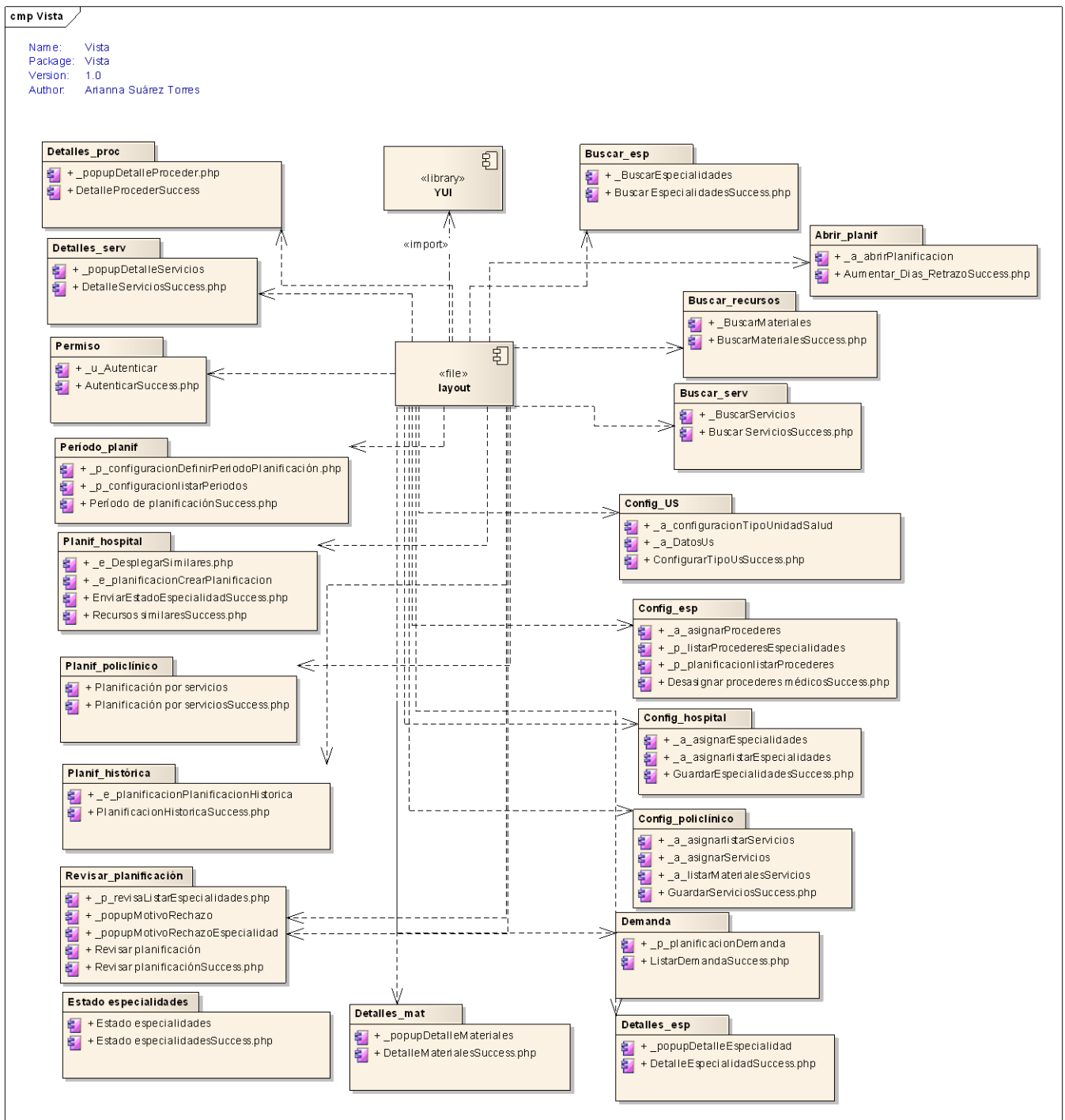


Figura 15. Diagrama de Componente Paquete Vista.

La Capa Modelo proporciona el acceso a datos, está constituida por las clases Peer las cuales contienen métodos estáticos para obtener los datos de las clases, así como su relación con otros

Capítulo 4: Implementación

módulos que brindan información al sistema como el Módulo Nomenclador, el Módulo de Almacén y el Componente de Seguridad.

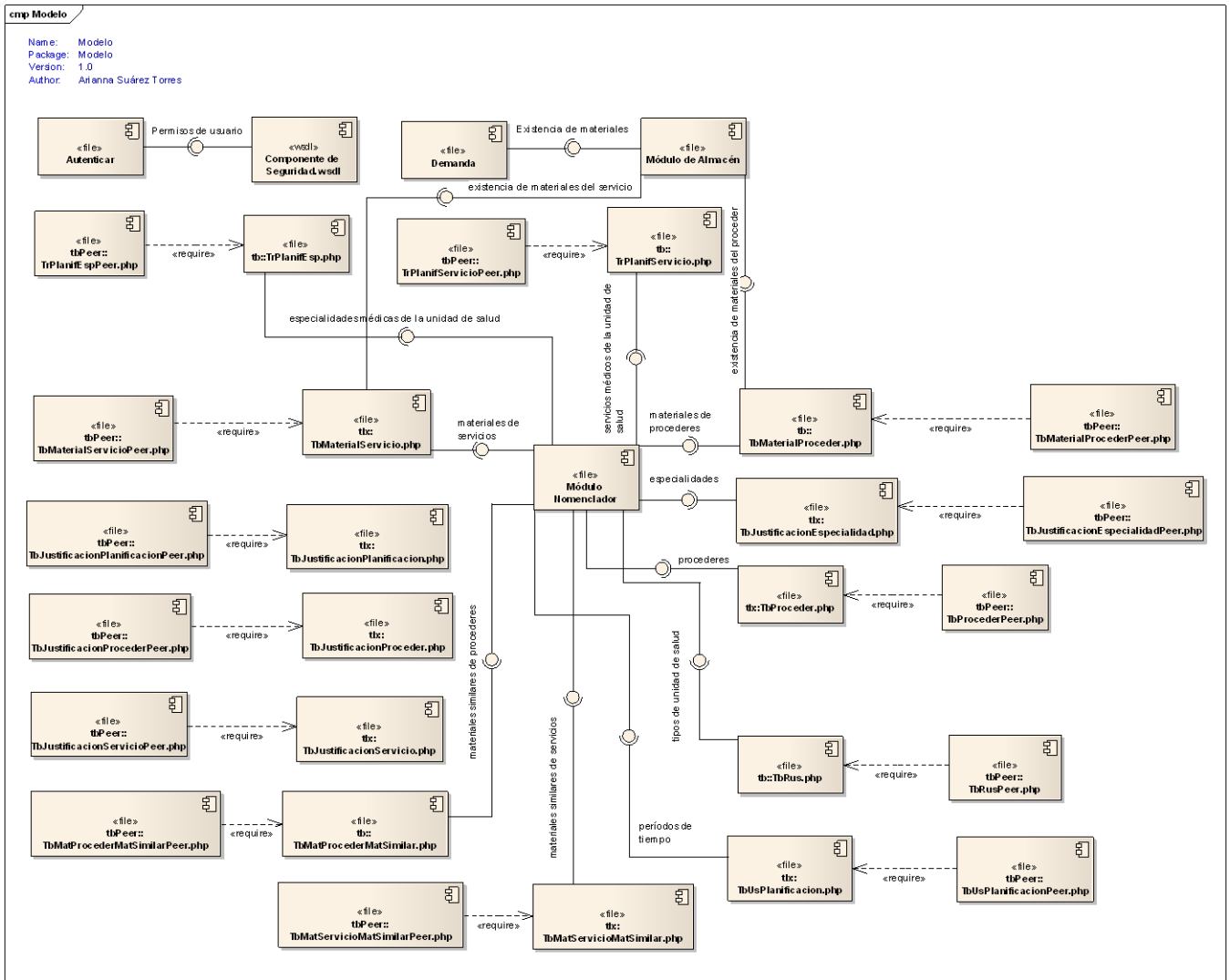


Figura 16. Diagrama de Componente Paquete Modelo.

En la Capa del Controlador se encuentra el Controlador Frontal quien actúa como único punto de entrada a la aplicación, se encarga de gestionar las peticiones realizadas en la aplicación y determina la acción a ejecutar.

Capítulo 4: Implementación

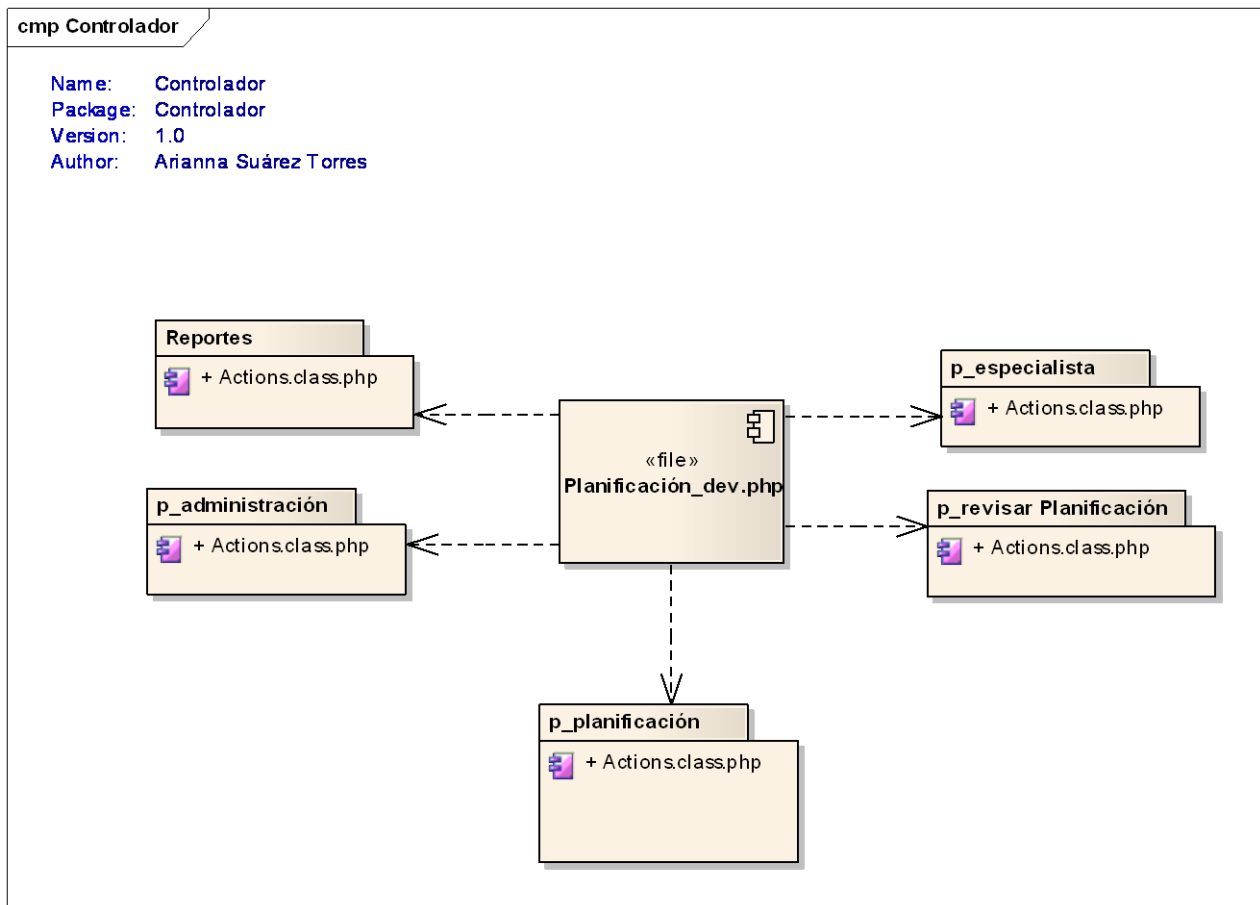


Figura 17. Diagrama de Componente Paquete Controlador.

Una vez mostrada la distribución del sistema mediante los paquetes creados para la organización de los componentes definidos, se muestran algunos de los Diagramas de Componentes que se realizaron durante este flujo de implementación, los restantes se podrán encontrar en el Expediente de Proyecto.

Capítulo 4: Implementación

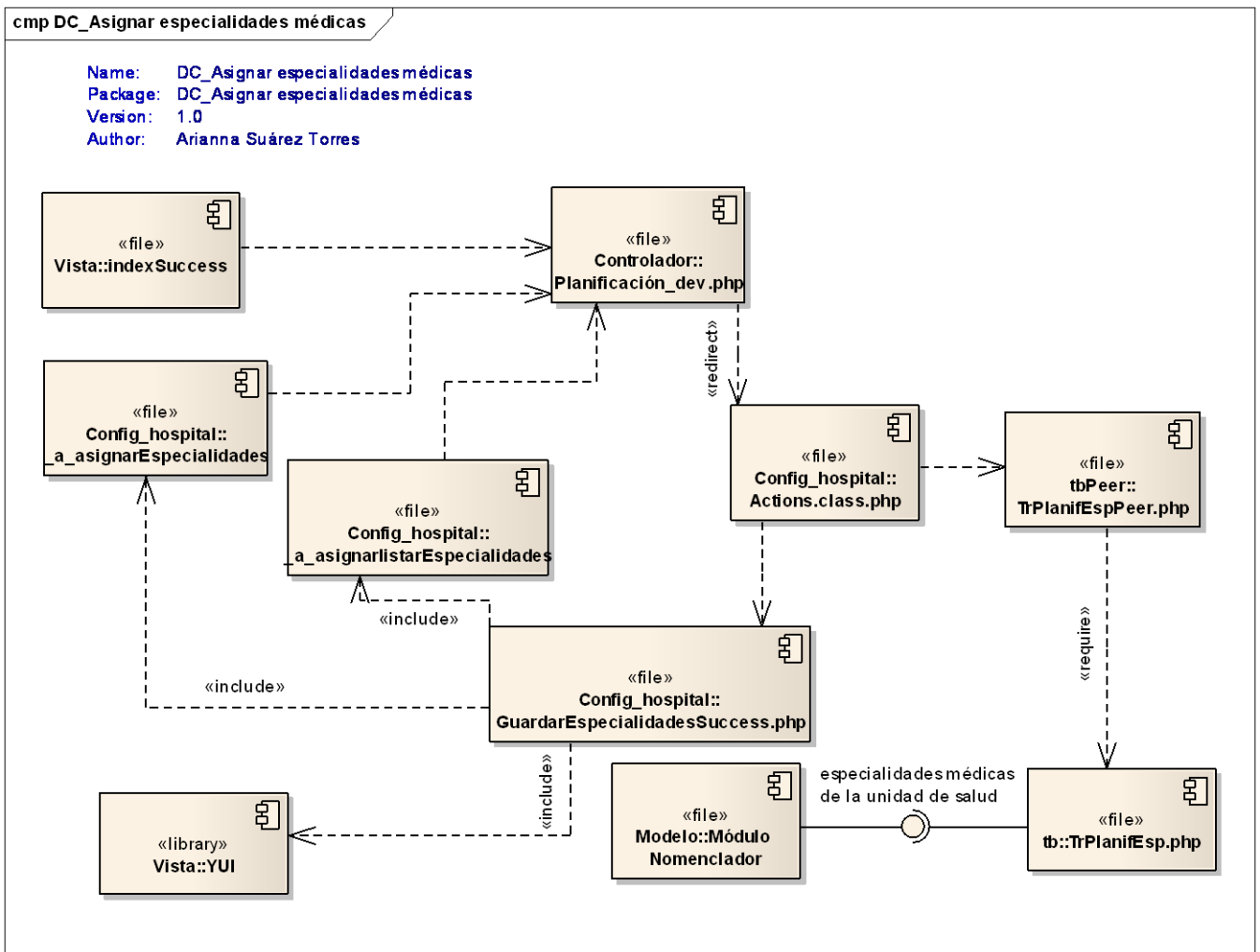


Figura 18. Diagrama de Componentes: DC_Asignar especialidades médicas.

Capítulo 4: Implementación

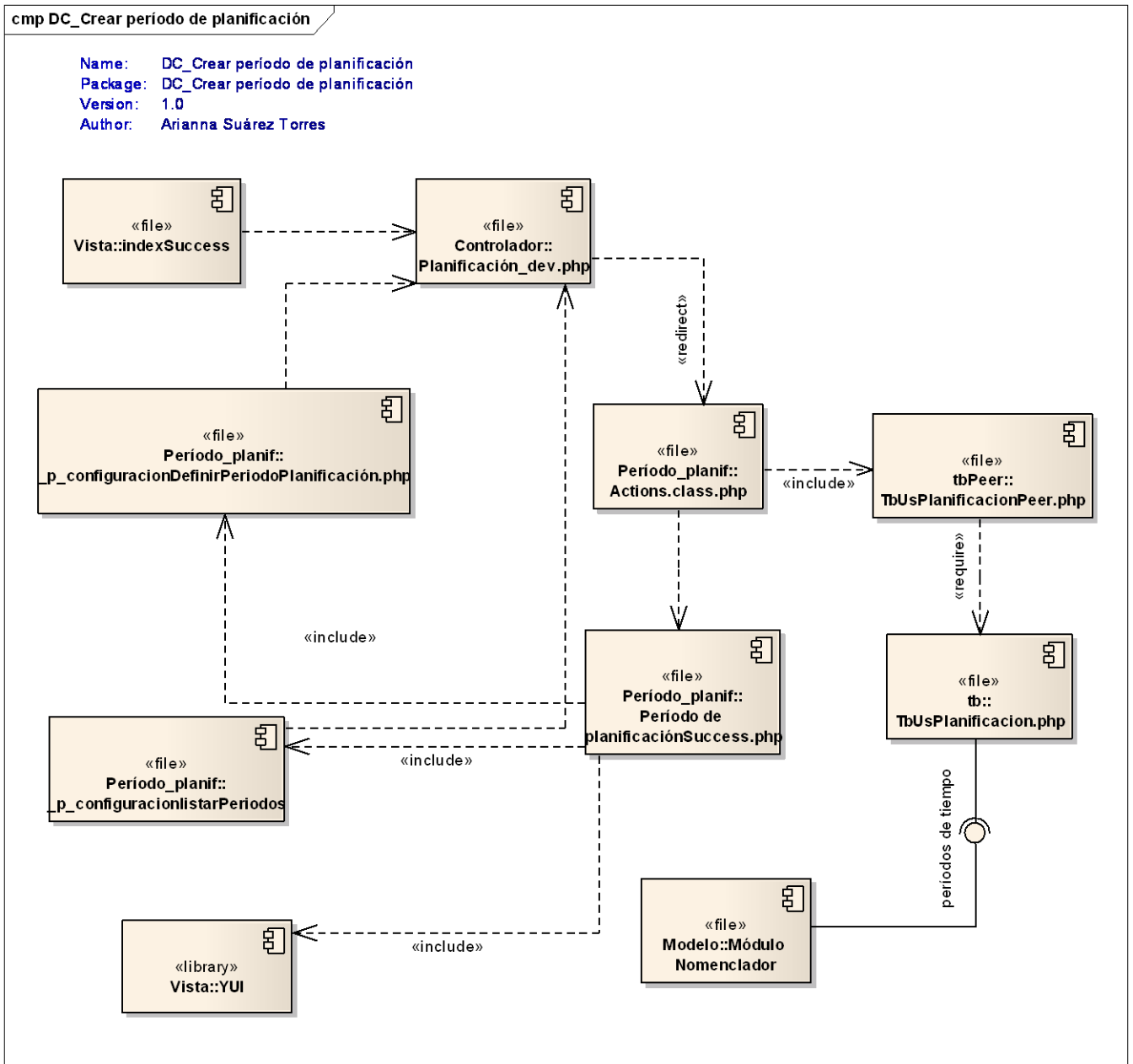


Figura 19. Diagrama de Componentes: DC_Crear período de planificación.

Capítulo 4: Implementación

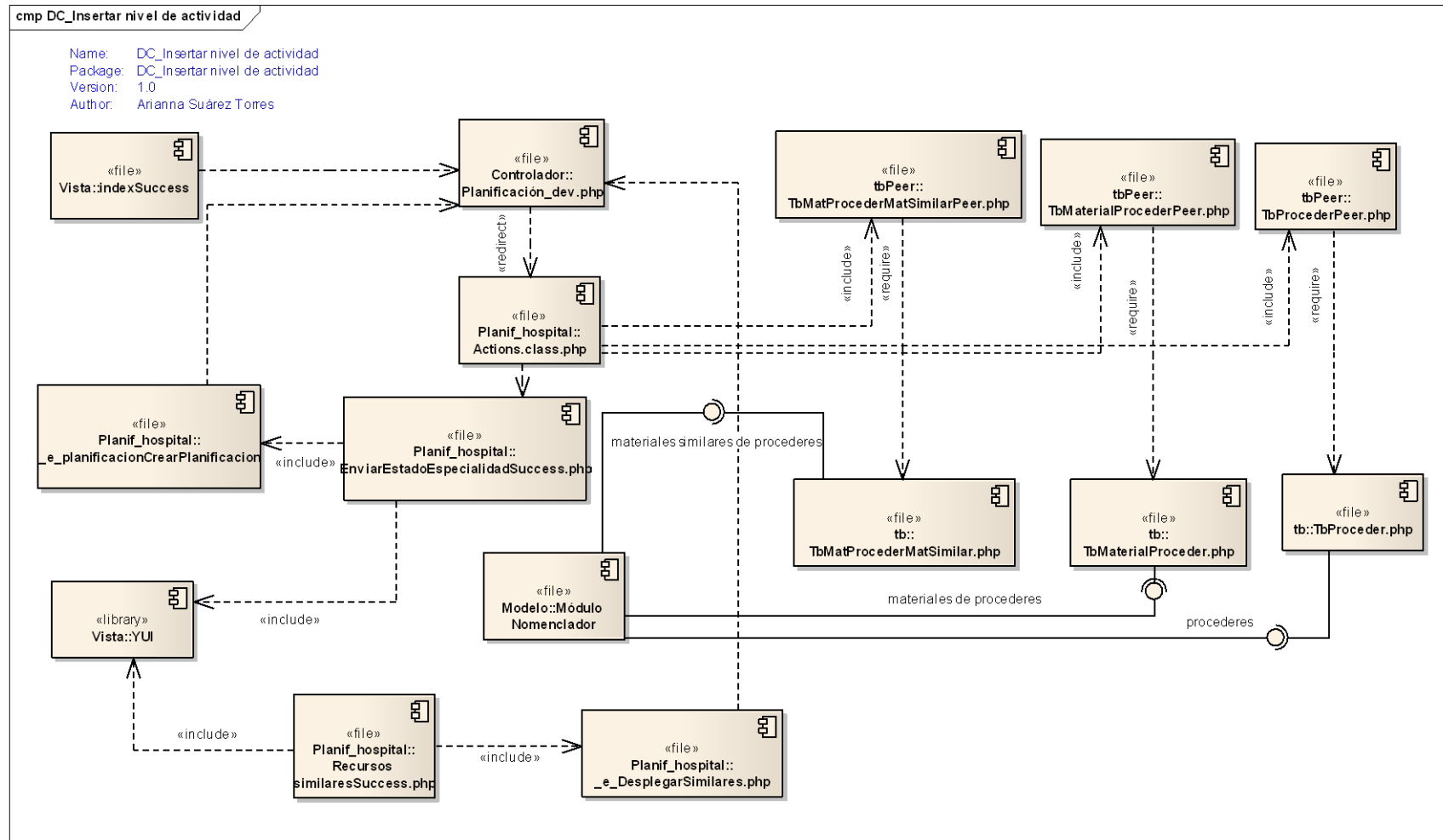


Figura 20. Diagrama de Componentes: DC_Insertar nivel de actividad.

Capítulo 4: Implementación

4.1.2. Diagrama de Despliegue

Un diagrama de despliegue muestra las relaciones físicas entre los componentes hardware y software en el sistema final, es decir, la configuración de los elementos de procesamiento en tiempo de ejecución y los componentes software.

Un diagrama de despliegue es un grafo de nodos unidos por conexiones de comunicación. Un nodo puede contener instancias de componentes software, objetos, procesos. Las instancias de componentes software pueden estar unidas por relaciones de dependencia, posiblemente a interfaces (ya que un componente puede tener más de una interfaz). [45]

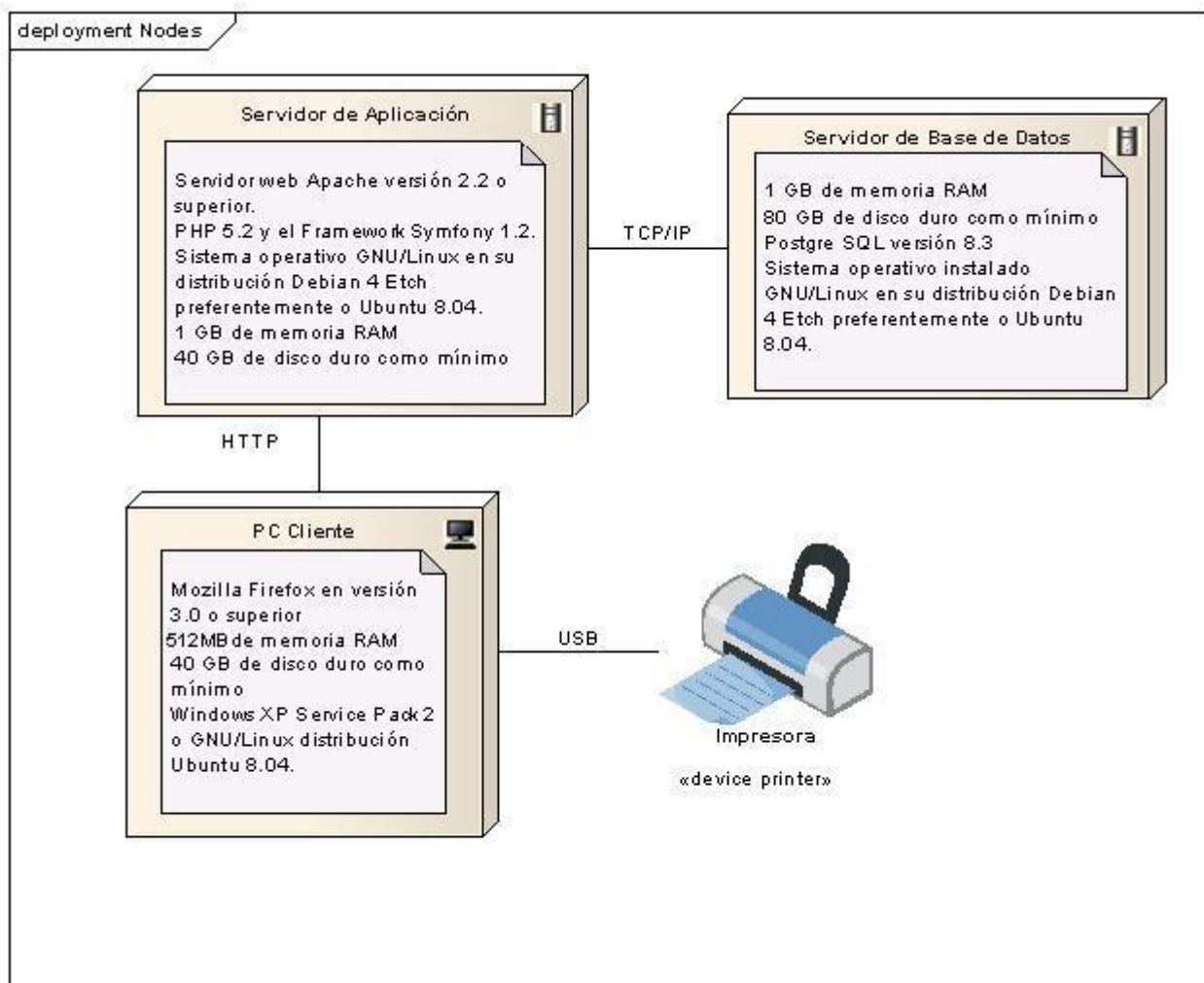


Figura 22. Diagrama de Despliegue.

Para el despliegue del sistema se proponen las siguientes condiciones:

Capítulo4: Implementación

Un Servidor de Aplicaciones que cuente con un ordenador Pentium IV, CPU 3.00GHz, con memoria RAM de 1GB, 40 GB de disco duro como mínimo y como software base Sistema operativo GNU/Linux en su distribución Debian 4 Etch preferentemente o Ubuntu 8.04, además del Apache2.2 o superior.

Un Servidor de Base de Datos con ordenador Pentium IV CPU 3.00GHz, con memoria RAM de 1GB, 80 GB de disco duro como mínimo y como software base GNU/Linux Distribución Debian 4 Etch preferentemente o Ubuntu 8.04 y como Sistema Gestor de Base de Datos PostgreSQL 8.3.

La PC Cliente mediante la cual los clientes podrán conectarse al sistema y operar en el mismo debe contar con el sistema operativo Windows XP Service Pack 2 o GNU/Linux Ubuntu 8.0; como navegador el Mozilla Firefox 3.0 o superior que le permita tener acceso a la aplicación y además, un ordenador Pentium IV con CPU 3.00GHz, memoria RAM de 512 MB y 10GB de disco duro.

La impresora que esté conectada a la PC Cliente permitirá al usuario imprimir los diferentes reportes que brinde el sistema.

Conclusiones

En este capítulo se mostraron los resultados obtenidos durante la etapa de implementación, donde los principales artefactos obtenidos fueron los Diagramas de Componentes y Despliegue, que representan los componentes del software y su distribución en hardware.

Recomendaciones

CONCLUSIONES

Con el desarrollo de este trabajo de Diploma se han obtenido los siguientes resultados:

- Las soluciones informáticas existentes para la planificación de materiales no constituyen una solución al problema resuelto. Las dos versiones anteriores, brindaron un grupo de funcionalidades que fueron reprogramadas pero constituyeron una base para asimilar el proceso de planificación en las unidades de salud.
- Las herramientas y tecnologías propuestas por el AT-SAS presentan características que facilitaron un buen desarrollo del sistema, las cuales en su mayoría constituyen software libres.
- La identificación de los requisitos contribuyó a que se lograra un entendimiento de las características que debía cumplir el sistema, permitiendo establecer un punto de partida para su confección.
- Con el módulo implementado para la planificación de materiales médicos se obtuvo el Módulo alas **BAP**Planificación v1.2, capaz de satisfacer las necesidades de las unidades de salud debido a las funcionalidades que le fueron incorporadas.

Recomendaciones

RECOMENDACIONES

Una vez concluido este trabajo, el equipo de desarrollo recomienda:

- Implementar nuevas funcionalidades que permitan la gestión y configuración del sistema de avisos, consultar la existencia de materiales en el almacén, crear la demanda con la existencia de materiales en el almacén en unidades de salud que no cuenten con el Módulo de Almacén, establecer similares durante la planificación y además, brindar reportes de planificaciones históricas.
- Estudiar el proceso de planificación en Venezuela para determinar en qué medida este sistema puede satisfacer sus necesidades.

Referencias Bibliográficas

REFERENCIAS BIBLIOGRÁFICAS

1. **Avila de la Cruz, Lisset del Carmen; Dominguez López, Rotceh y Bolmey Romero, Yurisnel.** *Sistema de Gestión de Información en el Proceso de Planificación de Materiales Gastables de Uso Médico del Ministerio de Salud Pública (MINSAP)*. Trabajo de Diploma para optar por el Título de Ingeniero Informático. Universidad de las Ciencias Informáticas. La Habana : s.n., 2008.
2. [En línea] [Citado el: 15 de Enero de 2009.] <http://www.monografias.com/trabajos15/sistema-nacional/sistema-nacional.shtml>.
3. [En línea] [Citado el: 15 de Enero de 2009.] http://www.sld.cu/sistema_de_salud/componentes.html.
4. Ídem a la Referencia 1.
5. Ídem a la Referencia 1.
6. Ídem a la Referencia 1.
7. Ídem a la Referencia 1.
8. **Macías, C.** Milestone. *Milestone*. [En línea] [Citado el: 2 de Marzo de 2009.] http://www.milestone.com.mx/articulos/uso_de_uml_en_aplicaciones_web.htm.
9. [En línea] [Citado el: 16 de Enero de 2009.] <http://hinetsistemas.com/servicios.htm?PHPSESSID=2ae263edd65717679c766d6c0a31020b>.
10. Informática 2009. *Informática 2009*. [En línea] [Citado el: 6 de Marzo de 2009.] http://www.informaticahabana.co.cu/evento_virtual/files/SWL11.pdf.
11. Ídem a la Referencia 8.
12. [En línea] [Citado el: 17 de Enero de 2009.] <http://www.sceu.frba.utn.edu.ar/Cursos/Informatica/Academias%20SAP/Archivos/academias%20M.pdf>.
13. [En línea] [Citado el: 17 de Enero de 2009.] http://www.sap.com/mexico/ecosystem/sap_professionals/modules/index.epx.
14. Qué es Internet. [En línea] [Citado el: 7 de febrero de 2008.] <http://www.civila.com/desenredada/que-es.html>.
15. Teleformación. *Teleformación*. [En línea] [Citado el: 17 de Enero de 2009.] <http://teleformacion.uci.cu/mod/resource/view.php?id=2059>.
16. **Corsi, Miguel.** Robin Good. Beneficios de las Aplicaciones basadas en Web. [En línea] [Citado el: 7 de diciembre de 2007.] http://www.masternewmedia.org/es/aplicaciones_web/temas_de_aplicaciones_web/Beneficios_De_Las_Aplicaciones_Basadas_En%20_Web_Y_El_Anuncio_De_Microsoft_De_La_Era_En_Vivo.htm.

Referencias Bibliográficas

17. Portal de Salud de Cuba. *Sistema Nacional de Salud*. [En línea] [Citado el: 9 de febrero de 2008.] http://www.sld.cu/sistema_de_salud/aspectos.html#misi3n.
18. **Arencibia Morales, Annia; Gómez Velázquez, Karel; González González, Leonardo**. *Centro de Control para el Sistema de Información para la Salud*. Trabajo de Diploma para optar por el Título de Ingeniero Informático. Universidad de las Ciencias Informáticas. La Habana, Cuba Junio 2007.
19. **Cinfonet**. [En línea] [Citado el: 7 de diciembre de 2007.] <http://hospedaje.cinfonet.com/?web=apache>.
20. **Presman, Roger S**. *Ingeniería de Software. Un Enfoque Práctico*. La Habana: Félix Varela, 2005. pág. 238. Vol. Parte 1.
21. **Legido Hernández, Marcos**. [En línea] 2005. http://www.wikilearning.com/caracteristicas_de_javascript-wkccp-3529-3.htm.
22. Herramientas Web para la enseñanza. *Protocolos de Comunicación*. [En línea] [Citado el: 7 de diciembre de 2007.] <http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html>
23. Instituto Nacional Estadísticas Informáticas. *Características del Modelo C/S*. [En línea] [Citado el: 8 de diciembre de 2007.] <http://www.inei.gob.pe/biblioiniepub/bancopub/inf/lib5038/carac.HTM>.
24. Instituto Nacional Estadísticas Informáticas. [En línea] [Citado el: 8 de diciembre de 2007.] <http://www.inei.gob.pe/web/metodologias/attach/lib616/INDE.HTM>.
25. LibrosWeb.es. [En línea] [Citado el: 15 de febrero de 2008.] http://www.librosweb.es/javascript/capitulo1/que_es_javascript.html. 20.
26. Wikilearning. [En línea] [Citado el: 15 de febrero de 2008.] http://www.wikilearning.com/caracteristicas_de_javascript-wkccp-3529-3.htm.
27. LibrosWeb.es. [En línea] [Citado el: 15 de febrero de 2008.] <http://www.librosweb.es/ajax/capitulo1.html>.
28. **Minnick, Chris; Valentine Chelsea**. *XHTML Serie Práctica*. Nueva York. Estados Unidos.2000. Pág. 7.
29. **Gallego Vázquez, José Antonio**. *Desarrollo Web con PHP y MySQL*. Madrid. España. Ediciones Anaya Multimedia. 2003. Pág. 20.
30. Consorcio World Wide Web (W3C). [En línea] [Citado el: 7 de mayo de 2008.] <http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>
31. **Minnick, Chris; Valentine, Chelsea**. *XHTML Serie Práctica*. Nueva York. Estados Unidos. 2000. Pág. 3, 4 y, 5.
32. [En línea] [Citado el: 4 de Diciembre de 2008.] <http://www.postgresql.org/about/press/presskit82.html.es>.
33. [En línea] [Citado el: 4 de Diciembre de 2008.]

Referencias Bibliográficas

http://www.lawebdelprogramador.com/news/mostrar_new.php?id=72&texto=PostgreSQL&n1=394636&n2=1&n3=0&n4=0&n5=0&n6=0&n7=0&n8=0&n9=0&n0=0.

34. **Eguíluz Pérez, Javier**. [En línea] [Citado el: 5 de diciembre de 2008.]
http://www.librosweb.es/css_avanzado/capitulo5/el_framework_yui.html
35. **Cardenas Thorlund, Manuel**. [En línea] [Citado el: 5 de diciembre de 2008.]
<http://nidea-soluciones.blogspot.com/2007/07/frameworks-para-rias.html>
36. **Cardenas Thorlund, Manuel**. [En línea] [Citado el: 4 de diciembre de 2008.]
<http://xynova.com/css/YuiLibrary.htm>
37. **Zaninotto, Francois; Potencier, Fabien**. [En línea] [Citado el: 4 de diciembre de 2008.]
http://www.librosweb.es/symfony/capitulo1/symfony_en_pocas_palabras.html
38. **Valerio, Adrián Anacleto**. [En línea] [Citado el: 4 de diciembre de 2008.]
http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=15
39. **Avila de la Cruz, Lisset del Carmen; Dominguez López, Rotceh; Bolmey Romero, Yurisel**. *Sistema de Gestión de Información en el Proceso de Planificación de Materiales Gastables de Uso Médico del Ministerio de Salud Pública (MINSAP)*. Trabajo de Diploma para optar por el Título de Ingeniero Informático. Universidad de las Ciencias Informáticas. La Habana, Cuba mayo 2008.
40. [En línea] [Citado el: 20 de Enero de 2009.] <http://www.sparxsystems.com.ar/new/>.
41. **Macías, C**. Millestone. *Millestone*. [En línea] [Citado el: 2 de Marzo de 2009.]
http://www.milestone.com.mx/articulos/uso_de_uml_en_aplicaciones_web.htm.
42. Teleformación. *Teleformación*. [En línea] [Citado el: 7 de Marzo de 2009.]
<http://teleformacion.uci.cu/mod/resource/view.php?id=13151>.
43. [En línea] [Citado el: 14 de Marzo de 2009.]
http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=96&Itemid=297.
44. Teleformación. *Teleformación*. [En línea] [Citado el: 24 de Marzo de 2009.]
<http://teleformacion.uci.cu/mod/resource/view.php?id=14094>.
45. [En línea] [Citado el: 25 de Marzo de 2009.] <http://tvdi.det.uvigo.es/~avilas/UML/node50.html>.

Bibliografía

BIBLIOGRAFÍA

- **Arencibia Morales, Annia; Gómez Velázquez, Karel; González González, Leonardo.** *Centro de Control para el Sistema de Información para la Salud.* Trabajo de Diploma para optar por el Título de Ingeniero Informático. Universidad de las Ciencias Informáticas. La Habana, Cuba Junio 2007.
- **Avila de la Cruz, Lisset del Carmen; Dominguez López, Rotceh y Bolmey Romero, Yurisnel.** *Sistema de Gestión de Información en el Proceso de Planificación de Materiales Gastables de Uso Médico del Ministerio de Salud Pública (MINSAP).* Trabajo de Diploma para optar por el Título de Ingeniero Informático. Universidad de las Ciencias Informáticas. La Habana : s.n., 2008.
- **Cardenas Thorlund, Manuel.** [En línea] [Citado el: 4 de diciembre de 2008.]
<http://xynova.com/css/YuiLibrary.htm>
- **Cardenas Thorlund, Manuel.** [En línea] [Citado el: 5 de diciembre de 2008.] <http://nidea-soluciones.blogspot.com/2007/07/frameworks-para-rias.html>
- Centro Virtual de Documentación Regulatoria. [En línea] [Citado el: 19 de Enero de 2009.]
http://www.mspas.gob.sv/regulacion/pdf/manual/Manual_estimacion_necesidades_medicamentos_e_insumos.pdf.
- **Cinfontet.** [En línea] [Citado el: 7 de diciembre de 2007.]
<http://hospedaje.cinfontet.com/?web=apache>.
- Consorcio World Wide Web (W3C). [En línea] [Citado el: 7 de mayo de 2008.]
<http://www.w3c.es/divulgacion/guiasbreves/HojasEstilo>
- Corporación Universitaria Autónoma del Cauca. *Corporación Universitaria Autónoma del Cauca.* [En línea] [Citado el: 4 de abril de 2009.]
www.uniautonoma.edu.co/docentes/hcordoba/SIGRequerimientos.ppt .
- **Corsi, Miguel.** Robin Good. Beneficios de las Aplicaciones basadas en Web. [En línea][Citado el: 7 de diciembre de 2007.]
http://www.masternewmedia.org/es/aplicaciones_web/temas_de_aplicaciones_web/Beneficios_De_Las_Aplicaciones_Basadas_En%20_Web_Y_El_Anuncio_De_Microsoft_De_La_Era_En_Vivo.htm.
- **Eguíluz Pérez, Javier.** [En línea] [Citado el: 5 de diciembre de 2008.]
http://www.librosweb.es/css_avanzado/capitulo5/el_framework_yui.html
- E-journal. *E-journal.* [En línea] [Citado el: 10 de abril de 2009.]
<http://www.ejournal.unam.mx/cys/vol07-04/CYS07402.pdf> .
- [En línea] [Citado el: 12 de Febrero de 2009.]

Bibliografía

- http://www.vico.org/TRAD_obert/TRAD_CasosDeUso.html.
- [En línea] [Citado el: 14 de Enero de 2009.]
<http://www.cub.ops-oms.org/boletinesEconom%C3%ADa%20y%20Salud-DMS2007%20Dr.Castro-MINSAP>
 - [En línea] [Citado el: 14 de Marzo de 2009.]
http://merinde.rinde.gob.ve/index.php?option=com_content&task=view&id=96&Itemid=297.
 - [En línea] [Citado el: 15 de Enero de 2009.]
<http://www.monografias.com/trabajos15/sistema-nacional/sistema-nacional.shtml>.
 - [En línea] [Citado el: 15 de Enero de 2009.]
http://www.sld.cu/sistema_de_salud/componentes.html.
 - [En línea] [Citado el: 16 de Enero de 2009.]
<http://hinetsistemas.com/servicios.htm?PHPSESSID=2ae263edd65717679c766d6c0a31020b>.
 - [En línea] [Citado el: 17 de Enero de 2009.]
http://www.sap.com/mexico/ecosystem/sap_professionals/modules/index.epx.
 - [En línea] [Citado el: 17 de Enero de 2009.]
<http://www.sceu.frba.utn.edu.ar/Cursos/Informatica/Academias%20SAP/Archivos/academias%20MM.pdf> .
 - [En línea] [Citado el: 20 de Enero de 2009.] <http://www.sparxsystems.com.ar/new/>.
 - [En línea] [Citado el: 22 de Marzo de 2009.]
<http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=mysqlphpeclipsepdt>.
 - [En línea] [Citado el: 23 de Enero de 2009.]
<http://www.enriquedans.com/2006/05/openbravo-un-erp-de-codigo-abierto.html>.
 - [En línea] [Citado el: 23 de Enero de 2009.]
http://www.fce.unal.edu.co/uifce/newuifce/conf/documentos/open_bravo.pdf.
 - [En línea] [Citado el: 25 de Marzo de 2009.] <http://tvdi.det.uvigo.es/~avilas/UML/node50.html>.
 - [En línea] [Citado el: 4 de Diciembre de 2008.]
http://www.lawebdelprogramador.com/news/mostrar_new.php?id=72&texto=PostgreSQL&n1=394636&n2=1&n3=0&n4=0&n5=0&n6=0&n7=0&n8=0&n9=0&n0=0.
 - [En línea] [Citado el: 4 de Diciembre de 2008.]
<http://www.postgresql.org/about/press/presskit82.html.es>.
 - [En línea] [Citado el: 6 de Marzo de 2009.]
<http://desarrolloerp.blogspot.com/2009/02/el-erp-cubano-tiene-nombre-cedrux.html>.
 - [En línea] 3 de Marzo de 2009.
<http://personas.confidare.cl/blog/wp-content/uploads/2006/12/symfony-proyector.pdf>.

Bibliografía

- **Freijó, Issac.** *Software Libre e Innovación: la Realidad y el Futuro ERP* [En línea] [Citado el: 29 Mayo 1995]. Disponible en: \\10.0.0.22\Documentacion\Curso de Verano\2008\ ISAAC - ERPs.pdf. También disponible en: ftp://10.0.0.22/documentacion/Curso 20de Verano/2008/ ISAAC - ERPs.pdf.
- Fundamentos de Ingeniería de Software. *Fundamentos de Ingeniería de Software*. [En línea] [Citado el: 20 de Febrero de 2009.]
<http://www.inf.utfsm.cl/~visconti/ili236/Documentos/15-Implementacion.pdf>.
- **Gallego Vázquez, José Antonio.** Desarrollo Web con PHP y MySQL. Madrid. España. Ediciones Anaya Multimedia. 2003. Pág. 20.
- Herramientas Web para la enseñanza. Protocolos de Comunicación. [En línea] [Citado el: 7 de diciembre de 2007.] <http://neo.lcc.uma.es/evirtual/cdd/tutorial/aplicacion/cliente-servidor.html>
- Infomed. *Infomed*. [En línea] [Citado el: 14 de Enero de 2009.]
http://www.sld.cu/sistema_de_salud/aspectos.html.
- Informática 2009. *Informática 2009*. [En línea] [Citado el: 6 de Marzo de 2009.]
http://www.informaticahabana.co.cu/evento_virtual/files/SWL11.pdf.
- Instituto Nacional Estadísticas Informáticas. [En línea] [Citado el: 8 de diciembre de 2007.]
<http://www.inei.gob.pe/web/metodologias/attach/lib616/INDE.HTM>.
- Instituto Nacional Estadísticas Informáticas. Características del Modelo C/S. [En línea] [Citado el: 8 de diciembre de 2007.] <http://www.inei.gob.pe/biblioiniepub/bancopub/inf/lib5038/carac.HTM>.
- Laboratorio de Televisión Digital Interactiva. *Laboratorio de Televisión Digital Interactiva*. [En línea] [Citado el: 8 de Marzo de 2009.] <http://tvdi.det.uvigo.es/~avilas/UML/node25.html>.
- **Legido Hernández, Marcos.** [En línea] 2005.
http://www.wikilearning.com/caracteristicas_de_javascript-wkccp-3529-3.htm.
- LibrosWeb.es. [En línea] [Citado el: 15 de febrero de 2008.]
http://www.librosweb.es/javascript/capitulo1/que_es_javascript.html. 20.
- LibrosWeb.es. [En línea] [Citado el: 15 de febrero de 2008.]
<http://www.librosweb.es/ajax/capitulo1.html>.
- **Macías, C.** Millestone. *Millestone*. [En línea] [Citado el: 2 de Marzo de 2009.]
http://www.milestone.com.mx/articulos/uso_de_uml_en_aplicaciones_web.htm.
- Maestros del Web. *Maestros del Web*. [En línea] [Citado el: 15 de Mayo de 2009.]
www.ops.org.bo/multimedia/cd/2008/SRI_6_2008/recursos/documentos/12_Resumen.ppt.
- Maestros del web. *Maestros del web*. [En línea] [Citado el: 7 de Marzo de 2009.]

Bibliografía

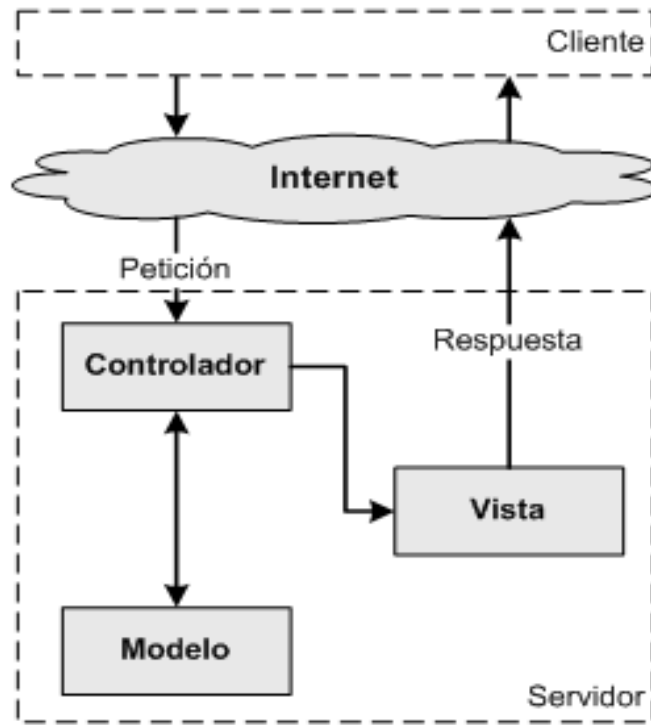
- <http://www.maestrosdelweb.com/editorial/el-framework-symfony-una-introduccion-practica-ii-parte/>.
- Maestros del Web. *Maestros del Web*. [En línea] 10 de Marzo de 2004. [Citado el: 8 de Febrero de 2009.] <http://www.maestrosdelweb.com/editorial/inyecsql>.
 - Ministerio de la Informática y las Comunicaciones. *Ministerio de la Informática y las Comunicaciones*. [En línea] [Citado el: 16 de Enero de 2009.] <http://www.mic.gov.cu/hinfosoc.aspx>.
 - **Minnick, Chris; Valentine Chelsea**. XHTML Serie Práctica. Nueva York. Estados Unidos. 2000. Pág. 7.
 - **Minnick, Chris; Valentine, Chelsea**. XHTML Serie Práctica. Nueva York. Estados Unidos. 2000. Pág. 3, 4 y, 5.
 - Portal de Salud de Cuba. *Sistema Nacional de Salud*. [En línea] [Citado el: 9 de febrero de 2008.] http://www.sld.cu/sistema_de_salud/aspectos.html#mision.
 - **Presman, Roger S**. Ingeniería de Software. Un Enfoque Práctico. La Habana: Félix Varela, 2005. pág. 238. Vol. Parte 1.
 - Qué es Internet. [En línea] [Citado el: 7 de febrero de 2008.] <http://www.civila.com/desenredada/que-es.html>.
 - Revistas médicas cubanas. *Revistas médicas cubanas*. [En línea] [Citado el: 16 de Enero de 2009.] http://bvs.sld.cu/revistas/spu/vol32_3_06/spu15306.htm.
 - Servicios Informáticos Profesionales. Servicios Informáticos Profesionales. [En línea] [Citado el: 20 de Marzo de 2009.] <http://www.dualsolution.com/noticias-informaticas/noticias-Software-40.html>.
 - Sistema de Bibliotecas. *Sistema de Bibliotecas*. [En línea] [Citado el: 16 de Febrero de 2009.] http://sisbib.unmsm.edu.pe/bibvirtualdata/Tesis/Basic/mendoza_nj/Cap5.pdf.
 - Teleformación. *Teleformación*. [En línea] [Citado el: 24 de Marzo de 2009.] <http://teleformacion.uci.cu/mod/resource/view.php?id=14094>.
 - Teleformación. *Teleformación*. [En línea] [Citado el: 26 de Enero de 2009.] <http://teleformacion.uci.cu/mod/resource/view.php?id=11569>.
 - Teleformación. *Teleformación*. [En línea] [Citado el: 7 de Marzo de 2009.] <http://teleformacion.uci.cu/mod/resource/view.php?id=13151>.
 - **Valerio, Adrián Anacleto**. [En línea] [Citado el: 4 de diciembre de 2008.] http://www.epidataconsulting.com/tikiwiki/tiki-read_article.php?articleId=15
 - Versvs. *Versvs*. [En línea] [Citado el: 5 de Febrero de 2009.] <http://www.versvs.net/anotacion/que-es-un-erp-enterprise-resource-planning-linux>.

Bibliografía

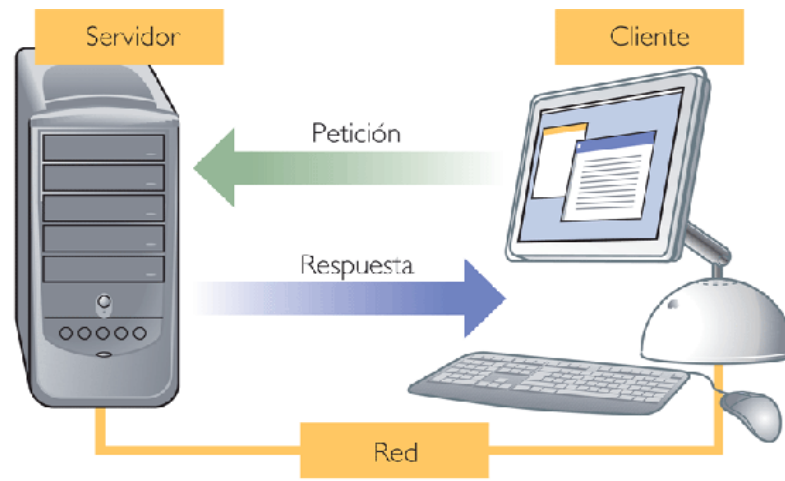
- Wikilearning. [En línea] [Citado el: 15 de febrero de 2008.]
http://www.wikilearning.com/caracteristicas_de_javascript-wkccp-3529-3.htm.
- willydev. *willydev*. [En línea] [Citado el: 20 de Marzo de 2009.]
<http://www.willydev.net/descargas/prev/Prolego.pdf>.
- **Zaninotto, Francois; Potencier, Fabien**. [En línea] [Citado el: 4 de diciembre de 2008.]
http://www.librosweb.es/symfony/capitulo1/symfony_en_pocas_palabras.html

ANEXOS

Anexo 1. Patrón de arquitectura: Modelo Vista Controlador.



Anexo 2. Arquitectura: Cliente Servidor.



Anexo 3. Fases y flujos de trabajos de la Metodología RUP.

