

Universidad de las Ciencias Informáticas
Facultad 6



Título: Análisis y Diseño del Sistema de Ayuda Médica para la
Atención de las Dislipoproteinemias

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

Autores: Herlyn Rodríguez Alvarez.
Yasel Villavicencio Isaic.

Tutores: Ing. Ania Bermúdez Peña
Ing. Ivalú Miranda Rodríguez

Co-tutor: Dr. Alfredo Nasiff Hadad

Ciudad de La Habana 21 de junio, 2008.

DECLARACIÓN DE AUTORÍA

DECLARACIÓN DE AUDITORÍA

Declaramos ser autores de la presente tesis y reconocemos 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 -- días del mes de -- del año --.

Yasel Villavicencio Isaic

Firma del Autor

Herlyn Rodríguez Álvarez.

Firma del Autor

Ing. Ania Bermúdez Peña.

Firma del Tutor

Ing. Ivalú Miranda Rodríguez.

Firma del Tutor

DATOS DE CONTACTOS

DATOS DE CONTACTOS

Ing. Ania Bermúdez Peña (email: abermudezp@uci.cu)

Profesora graduada de Ingeniería en Informática. Ha impartido la asignatura de Ingeniería de Software, Gestión de Software y Base de Datos. Posee categoría docente de Adiestrado. Actualmente se desempeña como profesora de Gestión de Software y Base de Datos, en la Facultad 6 de la Universidad de las Ciencias Informáticas (UCI).

Ing. Ivalú Miranda (email: imiranda@uci.cu)

Ingeniera en Ciencias Informáticas graduada en la UCI, Actualmente presta servicio como profesora de la universidad en las asignaturas Teleinformática I y II en la Facultad 6 de la Universidad de las Ciencias Informáticas (UCI). Posee categoría docente Instructor. Cursa la maestría de Las tecnologías en los procesos educativos.

DEDICATORIA

AGRADECIMIENTOS

Durante la realización de este trabajo influyeron determinadas personas que de una forma u otra permitieron que esta investigación se realizara con éxitos. No quisiera pasar por alto este momento sin antes brindarles mis más sinceros agradecimientos.

Al amor de mi vida Rocío del Carmen Mendoza González por ser la estrella que ilumina mi andar, gracias por todo tu apoyo y haberme dado fuerzas para salir siempre adelante en los buenos y en los malos momentos.

A mis abuelos, mis padres, mi hermano, mi tío, a mi familia le agradezco de todo corazón, haber sabido forjar en mí los mejores valores, a quienes debo cuanto soy, gracias por ser mis guías.

A mi compañero de tesis y amigo Herlyn, a mis amistades y compañeros de aula gracias por tanto apoyo.

A mis suegros, a Chaly, a Leo, a Ramón gracias por preocuparse siempre por mí, por brindarme tanto apoyo en mi carrera y en últimos años de mi vida.

A mis profesores, tutores que siempre me apoyaron.

Yasel

Han sido muchas las personas que a lo largo de estos 5 años me han brindado amistad, cariño y apoyo incondicional. Por eso no quisiera pasar por alto este momento para hacerles llegar mi más profundo agradecimiento.

A toda mi familia, que de una forma u otra siempre me ha ofrecido su amor y cariño en los momentos más difíciles.

A Mercedes y Jorge, por acogerme como un hijo.

A mis Amigos Maikel Muñoz, Nelvis Borges y Yasel Villavicencio, por ayudarme siempre que lo he necesitado.

A mis profesores y compañeros de aula, por haberme permitido compartir con ellos momentos inolvidables.

A mi tutora Anía Bermúdez e Ivalú Miranda, por haber sido comprensiva y paciente, por su ayuda sin límites.

A María Isabel Rodríguez, por su cariño, ayuda y comprensión.

A la revolución por haberme dado la oportunidad de estudiar una carrera universitaria.

Herlyn

DEDICATORIA

DEDICATORIA

A quienes me enseñan cada día el valor indestructible de la vida A mis abuelos, mis padres, mi hermano, mi tío.

A mi novia Rocío por todo su amor, dedicación y apoyo incondicional en estos 5 años en la universidad.

Yasel

A mis abuelos Alicia, José Ramón y mi mamá Idania, por dedicarme cada segundo de sus vidas, por su paciencia, por su Amor, por estar siempre allí ayudándome a poner "los pies sobre la tierra", por la confianza que han depositado en mí la cual me ha hecho crecer y triunfar.

A mi hermana linda que quiero con la vida.

A mis tías Carmen, Aleida y Librada, por todo su cariño y apoyo.

A mi novia Dayli Figueroa, por su amor, por apoyarme sin límites y estar siempre a mi lado en los momentos que más lo necesitaba.

Herlyn

RESUMEN

El punto de partida de esta investigación está dado por un estudio crítico y detallado de la aplicación Web Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias (SAMAD), con el objetivo de realizar el refinamiento del análisis y diseño de un software eficaz que permita obtener el diagnóstico y tratamiento de las dislipoproteinemias y a su vez pueda ser instalado en los servidores de Infomed, con todo los requisitos de seguridad que están establecidos.

En la presente investigación se hace el refinamiento del análisis y diseño de SAMAD en la cual van a estar presente un grupo importante de funcionalidades que permitirá a los médicos registrados en los servidores de Infomed la posibilidad de acceder a la aplicación Web permitiendo gestionar toda la información relacionada con la atención a pacientes, además de realizar todos los cálculos necesarios en las diferentes consultas. Se rediseña la aplicación Web teniendo en cuenta herramientas informáticas de software libre. El Sistema debe desplegarse siguiendo la arquitectura orientada por Softel en el Documento de Arquitectura del Software para el Ministerio de Salud Pública (MINSAP).

PALABRAS CLAVES:

Software, Dislipoproteinemias, Web, Refinamiento.

ÍNDICE

ÍNDICE

AGRADECIMIENTOS	I
DEDICATORIA.....	II
RESUMEN.....	III
ÍNDICE	IV
INTRODUCCIÓN.....	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA.....	5
1.1 APLICACIONES DE SOFTWARE MÉDICOS EN EL MUNDO	5
1.2 APLICACIONES Y SOFTWARE MÉDICOS EN CUBA	5
1.3 ¿POR QUÉ DESARROLLAR EL REFINAMIENTO DEL ANÁLISIS Y DISEÑO DE SAMAD	7
1.4 ARQUITECTURA DE SOFTWARE EN EL SISTEMA NACIONAL DE SALUD.	9
1.5 ANÁLISIS DE LAS HERRAMIENTAS INFORMÁTICAS SELECCIONADAS.	10
CONCLUSIONES.....	17
CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA.....	18
2.1 OBJETIVOS DE LA ORGANIZACIÓN	18
2.2 MODELO DEL NEGOCIO ACTUAL.....	19
2.3 PROPUESTA DEL SISTEMA.....	26
2.3.1.1 REQUISITOS FUNCIONALES.....	27
2.3.1.2 REQUISITOS NO FUNCIONALES.....	28
2.3.2.1 ACTORES DEL SISTEMA	30
2.3.2.2 DIAGRAMA DEL SISTEMA	31
2.3.2.3 DIAGRAMA DE CASOS DE USO DEL SISTEMA	31
2.3.2.4 DESCRIPCIONES DE LOS CASOS DE USO DEL SISTEMA.....	33
2.3.2.5 MATRIZ DE TRAZABILIDAD ENTRE REQUERIMIENTOS Y CUS.....	50
CONCLUSIONES.....	51
CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA	52
3.1 DEFINICIÓN DE LA ARQUITECTURA	52
3.2 MODELO DE ANÁLISIS.....	57
3.3 MODELO DE DISEÑO	60
3.4 DISEÑO DE LA BASE DE DATOS	83
3.5 MODELO DE DESPLIEGUE.....	86
CONCLUSIONES.....	87
CONCLUSIONES GENERALES.....	88
RECOMENDACIONES.....	89
REFERENCIAS BIBLIOGRAFICAS.....	90

ÍNDICE

BIBLIOGRAFIA.....	91
ANEXOS.	92
GLOSARIO DE TÉRMINOS	111

INTRODUCCIÒN

El Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias (SAMAD) está relacionado con la atención a las Dislipoproteinemias que son un grupo de enfermedades que *“están caracterizadas por el aumento de la concentración de una, varias o todas las fracciones lipídicas del plasma. Los principales lípidos plasmáticos son el colesterol (hipercolesterolemia) y los triglicéridos (hipertrigliceridemia), también otros circulando en sangre a través de las lipoproteínas.”* (1)

Las dislipoproteinemias son enfermedades de difícil diagnóstico cuando se trata de discriminar entre formas secundarias y primarias. Los profesionales de la salud afines al tema, se enfrentan a una gran variedad de situaciones que conllevan a diferentes soluciones teniendo en cuenta las particularidades de cada paciente; es importante que se evalúen cuidadosamente sus factores de riesgos, grupo de riesgos al que pertenece, causas secundarias, antecedentes patológicos personales y familiares, entre otras consideraciones médicas importantes que permitan la indicación de tratamientos adecuados según las características individuales en cada paciente.

Actualmente se conoce que con *las modificaciones favorables de los valores de las lipoproteínas plasmáticas, mediante acciones no farmacológicas y farmacológicas, se puede disminuir la incidencia de cardiopatía isquémica, enfermedades cerebrovasculares y enfermedad vascular de miembros inferiores, lo que posibilitaría no sólo la prevención sino incluso la regresión de la aterosclerosis.* (1)

En Cuba se trabaja incansablemente en la búsqueda e implementación de vías y alternativas que contribuyan al diagnóstico y tratamiento de las dislipoproteinemias, además de ofrecer una mejor preparación a los especialistas en esta rama de la salud. En torno al tema en el curso 2006-2007, la Universidad de las Ciencias Informáticas en conjunto con el hospital "Hermanos Ameijeiras" comenzaron el desarrollo de una aplicación Web denominada SAMAD, la cual tenía como objetivos fundamentales: automatizar el proceso de diagnóstico y tratamiento de la dislipoproteinemias, brindar información a la población de temas relacionados con este padecimiento y homogeneizar los conocimientos correspondientes a esta enfermedad en la comunidad médica cubana. Esta aplicación Web no quedó totalmente concluida, pues no se terminó el ciclo de

INTRODUCCION

consultas que se le deben realizar a un paciente que padezca esta enfermedad. La aplicación anterior no tenía una arquitectura definida lo que trajo como consecuencia que no se pudiera desplegar en los servidores de Infomed. El sistema de seguridad establecido no garantizaba la confidencialidad de la información. Este curso se hace necesario incluir al sistema otras funcionalidades que son importantes para el diagnóstico y tratamiento de las dislipoproteinemias. Se desea definir una arquitectura que permita que la aplicación Web cumpla con los estándares establecidos por Softel, además de establecer un sistema de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (SAAA)

Con el objetivo de promover el control y la prevención de enfermedades cardíacas y cerebrovasculares, disminuir la mortalidad por estas causas, así como educar a la población hacia estilos de vida más saludables, se han realizado, tanto a nivel internacional como nacional, numerosos estudios. Estas investigaciones han caracterizado como factores de riesgos de este tipo de enfermedades, a los altos niveles de colesterol total, triglicéridos y de lipoproteínas de baja densidad (LDL), junto con niveles bajos de lipoproteínas de alta densidad (HDL).

Por lo antes expuesto se plantea el siguiente **Problema científico** de la investigación **¿Cómo traducir los nuevos requerimientos identificados en SAMAD en elementos que puedan ser implementados?**

El objeto de estudio se centra en **El Proceso de desarrollo de software en el diagnóstico y tratamiento de las Dislipoproteinemias**. El **Campo de Acción** se define como **El proceso de desarrollo de software en el refinamiento del análisis y diseño de SAMAD**.

Los acelerados cambios en las Tecnología de la Información y las Comunicaciones (TIC) propician a la comunidad científica médica la creación de herramientas sofisticadas que difieren de las ya existentes.

A partir del problema científico planteado y de las ventajas que ofrece el proceso de desarrollo del software, se plantea como **objetivo general de la investigación: Desarrollar el refinamiento del análisis y diseño del Sistema de Ayuda Médica para**

INTRODUCCION

la **Atención de las Dislipoproteinemias**. Quedando desglosado en los siguientes **objetivos específicos**:

- **Refinar el análisis y diseño de la aplicación Web anterior.**
- **Desarrollar el análisis y diseño del Módulo de Atención de Pacientes en seguimiento.** (Diagnóstico, tratamiento farmacológico y tratamiento no farmacológico)
- **Desarrollar el análisis y diseño del Módulo de Cálculos auxiliares.**(Cálculo de Peso Ideal, Cálculo de Dietas según calorías)

Para lograr los objetivos se enuncian las siguientes **tareas**:

- **Análisis crítico de SAMAD.**
- **Investigación y selección del framework que permita una mejor integración con la arquitectura establecida por Softel.**
- **Investigación e inserción de la arquitectura establecida por Softel.**
- **Determinación de los requisitos funcionales y no funcionales de la aplicación Web.**
- **Descripción de los casos de uso del sistema para la aplicación Web que se propone.**
- **Diseño del diagrama de caso de uso del sistema de la aplicación Web.**
- **Realización de los diagramas de clases del análisis y clases del diseño de la aplicación Web.**
- **Realización de los prototipos no funcionales de la aplicación Web.**

La investigación consta de 3 capítulos:

En el capítulo 1: Fundamentación Teórica: se expone un estado del arte del tema tratado en la aplicación a nivel Internacional y Nacional, se hace un análisis crítico en la propuesta de por qué realizar el refinamiento del análisis y diseño de SAMAD. Se aborda el tema de la arquitectura establecida por Softel para la inserción del software en el

INTRODUCCION

Sistema de Salud Cubano. Se acuerda mediante qué tecnologías y herramientas informáticas se realizará esta nueva aplicación Web.

En el capítulo 2: Características del sistema SAMAD: se lleva a cabo un proceso de desarrollo del software donde se exponen los objetivos estratégicos de la organización; la modelación del negocio para lo cual se identifican y describen los actores, trabajadores y casos de uso del negocio, se realizan los diagramas de actividades correspondientes a cada caso de uso del negocio, y el diagrama del modelo de objetos. Se plantean los requisitos funcionales y no funcionales de la aplicación a desarrollar y se modela la misma en términos de casos de uso de sistema.

En el Capítulo 3: EL flujo de trabajo de análisis y diseño tiene un papel protagónico en la fase de elaboración. Su principal objetivo es traducir los requisitos a una especificación que describa cómo implementar el sistema. Con el análisis se obtiene una visión del sistema enmarcada en el qué hace, de modo que sólo se interesa por los requisitos funcionales y el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, o sea el cómo cumple el sistema sus objetivos. El resultado final más importante de este flujo de trabajo será el modelo del diseño.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Se expone un estado del arte del tema tratado en la aplicación a nivel Internacional y Nacional, se hace un análisis crítico en la propuesta de por qué realizar el refinamiento del análisis y diseño de SAMAD. Se aborda el tema de la arquitectura establecida por Softel para la inserción del software en el Sistema de Salud Cubano. Se acuerda mediante qué tecnologías y herramientas informáticas se realizará esta nueva aplicación Web.

1.1 APLICACIONES DE SOFTWARE MÉDICOS EN EL MUNDO

En el mundo existen muchos software que ayudan a prevenir el padecimiento de enfermedades cardíacas y cerebro vasculares. Donde la mayoría se limitan a recomendar los tipos de dietas que debe mantener una persona de acuerdo con su edad y su peso, brindan poca información acerca de la enfermedad, y otros también ofrecen la posibilidad de realizar algunos cálculos como el factor de riesgo de la enfermedad. A continuación se relacionan algunos ejemplos de aplicaciones a nivel mundial que tienen el objetivo de prevenir y tratar las dislipoproteinemias al igual que SAMAD.

➤ **Zonadiet**

Esta aplicación está referida a la alimentación, la nutrición y la salud. Explica en lenguaje cotidiano conceptos básicos que se deben tener en cuenta para la elección de alimentos, en que forma combinarlos y las cantidades. Esta aplicación es mayormente informativa.

➤ **GlucoGestor.**

GlucoGestor es una aplicación Web que ayuda a tener un registro electrónico de las medidas de glucosa en sangre que vaya tomando a lo largo de vida un paciente diabético. Mediante la anotación de sus glucemias en GlucoGestor se puede tener una relación de todas las glucemias que se pueden visualizar tanto en formato gráfico como en formato impreso.

1.2 APLICACIONES Y SOFTWARE MÉDICOS EN CUBA

Un aspecto significativo del programa de Informática Médica (IM) en Cuba que no puede pasar por alto es el desarrollo de la Red de Información de las Ciencias Médicas (INFOMED), la cual posibilita el intercambio entre gran cantidad de centros de salud e investigación de Cuba con el resto del mundo. INFOMED presenta un número importante de revistas, libros, publicaciones y

materiales científicos de gran utilidad para sus usuarios, enlaces a sitios Web nacionales y significativas instituciones internacionales.

En el país existen numerosos centros de investigación que dedican parte de su trabajo a crear software que sirven de apoyo para la detección, diagnóstico y tratamiento de enfermedades, herramientas para el aprendizaje de la medicina e importantes bases de datos médicas con amplios volúmenes de datos.

Algunos de estos centros son: el Instituto Central de Investigaciones Digitales (ICID), creador de LIPICID, que cuenta con un número importante de software de la más alta tecnología, el Centro de Estudios Genéticos y Biotecnología (CIGB), el Centro de Neurociencias de Cuba (CNC), el Centro de Cibernética Aplicada a la Medicina, el Centro de Desarrollo Informático para la Salud Pública, el Centro Nacional de Bioinformática (Bioinfo), entre otros.

Como resultado de los proyectos de IM cubanos se han desarrollado muchos software médicos, dentro de ellos se citan:

➤ **Hipertencid.**

Un sistema experto para la atención, prevención, diagnóstico, pronóstico y control de la hipertensión arterial, desarrollado por el Instituto Central de Investigación Digital y el Grupo Multidisciplinario de Hipertensión Arterial del Hospital "Hermanos Ameijeiras". Propicia que se ofrezca una atención por otros especialistas similar a la del conjunto de expertos del Hospital "Hermanos Ameijeiras". Se informa de la prevención y control de otros factores de riesgos, así como otras enfermedades crónicas no transmisibles. Se comprueba mediante la validación del sistema que el mismo es un método eficaz en el tratamiento ambulatorio del paciente hipertenso.

➤ **Lipicid.**

LIPICID es una aplicación Windows para el diagnóstico, evaluación y tratamiento de las dislipoproteinemia. Toda la información sobre los pacientes manipulada por LIPICID se almacena en bases de datos tipo FoxPro. LIPICID para su trabajo se guía por la experiencia tomada de la consulta de Dislipoproteinemia y Aterosclerosis del Hospital "Hermanos Ameijeiras". Esta aplicación es la base fundamental a partir de la cual se comenzó a desarrollar SAMAD para el diagnóstico y tratamiento de las Dislipoproteinemias.

Otros ejemplos de software médicos cubanos

- CoFar.
- Sistema para el registro y la evaluación de las respuestas.
- Sistema para el diagnóstico remoto de ECG.
- NeuroCid.
- CardioCid
- Software de registro y almacenamiento para el estudio del sueño.
- Medicid 5-132 (Polígrafo digital), etc.
- CLÍNICO 3.0
- Imagis (Sistema de transmisión de imágenes médicas digitales).

Relacionado con el desarrollo de la IM en Cuba, la Universidad de las Ciencias Informáticas (UCI), ha incorporado el perfil de medicina y bioinformática en la carrera de Ingeniería en Ciencias Informáticas, que se estudia en la facultad de bioinformática (Facultad 6), con la meta fundamental de formar un profesional capacitado en el desarrollo de software médico de alta calidad, para contribuir al desarrollo científico, tecnológico y económico del país.

1.3 ¿POR QUÉ DESARROLLAR EL REFINAMIENTO DEL ANÁLISIS Y DISEÑO DE SAMAD?

Se realiza el refinamiento del análisis y diseño de SAMAD debido a las dificultades que presenta la aplicación anterior para cumplir su objetivo de forma satisfactoria. Entre los problemas identificados se encuentran los requisitos funcionales, los cuales poseen dificultades en cuanto a la lógica del negocio. También existen otras dificultades de gran importancia que impidieron continuar con el desarrollo de SAMAD, tales como:

- La arquitectura de la aplicación anterior no se corresponde a la establecida por Softel.
- La seguridad de la aplicación anterior que se utilizó no es la más conveniente para garantizar la integridad, confidencialidad y disponibilidad de la información.
- La interfaz de usuario es poco amigable debido a que las funcionalidades fueron desarrolladas de manera que al médico le resulta incómodo trabajar con la aplicación.

CAPITULO I: FUNDAMENTACIÓN TEÓRICA

El análisis y diseño de esta nueva aplicación Web es la propuesta de un sistema informático en un ambiente Web que permite contribuir al diagnóstico y tratamiento de las dislipoproteinemias en Cuba. También contendrá nuevos módulos los que harán posible un diagnóstico y tratamiento más eficiente. Se puede mencionar que la aplicación tiene un corte educativo en cuanto a calidad de vida, pues lucha contra los malos hábitos de vida como el alcoholismo, el tabaquismo, la droga entre otros; mostrando los daños que pueden ocasionar estas enfermedades en la vida de una persona.

Para el desarrollo de la investigación los autores se han basado en los resultados del documento ATP III (para la entrada de datos al sistema y realización de algoritmos), entrevistas a médicos especialistas del Hospital Hermanos Ameijeiras, análisis del LIPICID desarrollado por el ICID e información médica obtenida de las bases de datos de Internet, libros de la especialidad y reutilización de algoritmos implementados en la aplicación anterior.

1.4 ARQUITECTURA DE SOFTWARE EN EL SISTEMA NACIONAL DE SALUD.

La red telemática de salud tiene una arquitectura de software ajustada a la establecida por las metodologías utilizadas por los especialistas de SOFTEL, empresa del Ministerio de la Informática y las Comunicaciones (MIC) encargada del desarrollo de tecnologías informáticas y de Programación para el Sistema Nacional de Salud (SNS) quedando acuñada la arquitectura de software del sistema nacional de salud con los siguientes elementos: (2)

- Ofrece los beneficios de reusabilidad de servicios, interoperabilidad de aplicaciones, utilización de un medio único de acceso a servicios.
- Web Services (WS) para desarrollar aplicaciones distribuidas altamente integradas que interactúen por XML, ofreciendo las ventajas de usar estándares abiertos, basados en texto pudiendo comunicar componentes escritos en distintos lenguajes y plataformas, muy fácil de implementar permitiendo ser re-empaquetadas la mayoría de las aplicaciones como servicio Web.
- El sistema deberá contar con un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (Authentication, Authorization and Accounting, AAA) con Autenticación de firma única (Single Sign On). 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. Para cada petición que se haga a un componente del sistema debe enviarse el token recibido en el certificado digital el cual será validado con la lista de tokens autorizados que el sistema ha registrado. Las trazas del sistema contendrán un texto descriptivo de las acciones realizadas así como el día, mes, año, hora, minuto, segundo en que se registra. El sistema permitirá a las personas designadas como administradores establecer las políticas de seguridad en su nivel de acceso.

Debido a todas las características de arquitectura descritas anteriormente para que pueda ser desplegado cualquier software desarrollado para el SNS en sus servidores centrales se torna necesario cumplir con los requerimientos de MySQL y desplegarse sobre servidor Linux. Estos

requisitos propuestos exigen a cualquier aplicación que se ejecute sobre esta red la utilización de PHP como tecnología del lado del servidor y Java Script como tecnología del lado del cliente.

1.5 ANALISIS DE LAS HERRAMIENTAS SELECCIONADAS.

1.5.1 Metodología de Desarrollo

La metodología utilizada para modelar el sistema propuesto es el Proceso Unificado de Desarrollo de Software (RUP, Rational Unified Process). RUP es la definición del conjunto de actividades que guían los esfuerzos de las personas implicadas en el proyecto, a modo de plantilla que explica los pasos necesarios para terminar el mismo. El Proceso de Desarrollo de Software tiene la misión de transformar los requerimientos del usuario en un producto de software de manera que los integrantes del equipo y todo aquel que pueda estar interesado en el producto final, tenga la misma visión, además de ser una metodología adaptable a proyectos de larga duración con equipos de trabajo de pocas experiencias.

Esta metodología tiene tres características principales:

- Guiado/Manejado por casos de uso: Los casos de uso constituyen la guía fundamental establecida para las actividades a realizar durante todo el proceso de desarrollo, incluyendo el diseño, la implementación y las pruebas del sistema. Esto se debe a que los casos de uso “[...] *proporcionan un medio sistemático e intuitivo de capturar requisitos funcionales*” (3) y dirigen todo el proceso de desarrollo del software.
- Centrado en la arquitectura: La arquitectura invoca los elementos más significativos del sistema y está influenciada entre otros por plataformas de software, sistemas operativos, gestores de bases de datos, consideraciones generales y requisitos no funcionales. La arquitectura abarca decisiones importantes sobre “[...] *la organización del sistema, los elementos estructurales, sus interfaces y comportamientos*”. (3)
- Iterativo e incremental: Para hacer más manejable un proyecto se recomienda dividirlo en ciclos. Para cada ciclo se establecen fases de referencia, cada una de las cuales debe ser considerada como un mini – proyecto, cuyo núcleo fundamental está constituido por

una o más iteraciones de las actividades principales básicas de cualquier proceso de desarrollo.

RUP propone 9 flujos de trabajo y para cada uno de ellos se definen roles y artefactos. La investigación se centrará en los siguientes 3 flujos:

- **Modelamiento del negocio:** Describe los procesos de negocio, identificando quiénes participan y las actividades que requieren automatización.

Roles: Analista y diseñador del negocio

Artefactos:

Actores del negocio, Casos de uso del negocio, Modelo de CUN (casos de uso del negocio), Realización de CUN (Diagramas de Actividades), Modelo de Objetos.

- **Requerimientos:** Define qué es lo que el sistema debe hacer, para lo cual se identifican las funcionalidades requeridas y las restricciones que se imponen.

Roles: Analista del sistema, Especificador de casos de uso, Diseñador de interfaz de usuario, Arquitecto.

Artefactos: Modelo de casos de uso, Actor, Casos de uso, Descripción de la arquitectura (vista del modelo de casos de uso), Prototipo de interfaz usuario.

Análisis y diseño: Describe cómo el sistema será realizado a partir de la funcionalidad prevista y las restricciones impuestas (requerimientos), por lo que indica con precisión lo que se debe programar.

Roles: Diseñador, Diseñador de interfaz de usuario, Diseñador de Base de Datos, Arquitecto.

Artefactos: Modelo del diseño, Clases del diseño, Realización de los CU, Diagramas de despliegue, Prototipos de interfaz de usuario, Modelo de datos.

1.5.2 UML

El Lenguaje Unificado de Modelación (UML, Unified Modeling Language) constituye una notación para especificar, construir, visualizar y documentar los artefactos de un sistema de software orientados a objetos. Un artefacto es una información que es utilizada o producida mediante un proceso de software y puede ser un modelo, una descripción o un software; los artefactos de UML se especifican en forma de diagramas, los que pueden ser definidos como una representación gráfica de una colección de elementos del modelo. (3)

Los diagramas fundamentales que emplea UML son los siguientes:

- Diagrama de casos de uso: Especifica las funcionalidades y el comportamiento de un sistema, su interacción con los usuarios del mismo y otros sistemas.
- Diagrama de Comportamiento o Interacción: Muestra las interacciones entre objetos ocurridas en un escenario o parte del sistema.
- Diagrama de Clases: Representa un conjunto de elementos del modelo que son estáticos. Estos aspectos estáticos modelan características del software como pueden ser su estructura interna y la representación que se hará de la información en la aplicación.
- Diagrama de Implementación: Muestra los aspectos físicos del sistema e incluye la estructura del código fuente y la implementación.

1.5.3 Framework Symfony

Para esta nueva aplicación de SAMAD se utiliza Symfony 1.0.10 pues es un framework que simplifica el desarrollo de la aplicación al presentar elementos que permiten garantizar la seguridad. Obliga a desarrollarse según el patrón arquitectónico Modelo-Vista-Controlador. Otra de las ventajas es que separa la lógica de negocio, la lógica de servidor y la presentación de la aplicación Web, resolviendo así unos de los principales problemas en el desarrollo de la arquitectura de SAMAD. Este framework proporciona estructura al código fuente, forzando al desarrollador a crear códigos más legibles y más fáciles de mantener. Symfony facilita la programación de aplicaciones, ya que encapsula operaciones complejas en instrucciones sencillas, es diseñado para optimizar el desarrollo de las aplicaciones Web. Proporciona varias herramientas y clases encaminadas a reducir el tiempo de desarrollo de una aplicación Web compleja. Además, automatiza las tareas más comunes, permitiendo al desarrollador dedicarse por completo a los aspectos específicos de cada aplicación. El resultado de todas estas ventajas es que no se debe reinventar la rueda cada vez que se crea una nueva aplicación Web.

Symfony está desarrollado completamente con PHP 5. Se decidió utilizarlo en el desarrollo de la aplicación porque 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 Unix como en plataformas Windows.

Incluye a Propel como mapeador Objeto-Relacional, Creole como capa de abstracción de la Base de Datos. Además puede ser fácilmente configurable con otros plugins probados, tales como el sfGuardPlugins para facilitar el proceso de autenticación.

1.5.4 Lenguaje de programación: PHP 5

Las siglas significan "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, Personal Home Page Tools). Es un lenguaje interpretado, su última versión desarrollada fue la 5.2.0. Se caracteriza por ser multiplataforma y por sus facilidades para ser incrustado en código HTML, se usa para crear páginas Web dinámicas, es un lenguaje para crear aplicaciones del lado del servidor lo que proporciona la característica de tener un código invisible al ejecutar las peticiones del cliente. En SAMAD se usa en combinación con el motor de bases de datos MYSQL aunque está preparado para soportar otros, trae un estándar ODBC, esto mejora en gran medida las posibilidades de conexión. También puede ser usado para programar en consola así como, para crear aplicaciones gráficas independientes del navegador. Es un lenguaje con un veloz procesamiento ya que no consume mucha memoria y en cuanto a su seguridad tiene niveles configurables por un archivo .INI, pudiendo ajustarse a las necesidades requeridas. Es el utilizado y estandarizado por el sistema informático de salud cubana.

1.5.5 Gestor de base de datos: MYSQL

Seleccionamos como gestor de base de datos MySQL ya que es un servidor de base de datos muy rápido, robusto, multitarea y multiusuario. Su principal objetivo de diseño fue la velocidad, además de consumir muy pocos recursos, tanto de CPU como de memoria. Tiene excelentes utilidades de administración como backup y recuperación de errores. Tiene una excelente integración con PHP. No hay límites en el tamaño de los registros, además de tener un control de acceso de los usuarios bastante amplio. Trabaja en diferentes plataformas además de soportar múltiples idiomas. Completo y optimizado uso del SQL. Usa listas de control de acceso para todas las conexiones, consultas y otras operaciones. En fin, es un servidor de Base de datos relacional, tiene interfaces para desarrollo de clientes en diversos lenguajes, está disponible en diferentes plataformas, es gratis y se puede disponer del código fuente.

1.5.6 APACHE

El servidor HTTP Apache 2.2.3 es un software (libre) servidor HTTP de código abierto que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. En cuanto a uso, sirve de referencia para el diseño y valoración de otros servidores Web. Es utilizado en el desarrollo de software para el sistema de salud cubano. El servidor Apache es un software estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías:

- Módulos Base: Módulo con las funciones básicas del Apache.
- Módulos Multiproceso: Son los responsables de la unión con los puertos de la máquina.
- Módulos Adicionales: Cualquier otro módulo que le añada una funcionalidad al servidor.

Las funcionalidades más elementales se encuentran en el módulo base, lo cual hace necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiproceso para cada uno de los sistemas operativos sobre los que se ejecuta Apache, optimizando el rendimiento y rapidez del código. El resto de las funcionalidades del servidor se consiguen por medio de módulos adicionales que se pueden cargar. Para introducir un conjunto de utilidades al servidor, hay que añadirle simplemente un módulo, de forma tal que no sea necesario reinstalar el software.

Ventajas:

- Open source
- Multi-plataforma
- Popular (fácil conseguir ayuda/suporte)
- Gratuito

1.5.7 ENTORNO DE DESARROLLO INTEGRADO. (IDE)

Un entorno de desarrollo integrado o en inglés Integrated Development Environment ('IDE') es un programa compuesto por un conjunto de herramientas para el programador que puede dedicarse

a un sólo lenguaje de programación o bien, poder utilizarse para varios. Es posible que un mismo IDE pueda funcionar con varios lenguajes de programación.

1.5.7.1 ECLIPSE

Se utilizará el eclipse con el plug-in PDT en ingles significa PHP Development Tools (plug-in para programar en PHP con el IDE Eclipse), el Eclipse con el plug-in presenta un entorno amigable, completamiento de código, alta integración con el framework Symfony, gran cantidad de plug-in que le extienden la funcionalidades tales como Aptana (para el desarrollo con AJAX), Subclipse para integración con Subversion, Mylyn para integración con Trac, que hacen al Eclipse una herramienta de desarrollo potente.

1.5.8 Combinación de herramientas: Linux, Apache, MYSQL, PHP.

El desarrollo de aplicaciones informáticas para la Salud Pública en Cuba definido por SOFTEL se basa en la utilización de un grupo de herramientas de software libre tales como Linux, PHP, MYSQL y Apache (LAMP). Formando una de las mejores herramientas que proporciona el software libre para tener un servidor Web muy versátil y poderoso, además de no tener que pagar licencias, para su manteniendo sólo hay que actualizar paquetes que son publicados en Internet, es bastante seguro pues los parches de seguridad son publicados de forma rápida luego de producida una alerta. Por otro lado brinda muy buenas prestaciones en una amplia gama de hardware con bajo nivel de requerimientos como son: Intel y compatibles, SPARC, Mips y PPC (Macintosh), es portable pues cuenta con un código bastante sencillo que no implica grandes cambios entre una plataforma u otra, con él se pueden incrementar los servicios y funciones desde el código fuente; mejorando así su rendimiento.

1.5.9 Ingeniería de Software Asistida por Computadora (Herramienta CASE).

El nombre CASE proviene de Computer-Aided Software Engineering (Ingeniería de Software Asistida por Computadora) la cual está diseñada para automatizar o apoyar una o más fases del ciclo de vida del desarrollo de sistemas software, logrando una íntegra calidad al cliente que solicita el producto a desarrollar. Representa una gran ayuda a todo el trabajo de mesa llevado a cabo por los ingenieros de software acortando el tiempo de trabajo y simplificando su

complejidad, además de reducir el costo de dicho desarrollo en términos de tiempo y dinero, ayudando al diseño de los proyectos.

1.5.9.1 Herramienta para el desarrollo de software. Visual Paradigm.

Esta aplicación está desarrollada por Visual Paradigm una de las principales herramientas CASE que tiene la capacidad de ejecutarse sobre diferentes sistemas operativos lo que le confiere la característica de ser multiplataforma, donde su mayor éxito consiste en su utilización sobre el sistema operativo libre GNU/Linux. Integra diferentes funcionalidades para el desarrollo de aplicaciones como lo es el modelado de UML, el modelado de base de datos, el modelado de requerimientos, el modelado del proceso de negocio, la generación de documentación entre otros.

Las transiciones del análisis al diseño, y de este a la implementación, están adecuadamente integradas dentro de la herramienta CASE, de manera que reduce significativamente los esfuerzos de todas las etapas del ciclo de desarrollo de software.

Se ha seleccionado la herramienta Visual Paradigm por poseer licencias de uso en la Universidad de las Ciencias Informáticas, además por continuar con las políticas establecidas por SOFTEL en el desarrollo de aplicaciones para el SNS.

Visual Paradigm ofrece las siguientes ventajas:

- Diseño centrado en casos de uso y enfocado al negocio, lo cual permite generar un software de mayor calidad.
- Uso de un lenguaje estándar común a todo el equipo de desarrollo que facilita la comunicación.
- Capacidades de ingeniería directa (en su versión profesional) e inversa.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Disponibilidad de múltiples versiones, para cada necesidad.
- Disponibilidad de integrarse en los principales IDE (Integrated Development Environment, por ejemplo con el IDE Eclipse a través de la herramienta VP-UML Integration).
- Disponibilidad en múltiples plataformas.

CONCLUSIONES

Se realizó un estudio del estado del arte del tema tratado en la aplicación a nivel Internacional y Nacional, se llevó a cabo un análisis crítico en la propuesta de por qué realizar el refinamiento del análisis y diseño de SAMAD. Se abordó el tema de la arquitectura establecida por Softel para la inserción del software en el Sistema de Salud Cubano. Las tecnologías empleadas (Java script y HTML) para el lado del cliente y (MySQL, PHP y APACHE) lado del servidor. Visual Paradigm como herramienta Case y como entorno de desarrollo integrado Eclipse.

CAPÍTULO 2: CARACTERÍSTICAS DEL SISTEMA

En el presente capítulo se lleva a cabo un proceso de desarrollo del software donde se exponen los objetivos estratégicos de la organización; la modelación del negocio para lo cual se identifican y describen los actores, trabajadores y casos de uso del negocio, se realizan los diagramas de actividades correspondientes a cada caso de uso del negocio, y el diagrama del modelo de objetos. Se plantean los requisitos funcionales y no funcionales de la aplicación a desarrollar y se modela la misma en términos de casos de uso de sistema.

2.1 OBJETIVOS DE LA ORGANIZACIÓN

El Hospital Hermanos Ameijeiras juega un papel fundamental en el cuidado de la salud, en el mismo existen varias consultas de especialidades que responden a las necesidades específicas de los pacientes que solicitan atención médica. Dentro de sus objetivos se encuentra la atención a paciente, además de formar médicos altamente calificados, para lo cual se promueve y lleva a cabo un amplio movimiento investigativo y docente.

El principal rol de los médicos en la organización es la atención a los pacientes, proveerles de la información necesaria y remitirlos hacia determinadas consultas en caso de ser necesario.

La consulta de Dislipoproteinemias de dicho hospital, tiene asociado un médico clínico y una enfermera. Está tiene como propósito la atención a personas con sospecha o presencia de enfermedades relacionadas con el trastorno del metabolismo de los lípidos. En ella se brinda al paciente toda la información necesaria de las dislipoproteinemias, recomendaciones de un modo de vida saludable, y se lleva a cabo todo lo referente al diagnóstico, tratamiento y posterior seguimiento de los pacientes.

En la historia clínica (HC) se incorporan todos los datos personales, familiares, síntomas, diagnóstico, recomendaciones, así como el tratamiento indicado de un paciente. El médico, para determinar con seguridad el padecimiento que presenta el paciente, requieren de los resultados de exámenes, tales como perfil lipídico, que es el fundamental entre otros los cuales dependerán de características más particulares de la salud de cada paciente.

Con la atención de las dislipoproteinemias el hospital tiene el fin de ayudar a mejorar la calidad de vida de la población y prevenir enfermedades cardíacas y cerebrovasculares, también esto posibilita tener información para realizar estudios estadísticos que permitan elaborar un programa cada vez más eficiente, orientado hacia la prevención y control de las enfermedades crónicas, toda la información recogida en estas consultas no pueden ser eliminadas del sistema para posteriores estudios.

2.2 MODELO DEL NEGOCIO ACTUAL

El primer paso en el proceso de desarrollo de software es precisamente comprender todos los procesos que se realizan manualmente, debido a que la aplicación anterior no fue totalmente concluida, además de no tener un negocio bien detallado, de ahí que sea muy útil la creación de modelos que organicen y presenten los detalles importantes de la problemática real vinculados con el sistema informático a construir. Estos modelos deben cumplir una serie de propiedades, entre ellas la de ser coherentes y relacionados.

La descripción del negocio propuesto en detalle tendrá entre sus actividades principales la identificación de los procesos del entorno organizacional, delimitación del modelo de casos de uso, la especificación de los casos de uso, la identificación de trabajadores y entidades que ejecutan las realizaciones de los casos de uso y detallar la definición de las entidades y las responsabilidades de los trabajadores.

2.2.1 FLUJO ACTUAL DE LOS PROCESOS

El paciente solicita atención en la consulta de dislipoproteinemias, en esta primera consulta el médico recibe al paciente y le confecciona una historia clínica con los datos personales y otros datos relacionados con la enfermedad. El médico le indica a la enfermera realizarle un examen físico a un paciente y posterior a esto le indica al paciente un conjunto de análisis que debe de traer en su próxima consulta entre los que se encuentran: Electrocronograma (ECG), Lipidograma, Hemoquímica y Ultrasonido.

El grado de seguridad del diagnóstico dependerá en gran medida de la información obtenida a partir de las preguntas al paciente y de cómo se manejen estos datos en conjunto, lo cual resulta ser lo más engorroso por tratarse de muchos elementos a tener en cuenta para un diagnóstico con mayor exactitud. Terminado la primera consulta el médico le orienta asistir a una segunda consulta con los resultados de estos análisis.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

El paciente se presenta a una segunda consulta luego de haberse realizado los análisis indicados por el médico en la primera consulta. El médico analiza los resultados del ECG, Lipodograma, Hemoquímica y Ultrasonido y los registra en la Historia Clínica junto con la presión arterial y el peso corporal que son examinados por la enfermera antes de entrar a la consulta. El médico realiza un diagnóstico más completo e indica un tratamiento al paciente de acuerdo con los resultados de los análisis.

Independientemente de estos resultados el médico orienta al paciente asistir a las consultas de seguimiento que se realizan entre 8 y 12 semanas en dependencia de lo que decida el médico con nuevos exámenes realizados. En esta consulta evolutiva se recogen los resultados de los exámenes en la historia clínica y se analizan si hay avances positivos en el paciente de acuerdo con el tratamiento anteriormente indicado. En caso que no exista ningún avance positivo en el paciente el médico determinará si se incrementa la dosis del medicamento que usa, intensifica las medidas no farmacológicas, añade otro medicamento o cambia la familia de medicamento que estaba usando. También el médico debe de poner en la HC la opción de que el paciente cumple o no con el tratamiento indicado, así como las recomendaciones que el paciente debe de cumplir. Es importante siempre consultar la HC ya que está posee toda la información relacionada con el paciente.

Una vez realizado un nuevo análisis de la situación del paciente el médico determina si este debe seguir asistiendo a otras consultas. En caso que todos sus exámenes sean correctos entonces se le orienta al paciente que debe asistir a una consulta dentro de 6 meses para un nuevo estudio de los lípidos y si en esta ocasión todo continua normal el período de consulta se extiende a un año.

2.2.2 PROCESOS QUE SON OBJETOS DE AUTOMATIZACIÓN

Gestión de datos del paciente:

- Automatización de los datos del Paciente en la HC: este proceso consiste en una base de datos que tendrá toda la información del paciente. La entrada de los datos personales así como la información de cada consulta se guardarán de modo que al ser consultada la HC se pueda estudiar el caso con todos los detalles. Cada consulta podrá ser visualizada y sólo los datos personales pueden ser modificados, por si

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

cambia de dirección o se detecta algún error en la información. Se permite completar los exámenes que estén sin llenar de la primera consulta. La HC sólo puede ser vista por el médico que atiende a su correspondiente paciente y mostraría una gráfica que refleja el estado del paciente, refiriéndose al comportamiento del LDL correspondiente a cada consulta.

- Diagnóstico: consiste en la generación de un diagnóstico que tendrá en cuenta los parámetros que se hayan entrado hasta el momento, el diagnóstico será más acertado a medida que aumente la información, considerándose muy importante desde un inicio, tener los resultados, del perfil lipídico del paciente. El diagnóstico terminado la segunda consulta cambiará en caso de completar algún examen que influya en este, para ello se tiene que generar el diagnóstico nuevamente para ser actualizado.
- Búsqueda de Pacientes: este proceso automatizado es más rápido que en el negocio y restringe la búsqueda, posibilitando que se pueda buscar por varios criterios de búsquedas por ejemplo: Nombre del paciente, 1er apellido del paciente, 2do apellido paciente, carne de identidad del paciente.
- Realizar Chequeo Evolutivo: consiste en ir llenando fichas de evolución del paciente en cada consulta realizada cada 12 semanas para llevar el estado del paciente con respecto al tratamiento indicado.

2.2.3 ACTORES DEL NEGOCIO

Para entender completamente el propósito del negocio de esta investigación, es necesario saber quién interactúa en el entorno en que se realizan las actividades de las consultas de dislipoproteinemias; o sea, quién envía o recibe algún beneficio de estas consultas. *“El término actor significa el papel que desempeña alguien o algo al interactuar con el negocio.”* (4)

Actores del negocio	Justificación
Paciente	Son los individuos que solicitan atención en el Hospital en la consulta de dislipoproteinemias y que se benefician al recibir diagnóstico y tratamiento asociado a su padecimiento.

Tabla 2. 1. Descripción de los actores del negocio.

2.2.4 TRABAJADORES DEL NEGOCIO

“Aquellas personas que ejecutan dentro del negocio los diferentes procesos, son llamados trabajadores del negocio.” (4)

Trabajadores del negocio	Justificación
Médico Clínico.	Es el encargado de atender al paciente, recoger sus datos, emitir diagnóstico y tratamiento.
Enfermera	Es la que realiza el examen físico al paciente.

Tabla 2. 2 Descripción de los trabajadores del negocio.

2.2.5 DIAGRAMA DE CASOS DE USO DEL NEGOCIO

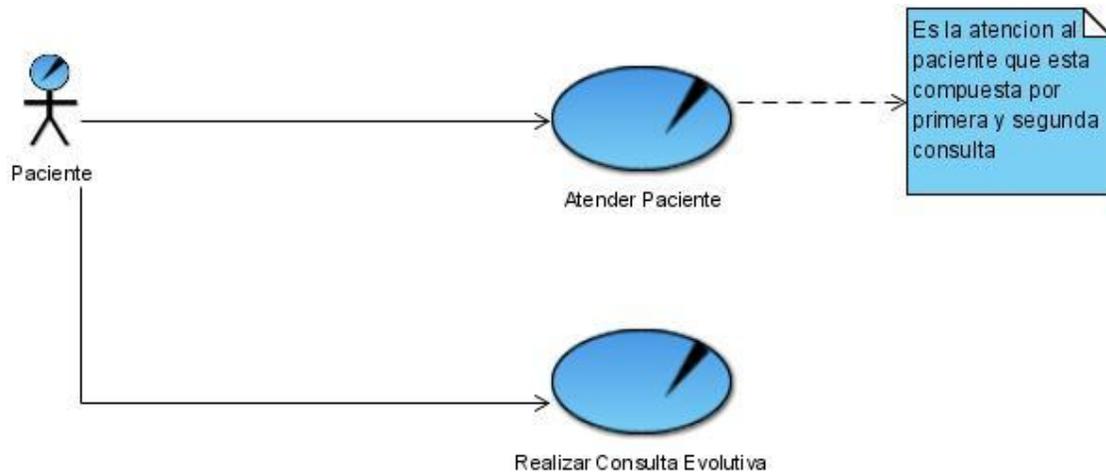


Fig. 2. 1 Diagrama de Casos de Uso del Negocio

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

2.6 DIAGRAMA DE ACTIVIDADES DEL NEGOCIO

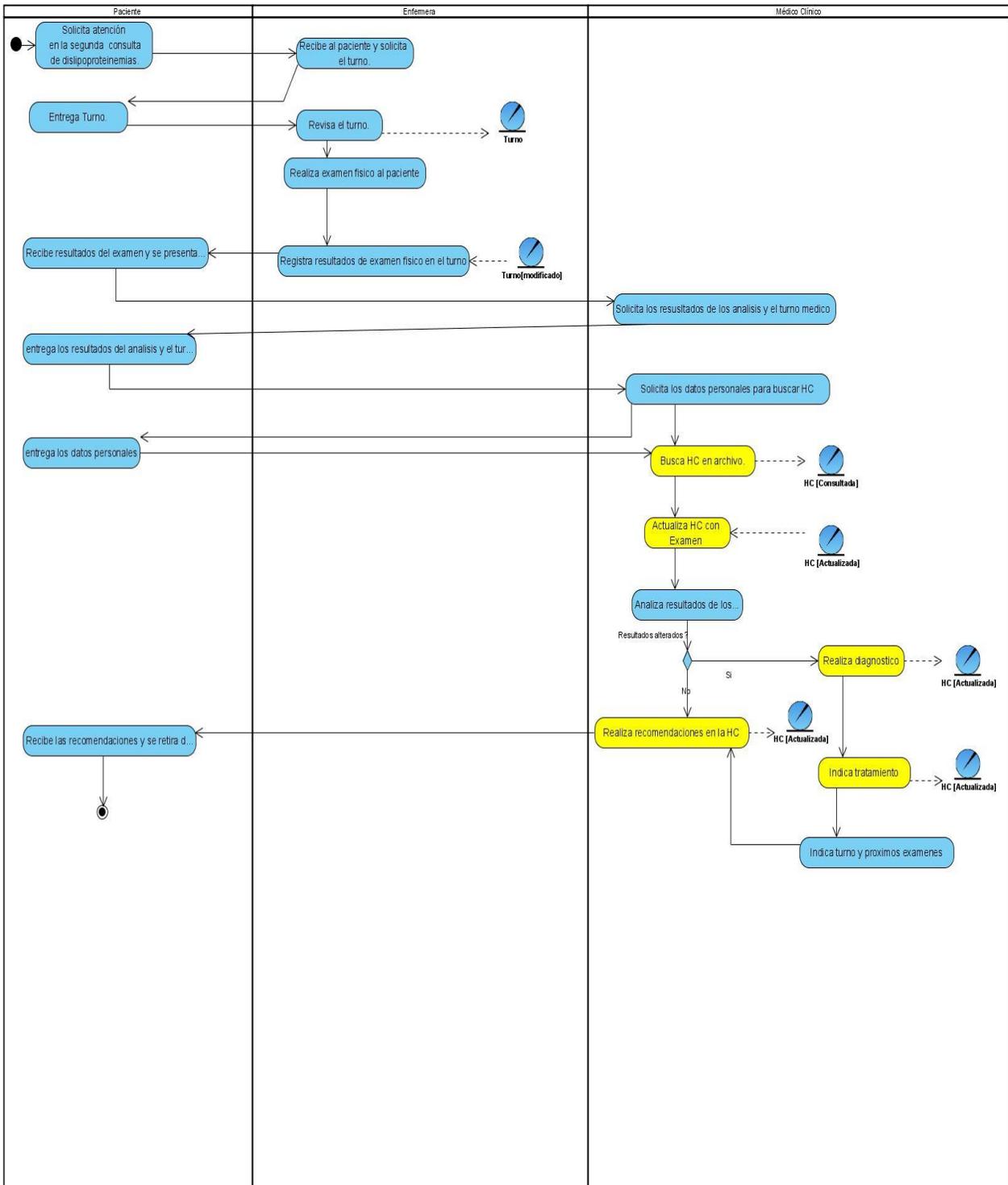


Fig.2.2 Diagrama de actividades: CUN Atender Paciente

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

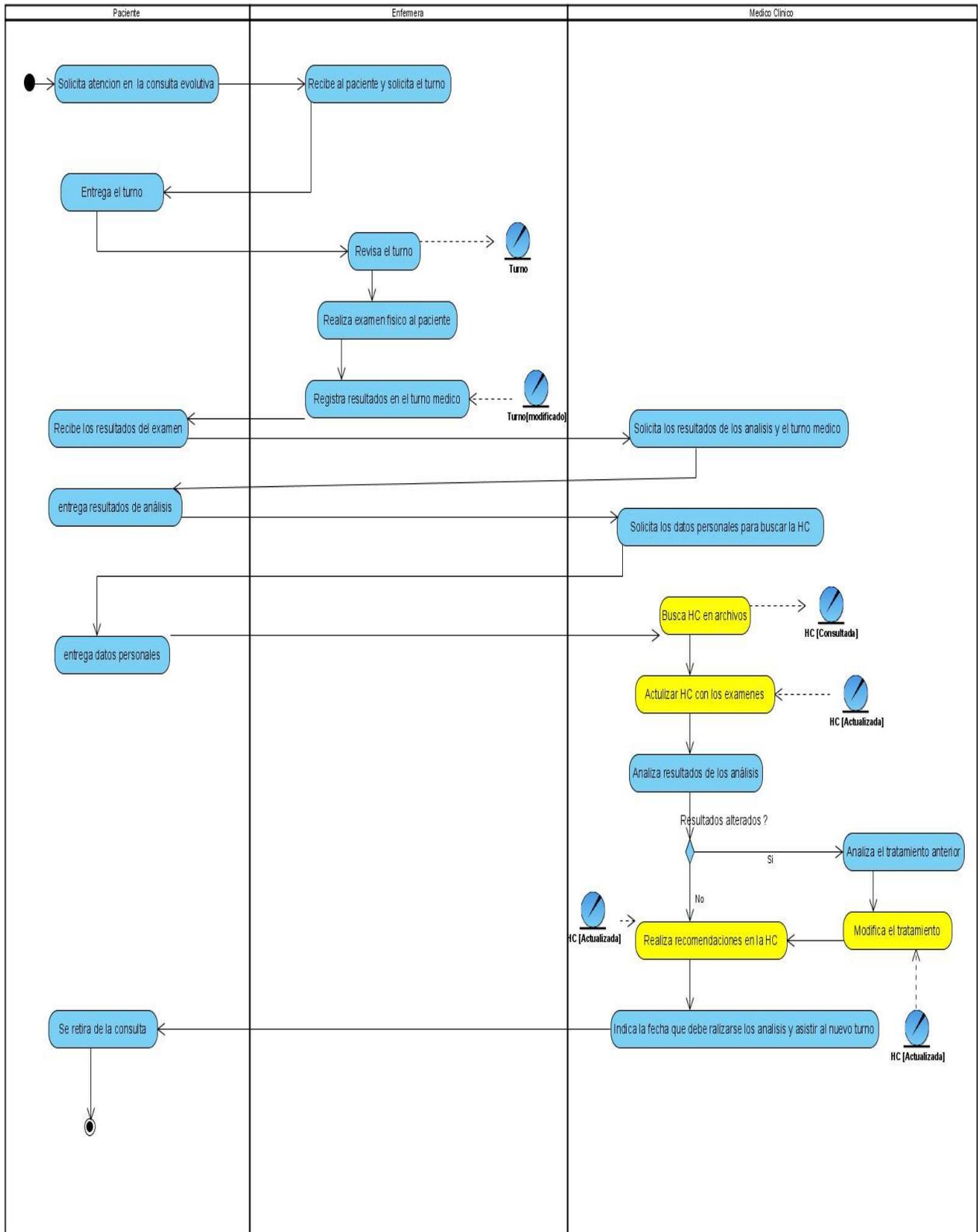


Fig.2.3 Diagrama de actividades: CUN Realizar Consultas Evolutivas

2.2.7 MODELO DE OBJETOS DEL NEGOCIO

El diagrama de clases del modelo de objeto, es un artefacto que se construye para describir el modelo de objetos del negocio, muestra la participación de los trabajadores y entidades del negocio y la relación entre ellos. (5)



Fig. 2.4 Diagrama de Objetos del negocio.

2.3 PROPUESTA DEL SISTEMA

SAMAD será un sistema desarrollado en ambiente Web con una agradable interfaz para introducir cada dato que se requiere, los cuales se almacenan de forma organizada por consulta en la Historia clínica (HC) de cada paciente. El sistema facilitará la actualización de parámetros que cambian a nivel internacional cada cierto tiempo y presenta emisión de reportes a partir de una búsqueda avanzada y del cambio de rangos y valores para la validación de exámenes, cálculo de factores de riesgos (FR), cálculo de índice de masa corporal (IMC), cálculo para el Peso Ideal (PI), cálculo de Dieta según las Calorías (DC), contendrá un módulo de Atención de Pacientes en seguimiento. Cada médico podrá estudiar los datos de sus pacientes, mediante la visualización de la HC. Además contará con artículos e información relacionados con las dislipoproteinemias y el colesterol.

2.3.1 ESPECIFICACIONES DE LOS REQUISITOS.

El flujo de trabajo de requerimientos ayuda a establecer y mantener el acuerdo con los clientes o los interesados en la aplicación, proporciona a los desarrolladores del sistema una mejor comprensión de los requisitos, define las fronteras del software, establece una base para planificar el contenido técnico de las iteraciones, costo y tiempo de desarrollo del sistema y define una interfaz para el usuario enfocada en las necesidades y metas planteadas.

2.3.1.1 REQUISITOS FUNCIONALES

“Los requerimientos funcionales son capacidades o condiciones que el sistema debe cumplir.” (5) Para el desarrollo de SAMAD se han definido los siguientes requisitos funcionales:

- R1. Validar usuario y contraseña a los usuarios del sistema.
 - R 2. Asignar Privilegio de Usuario.
 - R 3 Buscar y Visualizar datos de Usuario.
 - R4. Calcular Peso Ideal (PI).
 - R5. Calcular dietas según calorías. (DC)
 - R6 Calcular índice de masa corporal (IMC).
 - R7 Calcular factores de riesgos (FR).
 - R 8 Registrar Datos del Paciente.
 - R 9 Actualizar Datos del Paciente.
 - R 9.1 Modificar Datos personales.
 - R 9.2 Completar Exámenes.
 - R 9.3 Generar diagnóstico actualizado.
 - R 9.4 Generar tratamiento farmacológico y no farmacológico
 - R 10 Buscar y Visualizar Datos del Paciente.
 - R 10.1 Mostrar datos recogidos en primera consulta.
 - R 10.2 Mostrar datos recogidos en consultas evolutivas.
 - R10.3 Generar gráfica de estado del paciente (Valor de LDL vs # Consultas)
 - R 10.4 Mostrar diagnóstico.
 - R 10.5 Mostrar tratamiento farmacológico y no farmacológico.
 - R 10.6 Generar gráfica con porcentaje de usuarios que se han calculado sus
- FR.
- R 11 Insertar datos de la consulta Evolutiva
 - R 12 Rectificar datos de la consulta Evolutiva
 - R 13 Generar reportes de Búsqueda de datos combinados (Búsqueda avanzada).
 - R 14. Cambiar rangos y valores para la validación de exámenes.
 - R 15 Actualizar fármacos para el diagnóstico.

2.3.1.2 REQUISITOS NO FUNCIONALES

“Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Estas propiedades se ven como las características que hacen al producto atractivo, usable, rápido o confiable.” (5)

➤ Apariencia o interfaz externa.

La aplicación de SAMAD presenta un color que denota seriedad en el tema además de ser atractivo a la vista del usuario, se ha seleccionado como color predominante el verde. También requiere de información concreta y útil, con el objetivo de mantener a todos los usuarios informados y guiados en la prevención y el tratamiento de las dislipoproteinemias. Sólo se incorporan las imágenes necesarias para ayudar en la identificación de determinados signos que pueden ser difíciles de reconocer en el paciente.

➤ Usabilidad.

Garantiza un acceso fácil y rápido a los usuarios. El sistema podrá ser usado por cualquier persona que posea conocimientos básicos en el manejo de una computadora y de un ambiente Web en sentido general.

➤ Rendimiento.

Garantiza tiempos de respuestas generalmente rápidos al igual que la velocidad de procesamiento de la información.

➤ Seguridad.

El sistema deberá contar con un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (Authentication, Authorization and Accounting, AAA) con Autenticación de firma única (Single Sign On). Garantizando que la información solo pueda ser vista por los usuarios con el nivel de acceso autorizado para ello. Permitiendo que las funcionalidades del sistema se muestren de acuerdo al nivel de usuario que esté activo. El sistema debe contar con protección contra acciones no autorizadas o que puedan afectar la integridad de los datos. Cumpliendo con la confidencialidad de los datos, permitiendo que cada médico solo pueda ver sus pacientes.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

➤ Políticos-culturales.

Se emplea en la aplicación el idioma español. Cuenta con logotipos e imágenes que se encuentren en correspondencia con el carácter científico y profesional del tema. Algún cambio que se quiera realizar debe ser previamente consultado con la dirección del Hospital Hermanos Ameijeiras y canalizado por la UCI con la dirección de Producción central y de la Facultad a la que pertenece el proyecto.

➤ Legales.

Publicarse en Infomed como herramienta para la ayuda al diagnóstico y tratamiento de las dislipoproteinemias. Se estará usando herramientas de software libre, bajo la licencia GNU/GPL.

➤ Confiabilidad.

El sistema permitirá la creación de copias de respaldo que puedan restaurar el sistema en caso de fallo crítico o pérdida total de la información. La información manejada por el sistema será objeto de cuidadosa protección contra la corrupción y estados inconsistentes. Deberán existir mecanismos de chequeo de integridad. El sistema debe ser administrado solamente por una persona capacitada al cual le llamaremos administrador y será usada por médicos inscritos en la aplicación, por tanto, la información que fluirá será la que se emite realmente en una consulta de un médico. Validando la captación de datos en el caso de los médicos al inscribirse para evitar entradas inadecuadas.

➤ Software.

El software base debe responder a las políticas de Software Libre y de Fuente Abierta. Debe disponer además de un Servidor http Apache, Lenguaje de programación del lado del servidor php, Servidor de datos mysql y de un navegador capaz de interpretar secuencias de códigos Java Script.

➤ Hardware.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

PC servidor: 80 GB disco duro, microprocesador superior a 3.00 GHz, 512 MB mínimo de RAM.

2.3.2 DEFINICIÓN DE LOS CASOS DE USO DEL SISTEMA

2.3.2.1 ACTORES DEL SISTEMA

“Cada trabajador del negocio que tiene actividades a automatizar es un candidato a actor del sistema. Si algún actor del negocio va a interactuar con el sistema, entonces también será un actor del sistema.” (5)

Actores del sistema	Justificación
Médico	Gestiona toda la información relacionada con la atención al paciente, y tiene acceso al sistema de la gestión de tratamientos y de consultas evolutivas.
SAAA (Actor externo)	Gestiona toda la seguridad del sistema, es un mecanismo de seguridad basado en el modelo de Autenticación, Autorización y Auditoria (Authentication, Authorization and Accounting, AAA) con Autenticación de firma única (Single Sign On).
Administrador	Interactúa con todas las funcionalidades del sistema, tiene acceso a las actividades del rol de médico y con el rol de administrador específicamente puede cambiar los rangos y valores para la validación de exámenes, actualizar los fármacos para el diagnóstico y gestionar los datos del médico como usuario.
Usuario	Interactúa con el cálculo del peso ideal, cálculo de factores de riesgos, cálculo de índice de masa corporal y el cálculo de dietas según calorías, además de navegar por el sistema en la parte informativa (artículos, noticias).

Tabla 2. 3 Descripción de los actores del sistema.

2.3.2.2 DIAGRAMA DEL SISTEMA

DIAGRAMA DE RELACIÓN ENTRE ACTORES DEL SISTEMA

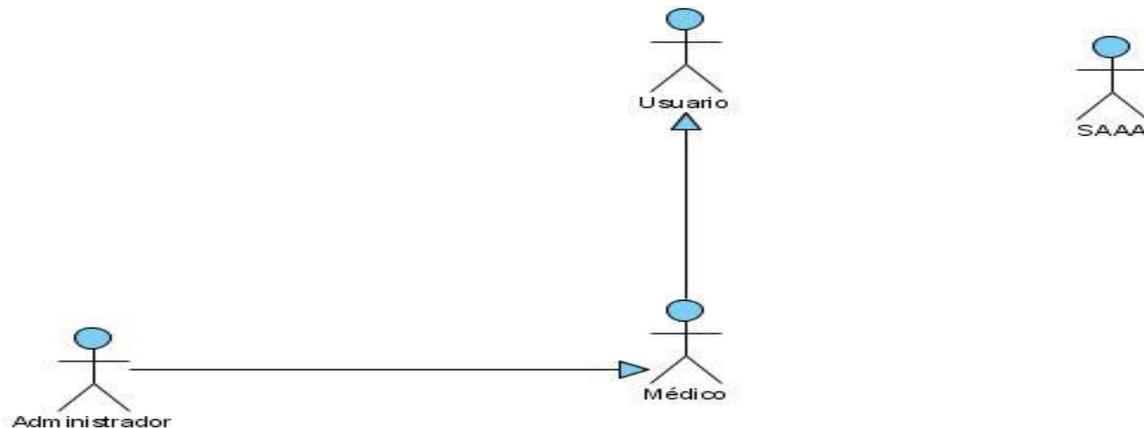


Fig. 2.5 Diagrama de relación entre actores.

El sistema cuenta con 12 casos de uso, de ellos 6 son arquitectónicamente significativos. Los cuales se representan en los diagramas siguientes.

2.3.2.3 DIAGRAMA DE CASOS DE USO DEL SISTEMA

“Los diagramas de casos de uso son importantes para visualizar, especificar y documentar el comportamiento de un elemento. Estos diagramas facilitan que los sistemas, subsistemas y clases sean abordables y comprensibles, al presentar una vista externa de cómo pueden utilizarse estos elementos en un contexto dado”. (5)

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

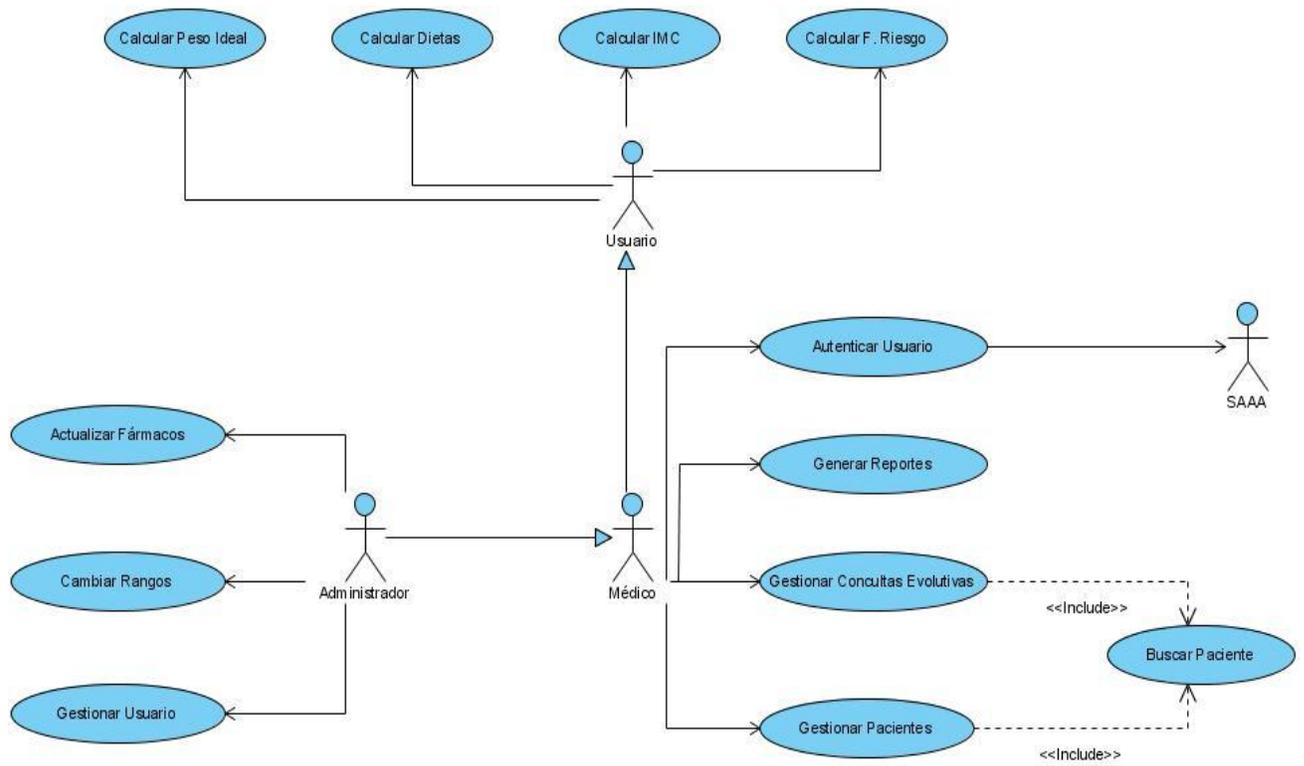


Fig.2.6 Diagrama de Casos de Uso del Sistema

Diagrama de vista de casos de uso arquitectónicamente significativos.

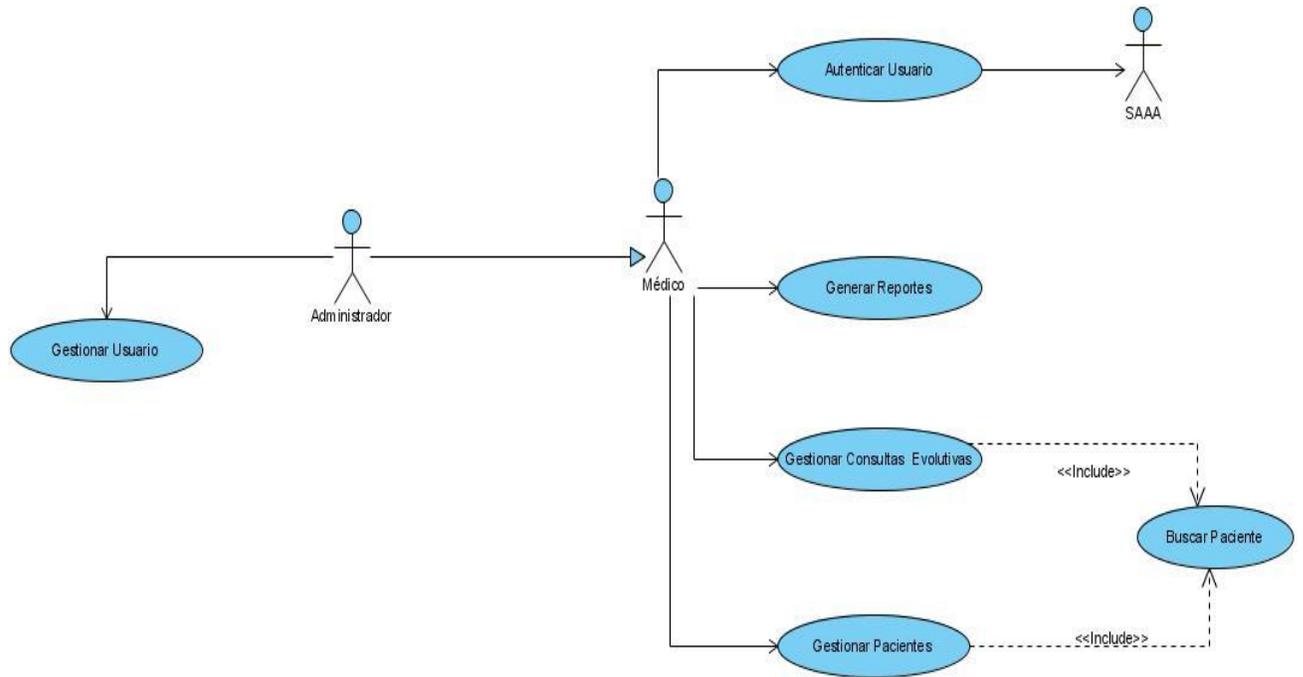


Fig.2.7 Diagrama de Vista de casos de uso arquitectónicamente significativos.

2.3.2.4 DESCRIPCIONES DE LOS CASOS DE USO DEL SISTEMA

Mediante la descripción de los casos de uso del sistema se detalla la secuencia de eventos que los actores llevan a cabo para completar un proceso a través de la aplicación.

CU 1: Gestionar Consultas Evolutivas

Nombre del Caso de Uso	Gestionar Consultas Evolutivas.
Actores	Médico(inicia)
Propósito	Insertar ó Rectificar una consulta evolutiva.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

Resumen	<p>El CU se inicia cuando el Médico pulsa sobre el vínculo consulta y el sistema le ofrece realizar algunas de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Insertar Consulta Evolutiva: Cuando el paciente ya existe en el sistema y asiste a la consulta, el Médico le pide los datos necesarios para crear la Consulta Evolutiva, finalizando así el CU. • Rectificar Consulta Evolutiva: Cuando un Médico necesita modificar los datos de una Consulta Evolutiva existente, rectifica los datos e indica actualizar los mismos, finalizando así el CU. 	
Referencias	R 11, R 12	
Precondiciones	El Médico debe estar autenticado y debe de haber realizado la búsqueda del paciente al cual le va a insertar ó rectificar una consulta evolutiva.	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del sistema.	
<p>1. El Médico selecciona cuál acción desea realizar:</p> <p style="margin-left: 40px;">a. Insertar Consulta Evolutiva.</p> <p style="margin-left: 40px;">b. Rectificar Consulta Evolutiva.</p>	<p>2. El sistema, en dependencia de la acción solicitada por el Médico, muestra la interfaz correspondiente:</p> <ul style="list-style-type: none"> • Insertar Consulta Evolutiva (Ir a la sección” Insertar Consulta Evolutiva”). • Rectificar Consulta Evolutiva (Ir a la sección” Rectificar Consulta Evolutiva”). 	
Sección “Insertar Consulta Evolutiva.”		
Acciones del Actor	Respuesta del Sistema	
	<p>1. El sistema muestra el formulario seleccionado permitiéndole al médico insertar</p>	

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	los datos de la consulta evolutiva y realizar los cálculos auxiliares al paciente.
2. El Médico inserta los datos del paciente en el formulario y le indica al sistema generar diagnóstico-tratamiento.	3. El sistema verifica que no exista ninguna consulta evolutiva con la misma fecha.
	4. El sistema verifica que todos los campos estén llenos.
	5. El sistema verifica que los datos introducidos sean correctos.
	6. El sistema genera automáticamente un diagnóstico y un tratamiento a partir de los datos entrados y los muestra en los campos correspondientes.
	7. El sistema ofrece la posibilidad al médico de emitir sus recomendaciones si así lo desea.
8. El Médico realiza sus recomendaciones de ser necesario e indica al sistema guardar los datos.	9. El sistema registra en la Base de Datos la consulta evolutiva.
	10. El sistema ofrece la posibilidad al médico de realizar los cálculos auxiliares.
11. EL Médico realiza los cálculos auxiliares y le indica al sistema guardar todos los datos.	12. El sistema guarda los cálculos realizados que corresponden a esta consulta y finaliza el caso de uso.
Nota: El formulario correspondiente a la opción de Datos de Consulta Evolutiva se	

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

muestra en los anexos en los prototipos no funcionales.	
Flujo alternativo 3.1 “Si ya existe una consulta con la misma fecha.”	
Acciones del Actor	Respuesta del Sistema
	3.1 El sistema muestra un mensaje “Ya existe una consulta evolutiva con la fecha señalada, solo puede rectificar los datos de la misma” y va a la acción 2 de esta Sección.
Flujo alternativo 4.1 “Si quedó algún campo vacío”	
Acciones del Actor	Respuesta del Sistema
	4.1 El sistema emite un mensaje “Todos los campos deben llenarse” y se indica en color rojo, el nombre de los campos vacíos y va a la acción 2 de esta Sección.
Flujo alternativo 6.1 “Si los datos no son correctos”	
Acciones del Actor	Respuesta del Sistema
	6.1 El sistema emite un mensaje “Los datos deben estar en el rango establecido” y se indica en color rojo, el nombre de los campos incorrectos y va a la acción 2 de esta Sección.
Flujo alternativo 11.1 “Si el médico no desea realizar los cálculos”	
Acciones del Actor	Respuesta del Sistema
11.1 El Médico no realiza los cálculos y finaliza el CU.	

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

Sección “Rectificar Consulta Evolutiva”	
Acciones del Actor	Respuesta del Sistema
	1. El sistema muestra una interfaz con la lista de las consultas existentes permitiéndole al médico seleccionar la consulta que desea rectificar.
2. El Médico selecciona la consulta a la que desea rectificar los datos.	3. El sistema muestra el formulario de la consulta evolutiva con los datos que existen en ella, permitiéndole al médico hacer los cambios necesarios.
4. El Médico rectifica los datos que desee de la consulta evolutiva seleccionada y le indica al sistema generar el diagnóstico-tratamiento.	5. El sistema verifica que no exista ninguna consulta evolutiva con la misma fecha.
	6. El sistema verifica que todos los campos estén llenos.
	7. El sistema verifica que los datos introducidos sean correctos.
	8. El sistema genera automáticamente un diagnóstico y un tratamiento a partir de los nuevos datos entrados y los muestra en los campos correspondientes.
	9. El sistema ofrece la posibilidad al médico de emitir sus recomendaciones si así lo desea.
10. El Médico realiza sus recomendaciones de ser necesario e	11. El sistema actualiza en la Base de Datos la consulta evolutiva.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

indica al sistema guardar los cambios.	
	12. El sistema ofrece la posibilidad de rectificar los cálculos auxiliares.
12. EL Médico realiza nuevamente los cálculos auxiliares y le indica al sistema guardar todos los datos.	13. El sistema guarda los cálculos realizados que corresponden a esta consulta y ofrece la posibilidad de modificar otra consulta.
14. El Médico selecciona otra consulta a la que desea rectificar los datos.	15. El sistema va a la acción 3 de esta Sección.
Nota: Es el mismo formulario de registrar Datos de Consulta Evolutiva.	
Flujo alternativo 1.1 “Si no existen consultas evolutivas”	
Acciones del Actor	Respuesta del Sistema
	1.1 El sistema muestra un mensaje “Este paciente no posee consultas evolutivas” y finaliza el caso de uso.
Flujo alternativo 3.1 “Si ya existe una consulta evolutiva con la misma fecha.”	
Acciones del Actor	Respuesta del Sistema
	3.1 El sistema muestra un mensaje “Ya existe una consulta en la fecha señalada” y va a la acción 4 de de esta Sección.
Flujo alternativo 6.1 “Si quedó algún campo vacío”	
Acciones del Actor	Respuesta del Sistema

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	6.1 E sistema emite un mensaje “Todos los campos deben llenarse” y se indica en color rojo, el nombre de los campos vacíos y va a la acción 4 de de esta Sección.
Flujo alternativo 13.1 “Si el Médico no desea rectificar otra consulta”	
Acciones del Actor	Respuesta del Sistema
13.1 Se cancela la operación y finaliza el CU.	

Tabla 2.4 Descripción del caso de Uso Gestionar Consultas Evolutivas.

CU 2: Gestionar Pacientes

Nombre del Caso de Uso	Gestionar Pacientes
Actores	Médico(inicia)
Propósito	Atender al paciente, lo que implica recoger o consultar información del paciente y posibilitar que el paciente se retire de la consulta con tratamiento adecuado.
Resumen	<p>El CU se inicia cuando el Médico realizar algunas de las siguientes operaciones:</p> <ul style="list-style-type: none"> • Atender Nuevo Paciente: Es la primera vez que el paciente acude a la consulta, el Médico recoge los datos necesarios en la Historia Clínica, finalizando así el CU. • Mostrar HC: Cuando un Médico necesita ver todos los datos de las consultas de los pacientes, finalizando así el CU. • Modificar Datos: Cuando el Médico necesita

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	rectificar los datos de la historia clínica de un paciente.
Referencias	R 8 ,R 9, R 9.1,R 9.2,R 9.3,R 9.4
Precondiciones	Que haya un paciente para consultar, que el médico esté registrado en el sistema y que sea satisfactoria la autenticación del mismo para acceder al sistema.
Poscondiciones	Que el paciente se retire de la consulta con un diagnóstico, si es preciso un tratamiento así como hábitos saludables de vida, ejercicios físicos entre otras recomendaciones.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del sistema.
<p>2. El Médico selecciona cuál acción desea realizar:</p> <p>a. Atender Nuevo Paciente.</p> <p>b. Mostrar HC.</p> <p>c. Modificar Datos.</p>	<p>2. El sistema, en dependencia de la acción solicitada por el Médico, muestra la interfaz correspondiente:</p> <ul style="list-style-type: none"> • Atender Nuevo Paciente (Ir a la sección" Atender Nuevo Paciente"). • Mostrar HC (Ir a la sección" Mostrar HC"). • Modificar Datos (Ir a la sección " Modificar Datos").
Sección "Atender Nuevo Paciente".	
Acciones del Actor	Respuesta del Sistema
	<p>1. El sistema muestra una interfaz con un menú de todos los formularios que el médico</p>

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	debe llenar a un paciente.
2. El Médico selecciona el formulario que va a llenar.	3. El sistema muestra el formulario seleccionado permitiéndole al médico insertar los datos del paciente en el mismo.
4. El médico inserta los datos del paciente en el formulario y le indica al sistema guardar los mismos.	5. El sistema verifica que los datos sean correctos.
	6. El sistema guarda los valores en la base de datos.
	7. El sistema emite un mensaje confirmando que los datos fueron guardados y ofrece la posibilidad llenar los demás formularios.
8. El Médico selecciona otro formulario que desee llenar.	9. El sistema va a la acción 3 de esta Sección.
<p>Nota: Los datos correspondientes a la opción de Datos personales del paciente se encuentran en los anexos, el resto de los datos correspondientes a APP, APF, Medicamentos Tomados, Examen Físico, síntomas y otros exámenes se encuentran en el expediente de proyecto.</p>	
<p>Flujo alternativo 5.1 “Si los datos no son correctos.”</p>	
Acciones del Actor	Respuesta del Sistema
	5.1 Si los datos introducidos no son permitidos se indican donde están los errores que se cometieron para que sean corregidos. Retorna a 4.
<p>Flujo alternativo 8.1 “Si el Médico no desea llenar los demás formularios.”</p>	

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

Acciones del Actor	Respuesta del Sistema
8.1 Se cancela la operación y finaliza el CU.	
Sección “Modificar Datos”.	
Acciones del Actor	Respuesta del Sistema
	1. El sistema muestra interfaz con vínculos a todos los formularios que pueden ser modificados.
2. El Médico selecciona el formulario que desea modificar.	3. El sistema muestra el formulario seleccionado permitiéndole al médico realizar la actualización del mismo.
4. El Médico realiza la actualización de los datos e indica al sistema guardarlos.	5. El sistema verifica que los datos sean correctos.
	6. El sistema actualiza los valores en la base de datos.
	7. El sistema emite un mensaje confirmando la actualización y da la posibilidad modificar otros datos.
8. El Médico indica modificar otros datos.	9. El sistema va a la acción 3 de esta Sección.
Nata: Es el mismo formulario del llenado de Datos personales de un paciente.	
Flujo alternativo 4.1 “Si el médico decide no guardar los cambios.”	
Acciones del Actor	Respuesta del Sistema

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

4.1 El médico selecciona otro formulario.	4.2 El sistema va a la acción 3 de esta Sección.
Flujo alternativo 5.1 “Si los datos no son correctos.”	
Acciones del Actor	Respuesta del Sistema
	5.1 Si los datos introducidos no son permitidos se indican donde están los errores que se cometieron para que sean corregidos. Retorna a la acción 4.
Flujo alternativo 8.1 “Si el Médico no sea modificar los datos de otros formularios”	
Acciones del Actor	Respuesta del Sistema
8.1 El médico no selecciona otros formularios y así finaliza el caso de uso.	
Sección “Mostrar HC”.	
Acciones del Actor	Respuesta del Sistema
	1. El sistema muestra la historia clínica del paciente permitiéndole al médico imprimirla.
2. El Médico indica al sistema imprimir historia clínica.	3. El sistema imprime la historia clínica del paciente, finalizando así el caso de uso.

Tabla 2. 5 Descripción del caso de Uso Gestionar Pacientes.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

CU 3: Buscar Paciente.

Nombre del Caso de Uso	Buscar Paciente.	
Actores	Médico(inicia)	
Propósito	Encontrar un paciente existente en la Base de Datos para poder realizar cualquier acción sobre él.	
Resumen	El caso de uso se inicia cuando el Médico solicita la búsqueda de un paciente en el sistema y finaliza con la obtención de los resultados de la búsqueda.	
Referencias	R 10,R 10.1,R 10.2,R 10.3,R 10.4,R 10.5,R 10.6	
Precondiciones	Que el Médico esté registrado en el sistema y que sea satisfactoria la autenticación del mismo para acceder a la consulta. Además tiene que tener datos del paciente para poder realizar la búsqueda.	
Poscondiciones	Obtener los datos guardados en el sistema de un paciente en específico dado su historia clínica.	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del sistema.	
1. El Médico selecciona la opción "Buscar Paciente".	2. El sistema muestra una interfaz que da la posibilidad al médico de buscar un paciente a partir de los parámetros siguientes: <ul style="list-style-type: none"> • Nombre • Apellidos • Carné de Identidad 	

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

<p>3. El Médico provee los datos por los cuales puede buscar a un paciente e indica al sistema realizar la búsqueda.</p>	<p>4. El sistema verifica que los datos introducidos sean correctos.</p>
	<p>5. El sistema busca todos los pacientes que coincidan con los criterios especificados.</p>
	<p>6. El sistema muestra un listado de todos los pacientes resultantes de la búsqueda permitiéndole al médico realizar las siguientes opciones al paciente que desee:</p> <ul style="list-style-type: none"> ▪ Mostrar Historia Clínica, ir al caso de uso Gestionar Paciente a la sección “Mostrar HC”. ▪ Modificar Datos del Paciente, ir al caso de uso Gestionar Paciente a la sección “Modificar Datos”. ▪ Consulta, ir al caso de uso Gestionar Consultas Evolutivas. <p>Finalizando así el caso de uso.</p>
<p>Nota: El formulario de buscar Paciente aparece en los anexos.</p>	
<p>Flujo alternativo 5.1 “No se encuentran ningún paciente.”</p>	
<p>Acciones del Actor</p>	<p>Respuesta del Sistema</p>
	<p>5.1 Si no existen ningún paciente con estos criterios, el listado aparecerá vacío y entonces</p>

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	el médico debe seleccionar opción Atender Nuevo Paciente, ir al caso de uso Gestionar Paciente a la sección “ Atender Nuevo Paciente “ .
Prioridad:	Critico.

Tabla 2. 6 Descripción del Caso de Uso Buscar Paciente.

CU 4: Generar Reportes.

Nombre del Caso de Uso	Generar Reportes.
Actores	Médico.
Propósito	Obtener resultados de la información existente en la base de datos según los criterios que se seleccionen en la búsqueda avanzada.
Resumen	El caso de uso se inicia cuando el Usuario Autorizado solicita la opción de la Búsqueda General, el mismo selecciona los criterios para buscar y finaliza con el resultado de la búsqueda. El resultado incluye nombre, apellidos, cantidad de pacientes encontrados, fecha de ingreso, entre otros datos asociados a los pacientes, además de los criterios por los que se buscó.
Referencias	R13
Precondiciones	Estar registrado como Médico o como administrador y seleccionar la opción de Búsqueda Avanzada.
Poscondiciones	Que se muestre los resultados, asociados a los criterios seleccionados.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del sistema.

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

<p>1. El usuario autorizado selecciona la opción por la cual desea buscar:</p> <ul style="list-style-type: none"> ▪ Datos personales ▪ APP ▪ Otros 	<p>2. El sistema, en dependencia de la acción solicitada por el usuario registrado, muestra la interfaz correspondiente:</p> <ul style="list-style-type: none"> ▪ Datos personales, ir a la sección “Datos Personales”. ▪ APP, ir a la sección “APP”. ▪ Otros, ir a la sección “Otros”.
Sección “Datos Personales”	
Acciones del Actor	Respuesta del Sistema
	1. El sistema muestra la interfaz para realizar la búsqueda a partir de los datos personales.
2. El usuario registrado llena los campos que brinda el sistema y le indica a este realizar la búsqueda.	3. El sistema verifica que los datos introducidos sean correctos.
	4. El sistema realiza la búsqueda en la base de datos.
	5. El sistema muestra una interfaz con los resultados de la búsqueda.
6. El usuario registrado solicita la opción de imprimir los datos.	7. El sistema ejecuta la acción y finaliza el caso de uso.
Flujo alternativo 3.1 “Si los datos no son correctos”.	
Acciones del Actor	Respuesta del Sistema
	3.1 Si los datos introducidos no son correctos se indican donde están los errores que se

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	cometieron para que sean corregidos. Retorna a la acción 2.
Flujo alternativo 4.1 “Si no se encontró ningún paciente”.	
Acciones del Actor	Respuesta del Sistema
	4.1 El sistema emite un mensaje “La búsqueda no arrojó ningún resultado”.
Flujo alternativo 6.1 “Si no desea imprimir los resultados”.	
Acciones del Actor	Respuesta del Sistema
6.1 El usuario registrado cancela la operación y finaliza el caso de uso.	
Nota: La Sección APP y Otros se lleva a cabo el mismo flujo de eventos que en la sección Datos Personales y los formularios se encuentran en el expediente de proyecto.	

Tabla 2. 7 Descripción del Caso de Uso Generar Reportes.

CU 5: Autenticar Usuario

Nombre del Caso de Uso	Autenticar usuario.
Actores	Médico(inicia)
Propósito	Garantizar la seguridad de los usuarios al acceder al sistema.
Resumen	El Caso de Uso se inicia cuando el Usuario introduce los datos que se le piden para acceder a la aplicación, estos se verifican

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	a través de SAAA y el caso de uso finaliza cuando se habilita el acceso al sistema.
Referencias	R1
Precondiciones	Estar registrado como administrador o médico.
Poscondiciones	Obtener acceso a las funcionalidades que el rol incluya.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del sistema.
1. El usuario introduce su nombre y contraseña.	2. El sistema verifica a través del Web Server (SAAA) que los datos del registro estén en la base de datos.
	3. El sistema registra al usuario mediante los datos obtenidos del SAAA con los privilegios según el tipo de usuario y visualiza la página principal de la aplicación, finalizando así el CU.
Flujo alternativo 3.1 “ Si el usuario y la contraseña no coincide con la registrada en el Web Server”	
Acciones del Actor	Respuesta del Sistema
	2.1 El sistema muestra el mensaje: “Usuario y Contraseña no válida”.
Flujo alternativo 3.2 “ Si el usuario no existe ”	
Acciones del Actor	Respuesta del Sistema

CAPITULO II: CARACTERÍSTICAS DEL SISTEMA

	3.2 El sistema muestra el mensaje “Usuario inexistente”. Va a la acción 1 del Flujo Normal de esta sección.
Prioridad:	Crítico.

Tabla 2. 8 Descripción del Caso de Uso Autenticar Usuario.

Las descripciones de los CUS tales como Cálculo de Peso Ideal, Cálculo de dietas, Actualización de Fármacos se encuentra en los anexos, el resto en el expediente de proyecto.

2.3.2.5 MATRIZ DE TRAZABILIDAD ENTRE REQUERIMIENTOS Y CUS

Requerimientos	Autenticar usuario	Generar Reportes	Gestionar Paciente	Gestionar Consultas Evolutivas	Buscar Paciente	Actualizar Fármacos	Cambiar Rangos	Gestionar Usuario	Cal. PI	Cal. dietas	Cal. FR	Cal. IMC
R 1	X											
R 2								X				
R 3								X				
R 4									X			
R 5										X		
R 6												X
R 7											X	
R 8			X									
R 9			X									
R 9.1			X									
R 9.2			X									
R 9.3			X									
R 9.4			X									
R 10					X							
R 10.1					X							
R 10.2					X							
R 10.3					X							
R 10.4					X							
R 10.5					X							
R 10.6					X							
R 11				X								
R 12				X								
R 13		X										
R 14							X					
R 15						X						

Tabla 2.9 Matriz de trazabilidad entre requerimientos y CUS.

CONCLUSIONES.

En este capítulo se enunciaron los objetivos estratégicos de la organización y el flujo actual del negocio. Se determinaron los casos de uso, actores y trabajadores del negocio, de los cuales se realizaron los diagramas necesarios para la correcta modelación del negocio. Se mostraron las funcionalidades del sistema con el diagrama de casos de uso del sistema el cual marca el amplio alcance del mismo, en perspectivas del análisis y diseño de la aplicación SAMAD. Las descripciones del resto de los casos de usos se encuentran en los **Anexos**.

CAPÍTULO 3: ANÁLISIS Y DISEÑO DEL SISTEMA

EL flujo de trabajo de análisis y diseño tiene un papel protagónico en la fase de elaboración. Su principal objetivo es traducir los requisitos a una especificación que describa cómo implementar el sistema. Con el análisis se obtiene una visión del sistema enmarcada en el qué hace, de modo que sólo se interesa por los requisitos funcionales y el diseño es un refinamiento del análisis que tiene en cuenta los requisitos no funcionales, o sea el cómo cumple el sistema sus objetivos. El resultado final más importante de este flujo de trabajo será el modelo del diseño.

En el presente capítulo se muestran los diagramas de clases del análisis y de clases del diseño asociados a los casos de uso crítico descrito en el capítulo anterior, los restantes diagramas se incluyen en los **Anexos**.

3.1 DEFINICIÓN DE LA ARQUITECTURA

La arquitectura del sistema estará orientada a objetos, a través del framework Symfony el cual *está basado en un patrón clásico del diseño web conocido como arquitectura MVC* formado por tres niveles:

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

Estos niveles estarán regidos por patrones de diseño, los cuales se explican a continuación:

➤ **Modelo**

El modelo se puede dividir en la capa de acceso a los datos y en la capa de abstracción de la base de datos, para lograr que las funciones que acceden a los datos no utilicen sentencias ni consultas que dependan de una base de datos, sino que utilicen otras

funciones; para realizar las consultas, Symfony emplea los patrones de diseño Active Record y Active Table:

Active Record: Representa de forma Orientada a Objetos los datos de una Base de Datos Relacional - modelo conocido también como ORM o “Object-Relational Mapping” - , definiendo interfaces sencillas para acceder y manipular esos datos.

Active Table: El Active Table es un objeto que puede persistir otros objetos en medios no volátiles, como bases de datos. La función de una clase que implementa este patrón es de comunicarse con la base de datos y regresarnos objetos como los tenemos definidos para nuestra aplicación.

Así, si se cambia de sistema gestor de bases de datos, solamente es necesario actualizar la capa de abstracción de la base de datos.

➤ Vista

La vista también puede aprovechar la separación de código, las páginas web suelen contener elementos que se muestran de forma idéntica a lo largo de toda la aplicación, por este motivo en Symfony la vista se separa en un layout y en una plantilla. Normalmente, el layout es global en toda la aplicación o al menos en un grupo de páginas, el contenido de la plantilla se integra en el layout, o si se mira desde el otro punto de vista, el layout decora la plantilla. Para realizar este comportamiento Symfony utiliza el patrón de diseño llamado “**Decorator**”.

➤ Controlador

El controlador es el componente que recibe los requerimientos, los envía a los elementos encargados de procesar la lógica y luego los envía nuevamente a la vista (esta vez incluyendo los datos obtenidos) a través del patrón de diseño '**Front Controller**'.

El controlador normalmente se divide en un controlador frontal, que es único para cada aplicación, y las acciones, que incluyen el código específico del controlador de cada página. Una de las principales ventajas de utilizar un controlador frontal es que ofrece un punto de entrada único para toda la aplicación. Así, en caso de que sea necesario impedir el acceso a la aplicación, solamente es necesario editar el script correspondiente

al controlador frontal, si no tuviéramos un punto de acceso único, tendríamos los siguientes problemas:

- Código duplicado.
- Manejo no centralizado de vistas.

Symfony toma lo mejor de la arquitectura Modelo-Vista-Controlador (MVC) y la implementa de forma que el desarrollo de aplicaciones sea rápido y sencillo. En primer lugar, el controlador frontal y el layout son comunes para todas las acciones de la aplicación, El controlador frontal es un componente que sólo tiene código relativo al MVC, por lo que no es necesario crear uno, ya que Symfony lo genera de forma automática. Las clases de la capa del modelo también se generan automáticamente, en función de la estructura de datos de la aplicación. La librería Propel se encarga de esta generación automática, ya que crea el esqueleto o estructura básica de las clases y genera automáticamente el código necesario. La abstracción de la base de datos es completamente invisible al programador, ya que la realiza otro componente específico llamado Creole. Así, si se cambia el sistema gestor de bases de datos en cualquier momento, no se debe reescribir ni una línea de código, ya que tan sólo es necesario modificar un parámetro en un archivo de configuración.

Symfony utiliza otro patrón de diseño conocido como “**Singleton**”, es una clase que únicamente permite que exista simultáneamente una única instancia de si misma y que ofrece un punto de acceso común a ella.

En Symfony se trata de un objeto muy útil que guarda una referencia a todos los objetos del núcleo de Symfony. Ejemplos: sfResponse, sfRequest, sfUser.

En la siguiente figura se muestra el flujo de trabajo de Symfony.

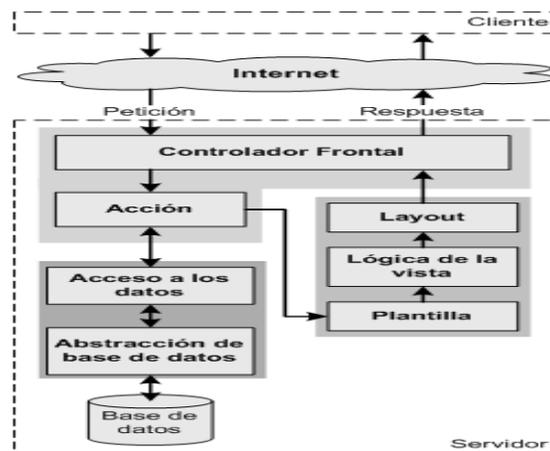


Fig. 3.1. Flujo de Trabajo del Symfony

Otros patrones de diseño son los conocidos patrones GRASP que describen los principios fundamentales de la asignación de responsabilidades a objetos, expresados en forma de patrones. *GRASP es un acrónimo que significa General Responsibility Assignment Software Patterns (Patrones Generales de Software para Asignar Responsabilidades)* (5). Estos fueron los patrones que en conjunto con los patrones GoF mencionados anteriormente se aplicaron para realizar el diseño de la aplicación que propone la investigación.

- **Experto:** Este patrón es uno de los más utilizados relacionados con el framework utilizado, puesto que Propel es la librería externa que utiliza Symfony para realizar su capa de abstracción en el modelo, encapsula toda la lógica de los datos y son generadas las clases con todas las funcionalidades comunes de las entidades, por ejemplo en el caso de uso Gestionar Consultas Evolutiva la entidad ConsultaE encapsula toda la lógica de los datos relacionados con la gestión de consultas.
- **Creador:** El patrón Creador guía la asignación de responsabilidades relacionadas con la creación de objetos, tarea muy frecuente en los sistemas orientados a objetos. El propósito fundamental de este patrón es encontrar un creador que debemos conectar con el objeto producido en cualquier evento. Al

escogerlo como creador, se da soporte al bajo acoplamiento por ejemplo en la clase Actions se encuentran las acciones definidas para SAMAD y se ejecutan cada una de ellas. En las acciones se crean los objetos de las clases que representan las entidades, evidenciando de este modo que la clase Actions es "creador" de dichas entidades, por ejemplo en el caso de uso Actualizar Fármacos la clase ActualizarF_Actions es creadora de la entidades(Fármacos) que se encuentran en el modelo .

- **Alta Cohesión:** Asignar una responsabilidad de modo que la cohesión siga siendo alta. Una clase con mucha cohesión es útil porque es bastante fácil darle mantenimiento, entenderla y reutilizarla. La ventaja que significa una gran funcionalidad también soporta un aumento de la capacidad de reutilización. Symfony permite asignar responsabilidades con una alta cohesión, por ejemplo la clase Actions tiene la responsabilidad de definir las acciones para las plantillas y colabora con otras para realizar diferentes operaciones, instanciar objetos y acceder a las properties, es decir, está formada por diferentes funcionalidades que se encuentran estrechamente relacionadas proporcionando que el software sea flexible frente a grandes cambios.
- **Bajo Acoplamiento:** Asignar una responsabilidad para mantener bajo acoplamiento. Las clases deben comunicarse con un número pequeño de clases tanto como sea posible. Ejemplo en el caso de uso Generar Reportes la clase Reportes_Actions hereda solamente de sfActions para lograr un bajo acoplamiento de clases.
- **Controlador:** Asignar la responsabilidad del manejo de un mensaje de los eventos de un sistema a una clase. En la investigación se aplican todos los patrones básicos de GRASP en los diagramas de clases del diseño. Todas las peticiones Web son manejadas por un solo controlador frontal (sfActions), que es el punto de entrada único de toda la aplicación en el entorno determinado. Cuando el controlador frontal recibe una petición, utiliza el sistema de enrutamiento para asociar el nombre de una acción y el nombre de un módulo. Ejemplo en el caso de uso Gestionar Paciente todas las peticiones deben pasar

por la clase HC_Actions la cual será el único punto de entrada de la aplicación en este módulo.

3.2 MODELO DE ANÁLISIS

Con el flujo de trabajo análisis se obtiene una visión general del futuro sistema en función de los requisitos funcionales anteriormente definidos. En dicha etapa se analizan cada caso de uso siendo transformados a un lenguaje técnico para obtener la primera aproximación al diseño e incluso se define la arquitectura. De forma general, en esta etapa se han definido por caso de uso todas las clases de interfaz externa expresando la interacción del usuario con el sistema en dependencia del nivel jerárquico que ocupe, una clase de control que encapsula funciones complejas para el manejo de la información almacenada mostrando el comportamiento de los casos de usos y se definen también todas las clases entidades que modelan la información utilizada por el sistema.

En el análisis el principal objetivo es refinar y estructurar los requisitos, lo que conlleva a un estudio profundo de los mismos.

Las clases del análisis se clasifican en tres tipos y cada clase posee su estereotipo y sus características específicas. (5) Según RUP, se definen como se muestra a continuación:

- **Interfaz:** son las clases que se utilizan para modelar la interacción entre el sistema y sus actores (es decir, usuarios y sistemas externos) y cada clase Interfaz debe relacionarse al menos con un actor y viceversa.

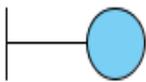


Fig. 3.2 Clase Interfaz.

- **Control:** son las clases que se utilizan para representar coordinación, secuencia, transacciones y control de otros objetos y se usan con frecuencia para encapsular el control de un caso de uso en concreto.



Fig. 3.3 Clase Control.

- **Entidad:** son las clases que se utilizan para modelar información que posee una vida larga y que es a menudo persistente. Modelan la información y el comportamiento asociado de algún fenómeno o concepto, como una persona, un objeto del mundo real, o un suceso del mundo real.

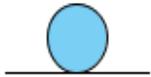


Fig. 3.4 Clase Entidad.

3.2.1 DIAGRAMA DE CLASES DEL ANÁLISIS (DCA)

- **Diagrama de Clases del Análisis : Gestionar Consulta Evolutiva**

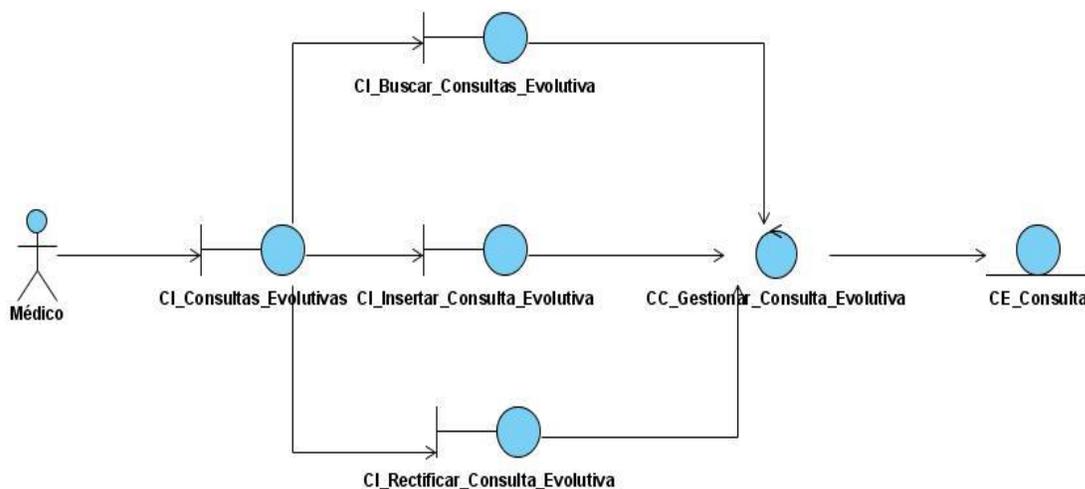


Fig. 3.5 Diagrama de Clases del Análisis Gestionar Consulta Evolutiva asociado al actor médico.

- **Diagrama de Clases del Análisis: Gestionar Datos del Paciente**

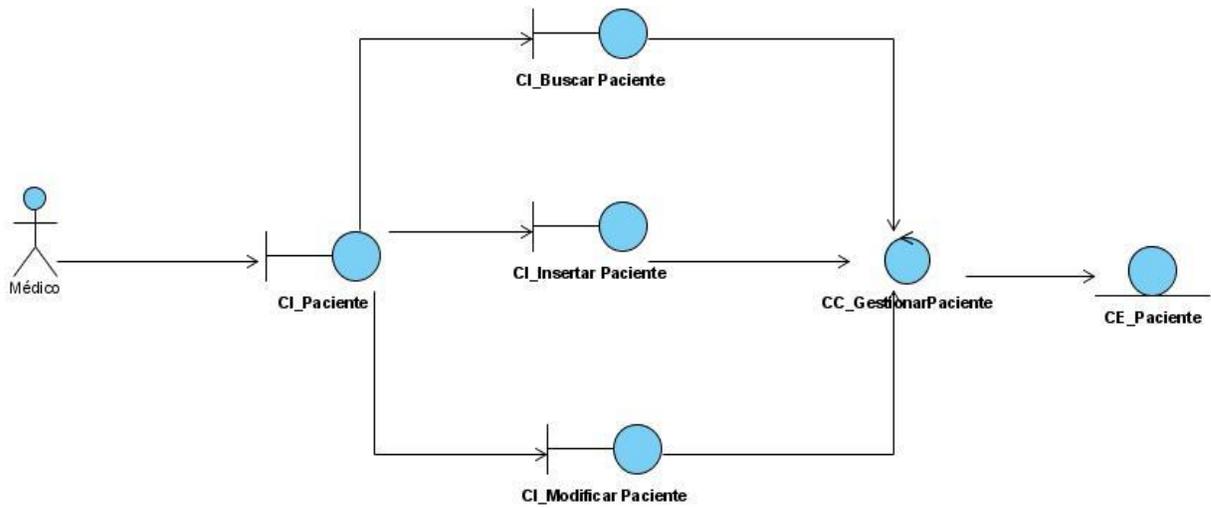


Fig. 3.6 Diagrama de Clases del Análisis Gestionar Datos del Paciente asociado al médico.

➤ Diagrama de Clases del Análisis: Generar Reportes

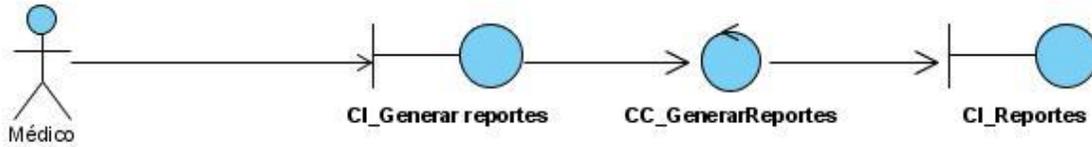


Fig. 3.7 Diagrama de Clases del Análisis Generar Reportes asociado al Médico.

➤ Diagrama de Clases del Análisis: Autenticar Usuario

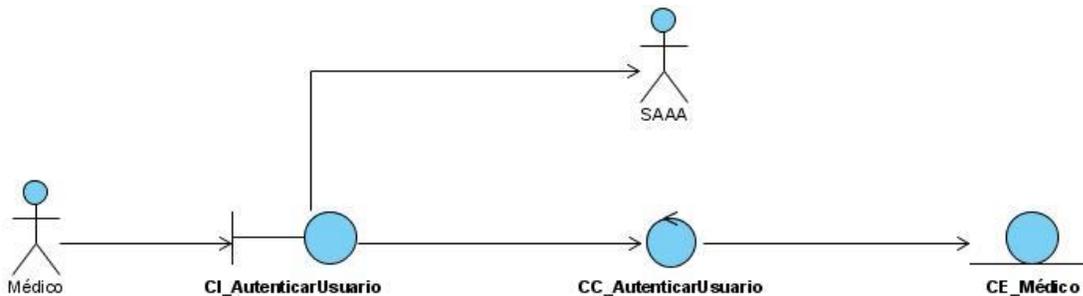


Fig. 3.8 Diagrama de Clases del Análisis Autenticar Usuario asociado al médico.

En el modelo de análisis realizado para una posterior implementación de SAMAD sólo se ha tenido en cuenta los diagramas de clases de análisis de los casos de usos significativos, el resto de los diagramas aparecen en los **Anexos**. A diferencia del análisis, el diseño sí muestra una visión más certera del sistema que se desea alcanzar, es por ellos que en este se realizan tanto los diagramas de clases del diseño como los diagramas de interacción. En este punto de la ingeniería (el diseño) sí es de gran importancia y necesidad del modelado de la parte dinámica del interior del sistema, ya que de ahí se obtiene el flujo de comunicación entre los objetos que se necesitan implementar en SAMAD.

3.3 MODELO DE DISEÑO

Con el flujo de trabajo diseño se modela el futuro sistema que de soporte a los requisitos funcionales planteados en la etapa de la captura de requisitos. También se aterrizan

todos los aspectos relacionados con las restricciones y características del sistema como lo son el lenguaje de programación a utilizar, el sistema operativo donde se podrá ejecutar la aplicación, las tecnologías de interfaz de usuario, en fin, agrupar en el diseño los requerimientos no funcionales definidos.

En el diseño al igual que en el análisis los trabajadores construyen artefactos muy importantes para describir el sistema a implementar. *Los diagramas de clases del diseño y de interacción en el diseño se comportan de igual forma que en el análisis, aunque tienen un nivel más detallado de descripción de las clases, ya que se tiene en cuenta el lenguaje de programación que se utilizará y se definen los métodos para cada clase.* (5) Además de estos artefactos los trabajadores también construyen: subsistemas de diseño, diagrama de clases persistentes y los modelos de datos para obtener la propuesta de la base de datos del sistema.

3.3.1 EXTENSIÓN PARA EL MODELADO DE APLICACIONES WEB.

El artefacto subsistema de diseño es una forma de organizar los artefactos del modelo de diseño en piezas más manejables, puede constar de clases de diseño, realizaciones de casos de uso, interfaces y otros subsistemas.

En el caso del presente trabajo se lleva a cabo el diseño para una aplicación Web siendo la aproximación más concreta del diagrama de clases del análisis y utilizándose los estereotipo para las clases en función de la aplicación Web que es una extensión que propone UML (3). Los estereotipos son los que se muestran a continuación .

- **Server Page:** Representa la página Web que tiene código que se ejecuta en el servidor. Este código interactúa con recursos en el servidor. Las operaciones representan las funciones del código y los atributos las variables visibles dentro del alcance de la página. Esta clase sólo puede tener relaciones con objetos en el servidor, una relación 1:1 con un fichero en el servidor. En las aplicaciones en PHP se corresponde con un fichero .php



Fig. 3.9 <<Server Page>>

- **Client Page:** Una instancia de una Página Cliente es una página Web, con formato HTML; mezcla de datos, presentación y lógica. Cada página cliente solo puede ser construida por una página servidor.



Fig. 3.10 <<Client Page>>

- **Form:** Grupo de elementos de entrada que son parte de una página cliente. Se relaciona directamente con la etiqueta de igual nombre del HTML. Sus atributos son los elementos de entrada del formulario (Text Field, Text Area, Button, Label, Radio Button, Radio Group, Select, Check Box y Hidden Fields).



Fig. 3.11 <<Form>>

- **Build:** Representa una asociación especial que relaciona las páginas cliente con las páginas servidor, de forma general se expresa como que las páginas que se encuentran en el servidor construyen las páginas en el cliente. Debe ser una relación direccional, donde una página servidor puede construir una o más páginas cliente.



- **Submit:** Es la relación que se crea siempre entre una página servidor y un formulario, a través de esta relación el formulario manda los valores de sus campos al servidor, para ser procesados por la página servidor.



3.3.2 DIAGRAMAS DE CLASES DEL DISEÑO (DCD)

➤ Diagrama de Clases del Diseño: Gestionar Consulta Evolutiva

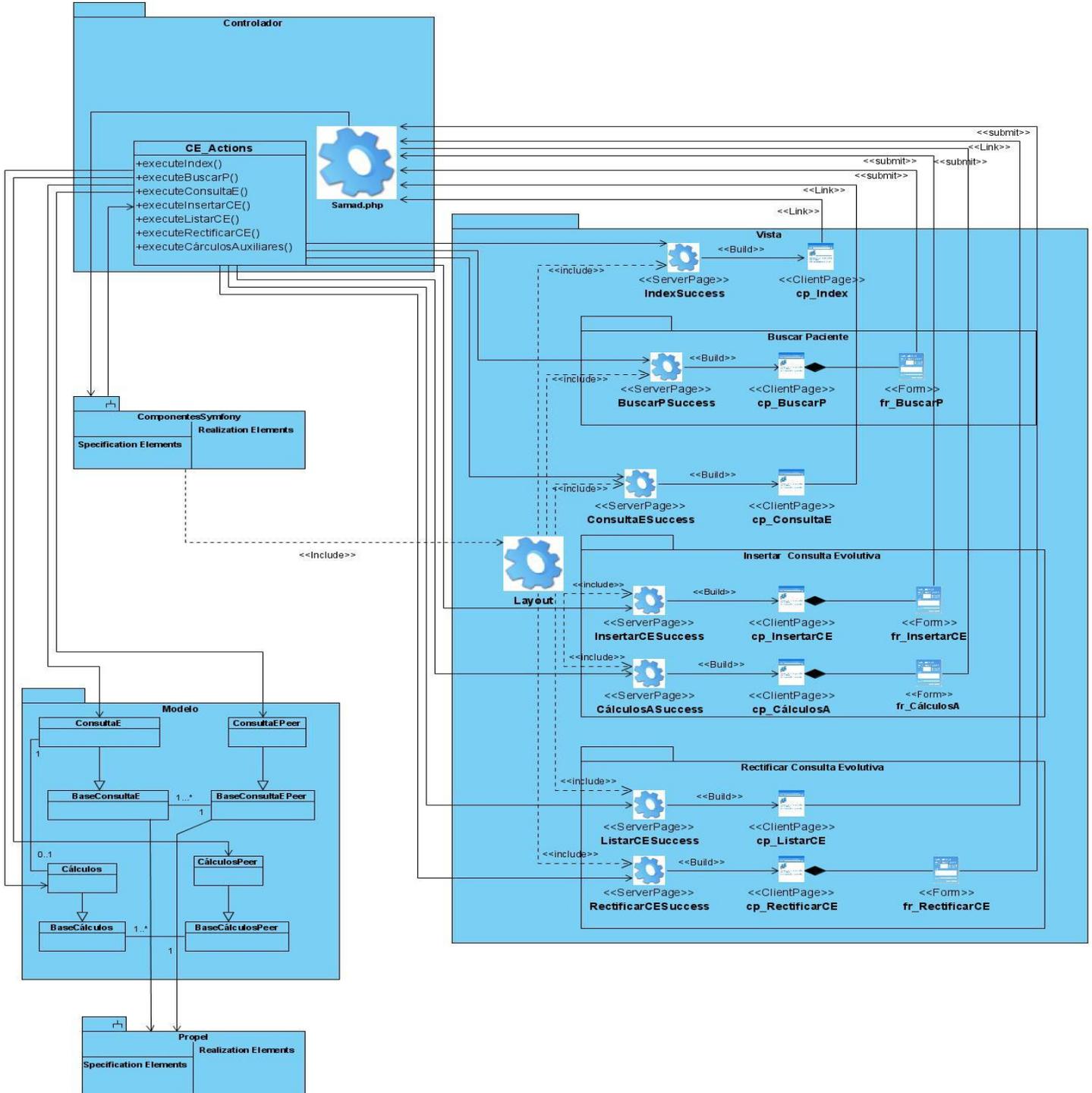


Fig. 3.14 Diagrama de Clases del Diseño: Gestionar Consulta Evolutiva

➤ Diagrama de Clases del Diseño: Gestionar Paciente

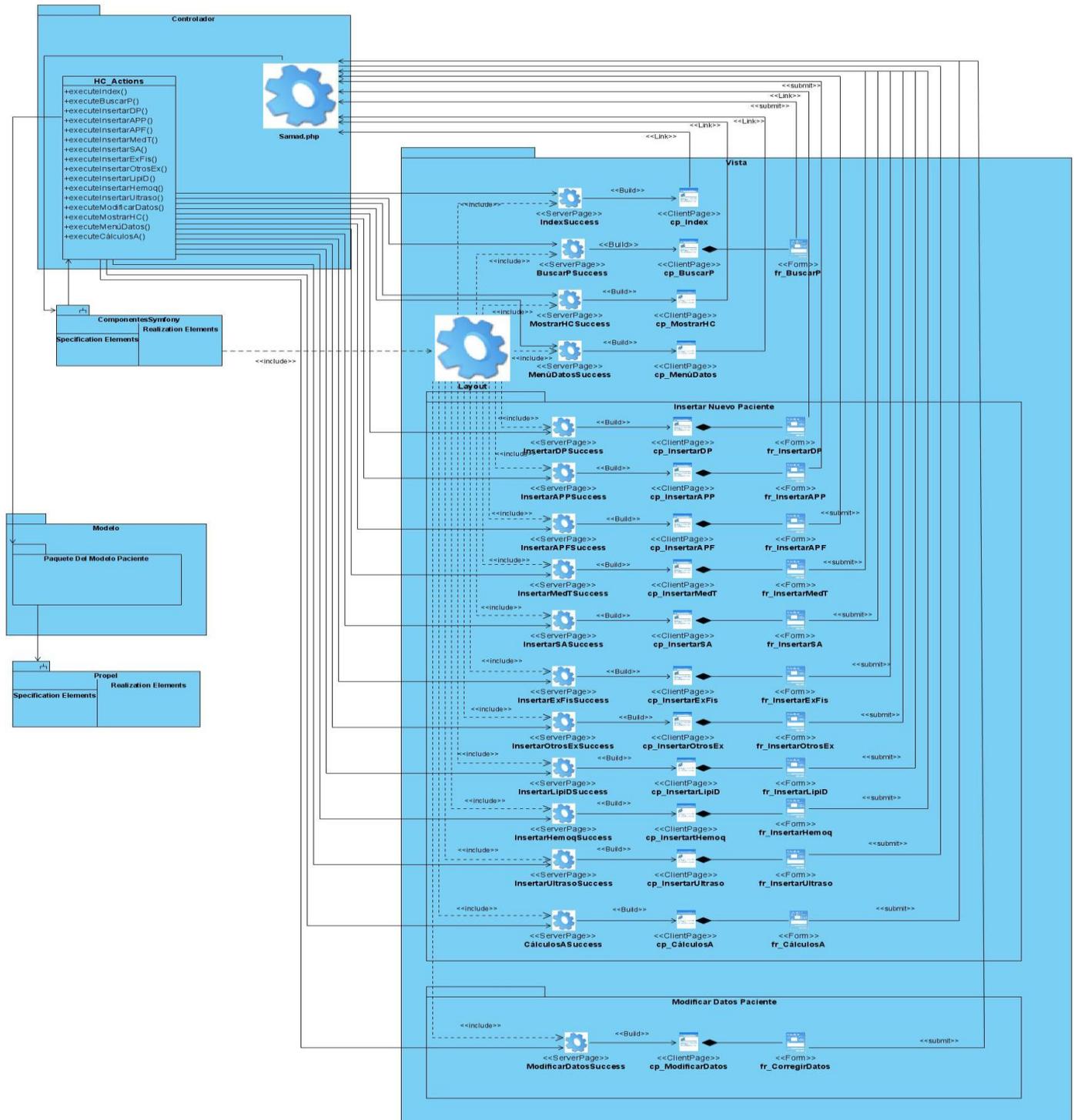


Fig. 3.15 Diagrama de Clases del Diseño: Gestionar Paciente

➤ Diagrama de Clases del Diseño: Generar Reportes

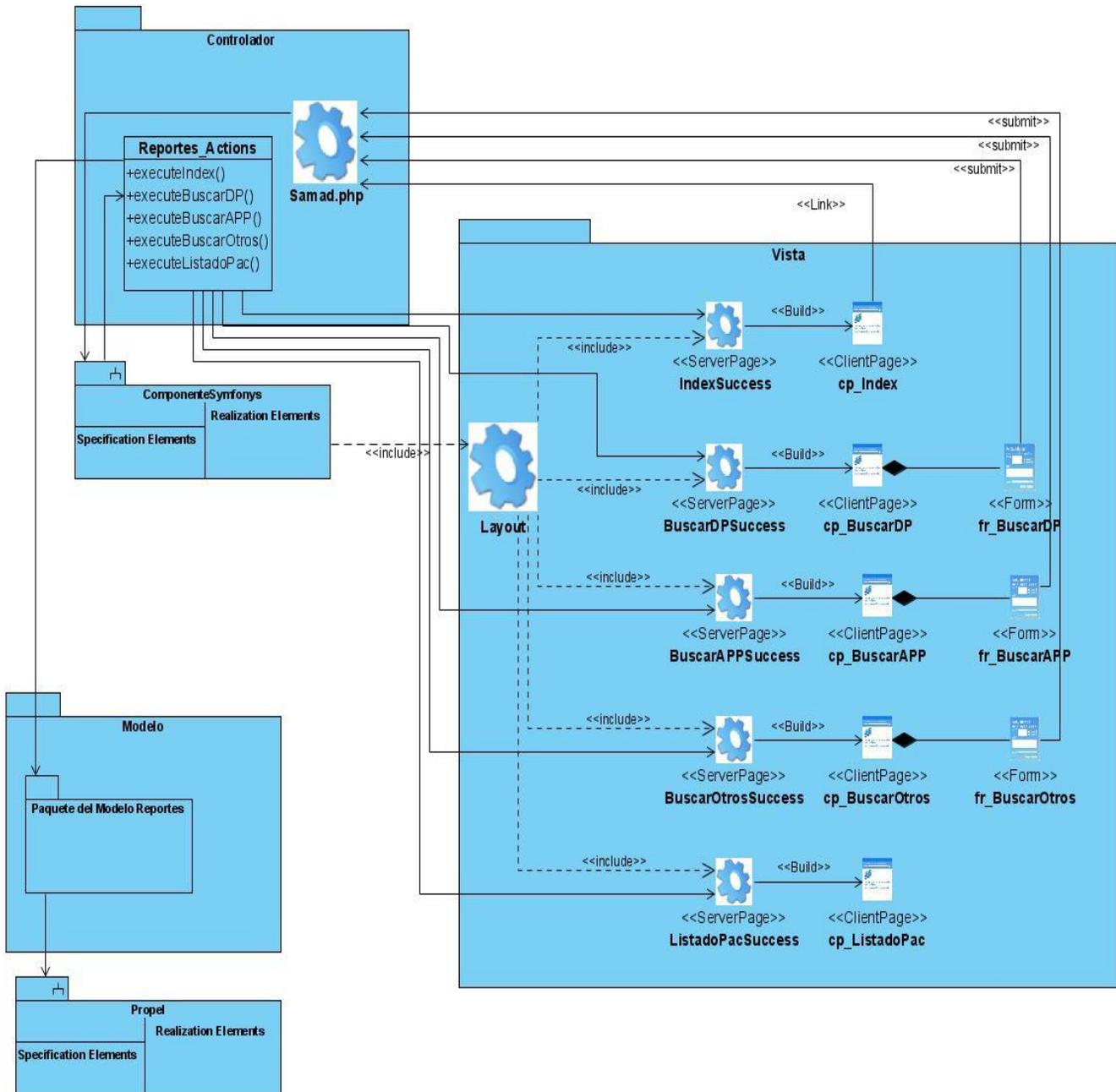


Fig. 3.16 Diagrama de Clases del Diseño: Generar Reportes

DESCRIPCIONES DE LAS CLASES DEL DISEÑO

GESTIONAR PACIENTE

Nombre: HC_Actions	
Tipo: Clase (Esta clase contiene todas las funciones correspondientes para la recogida de datos de un paciente nuevo. Todas las funciones van a utilizar los métodos relacionados con el paquete Modelo).	
Atributos-tipo (Datos Personales) hc - varchar , código- varchar , nombre- varchar , apellidos- varchar , fechanac-varchar , calle- varchar, entre- varchar, no- varchar , apto- varchar, localidad- varchar, municipio-varchar, provincia- varchar, pais- varchar, fechaing-varchar , edad-char , sexo- char El resto de los datos APP, APF, Medicamentos Tomados, Examen Físico, síntomas y otros exámenes se encuentran en el expediente de proyecto.	
Responsabilidades: Insertar datos del Paciente	
Nombre:	Descripción:
executeBuscarP	Da la opción de realizar una la búsqueda de un paciente a partir del nombre, apellidos y carnet de identidad, permitiendo mostrar los resultados de la búsqueda. En el BuscarP se van a utilizar varios métodos de las clases Peer para la buscar y mostrar los resultados dentro de los cuales podemos mencionar: Do Select, SendJoin.
executeMenuDatos()	Muestra un menú a partir del cual podrá acceder al llenado todos los formularios correspondientes a un paciente que sea nuevo.

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

executeInsertarDP()	Muestra un formulario permitiéndole al médico llenar todos los datos personales de un paciente y llama a la función InsertarDP() para guardar los datos introducidos por el médico. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarAPP()	Muestra un formulario permitiendo llenar los antecedentes patológicos de un paciente y llama a la función InsertarAPP() para guardar los datos introducidos por el médico. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarAPF()	Muestra un formulario permitiendo insertar los antecedentes familiares de un paciente y llama a la función InsertarAPF() para guardar los datos introducidos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarMedT()	Muestra un formulario permitiendo la inserción de los medicamentos tomados por un paciente y llama a la función InsertarMedT () para guardar los datos introducidos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarSA()	Permite introducir los síntomas asociados de un paciente y llama a la función InsertarMedT() para guardar los datos introducidos. Para la implementación de esta función se van a utilizar

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

	métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarExFis()	Permite introducir los exámenes físicos de un paciente y llama a la función InsertarExFis() para guardar los datos introducidos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarOtrosEx()	Permite introducir otros exámenes a un paciente y llama a la función InsertarOtrosEx() para guardar los datos introducidos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarLipiD()	Permite introducir los resultado del lipidograma y llama a la función InsertarLipiD() la cual va a generar y mostrar un diagnóstico y un tratamiento a partir de los datos introducidos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria.
executeInsertarHemoq()	Permite introducir datos hemoquímicos a un paciente y llama a la función InsertarHemoq() para guardar los datos introducidos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeInsertarUltrasono()	Permite introducir los resultado de Ultrasonido de un paciente y llama a la función InsertarUltrasono() para guardar los datos introducidos. Para la

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

	implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeCalculosA()	Permite realizar cálculos un paciente determinado a partir de los datos introducidos y llama a la función CálculosA() para guardar los resultados. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
executeModificarDatos()	Llama a la función ModificarDatos() para mostrar todos los datos de un paciente ya existe, permitiendo la modificar los que tengan errores y guardará los nuevos valores. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria, entre Otros.
executeMostrarHC()	Llama al función MostrarHC() a través de la cual se va a realizar una búsqueda de las consultas y datos de un paciente seleccionado. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria, entre Otros.
[validar formulario]	Permite las validaciones que no son posibles de realizar en los archivos .yml, haciéndose directamente contra la BD.
hasErrorMetodo()	Este método se ejecuta cuando el archivo codificarmf.yml o la función validate correspondiente al execute lanzan un error, el resultado es que no se envíe el formulario y se muestren los mensajes

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

	de errores en la página. Se encarga de mantener la estructura de la página con su información.
--	--

Tabla 3.1 CUS: Gestionar Paciente

GESTIONAR CONSULTA EVOLUTIVA

Nombre: CE_Actions	
Tipo: Clase (Esta clase contiene todas las funciones correspondientes al paquete del diseño Consulta Evolutiva. Todas las funciones van a utilizar los métodos relacionados con el paquete Modelo).	
Atributo-Tipo Numero-varchar, fecha-varchar, tipo-int, diagnostico-varchar, tratamiento-varchar, recomendaciones-varchar	
Responsabilidades: Insertar Consulta Evolutiva Rectificar Consulta Evolutiva	
Nombre:	Descripción:
executeBuscarP()	Permite realizar la búsqueda de un paciente a partir del nombre, apellidos y carnet de identidad, permitiendo mostrar los resultados de la búsqueda. En la función BuscarP() se van a utilizar varios métodos de las clases Peer los cuales buscan y muestran los resultados, alguno de ellos son: Do Select, Sendjoin.
executeConsultaE()	Muestra un menú a partir del cual podrá Insertar o rectificar los datos de una consulta del paciente

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

	previamente buscado.
executeInsertarCE()	Permite introducir los resultado de los exámenes y llama a la función InsertarDatosCE() la cual va a generar y mostrar un diagnóstico y un tratamiento a partir de los datos introducidos. Esta función también va a guardar todos los datos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria.
executeListarCE()	Esta función lista todas las consultas evolutivas y permite seleccionar una para rectificar sus datos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria, entre Otros.
executeRectificarCE()	Esta función muestra la consulta con los datos que posee y permite que estos sean modificados y posteriormente guardados. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria, entre Otros.
executeCálculosAuxiliares()	Permite realizar cálculos un paciente determinado a partir de los datos introducidos y llama a la función CalculosAuxiliares() para guardar los resultados. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.
[validar formulario]	Permite las validaciones que no son posibles de realizar en los archivos .yml, haciéndose directamente contra la BD.

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

hasErrorMétodo()	Este método se ejecuta cuando el archivo codificarmf.yml o la función validate correspondiente al execute lanzan un error, el resultado es que no se envíe el formulario y se muestren los mensajes de errores en la página. Se encarga de mantener la estructura de la página con su información.
------------------	--

Tabla 3.2 CUS: Gestionar consulta evolutiva

Generar Reportes

Nombre: Reportes_Actions	
Tipo: Clase (Esta clase contiene todas las funciones correspondientes al paquete del diseño Generar Reportes. Todas las funciones van a utilizar los métodos relacionados con el paquete Modelo).	
Atributos-Tipo (generar reportes DP) nombre- varchar , apellidos- varchar , fechanac-varchar , calle- varchar, entre- varchar, no- varchar , apto- varchar, localidad- varchar, municipio- varchar, provincia- varchar, pais-varchar, sexo- char ,teléfono-varchar El resto de los atributos y valores de la búsqueda avanzada se encuentran en el expediente de proyectos.	
Responsabilidades: Realizar Búsquedas avanzadas	
Nombre:	Descripción:
executeBuscarDP()	Permite realizar la búsqueda de un paciente a partir de los datos personales como nombre, apellidos, país, provincia, municipio y sexo permitiendo mostrar los resultados de la búsqueda. En la

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

	función BuscarDP() se van a utilizar varios métodos de las clases Peer los cuales buscan y muestran los resultados, alguno de ellos son: Do Select.
executeBuscarAPP()	Permite realizar la búsqueda de un paciente a partir de los antecedentes patológicos personales como tiempo de padecimiento, ingestión de bebidas alcohólicas, diabetes mellitus, hipertensión arterial, tabaquismo, cardiopatía isquémica, enfermedades renales, y otras, permitiendo mostrar los resultados de la búsqueda, los cuales se van a mostrar a través del Do Select
executeBuscarOtros()	Permite realizar la búsqueda de un paciente a partir de otros exámenes de LDL permitiendo mostrar los resultados de la búsqueda.
executeListadoPac()	Esta función lista todos los pacientes que están en la Base de datos relacionados con algún tipo de búsqueda que se este realizando. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criterias, entre Otros.

Tabla 3.3 CUS: Generar reportes

Autenticar Usuario

Nombre: AutenticarU_Actions
Tipo: Clase (Esta clase contiene la funciones a través de la cual se podrá acceder al sistema con los permisos establecidos. Esta función van a utilizar los métodos del WebServer SAAA si esta funcionando y sino va utilizar los métodos relacionados con el

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

paquete Modelo).	
Nota: Para el caso que el WebServer no este funcionando se va hacer una copia de la base de datos del mismo en el sistema.	
Atributo-Tipo	
usuario- varchar, contraseña-varchar	
Responsabilidades:	
Nombre:	Descripción:
executeAutenticar()	Permite realizar la búsqueda de un paciente a partir del nombre, apellidos y carnet de identidad, permitiendo mostrar los resultados de la búsqueda. En la función BuscarP() se van a utilizar varios métodos de las clases Peer los cuales buscan y muestran los resultados, alguno de ellos son: Do Select, Sendjoin.
executeConsultaE()	Muestra un menú a partir del cual podrá Insertar o rectificar los datos de una consulta del paciente previamente buscado.
executeInsertarCE()	Permite introducir los resultado de los exámenes y llama a la función InsertarDatosCE() la cual va a generar y mostrar un diagnóstico y un tratamiento a partir de los datos introducidos. Esta función también va a guardar todos los datos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria.
executeListarCE()	Esta función lista todas las consultas evolutivas y permite seleccionar una para rectificar sus datos.

CAPITULO III: ANÁLISIS Y DISEÑO DEL SISTEMA

	Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria, entre Otros.
executeRectificarCE()	Esta función muestra la consulta con los datos que posee y permite que estos sean modificados y posteriormente guardados. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save, Criteria, entre Otros.
executeCálculosAuxiliares()	Permite realizar cálculos un paciente determinado a partir de los datos introducidos y llama a la función CálculosAuxiliares() para guardar los resultados. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: Do Select, Set, Save.

Tabla 3.4 CUS: Autenticar usuario

Las descripciones de las clases del diseño de los CUS tales como Cálculo de Peso Ideal, Cálculo de dietas, Actualización de Fármacos se encuentra en los anexos, el resto en el expediente de proyecto.

3.3.3 DIAGRAMAS DE SECUENCIAS (DS)

➤ Diagrama de Secuencia: Gestionar Consulta Evolutiva {Insertar Consulta}

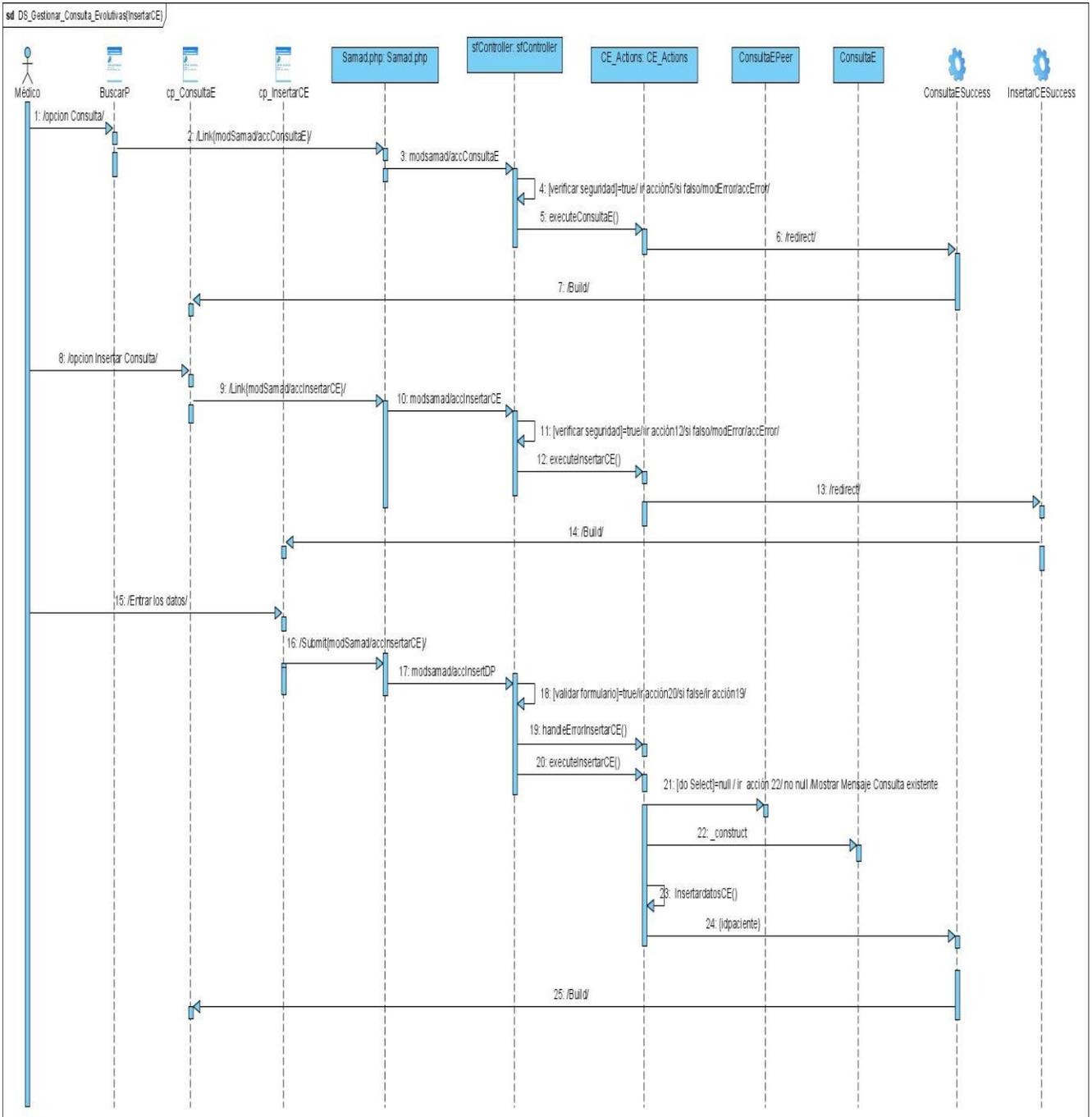


Fig. 3.18 Diagrama de Secuencia: Gestionar Consulta Evolutiva {Insertar Consulta}

➤ Diagrama de Secuencia: Gestionar Consulta Evolutiva {Rectificar Consulta}

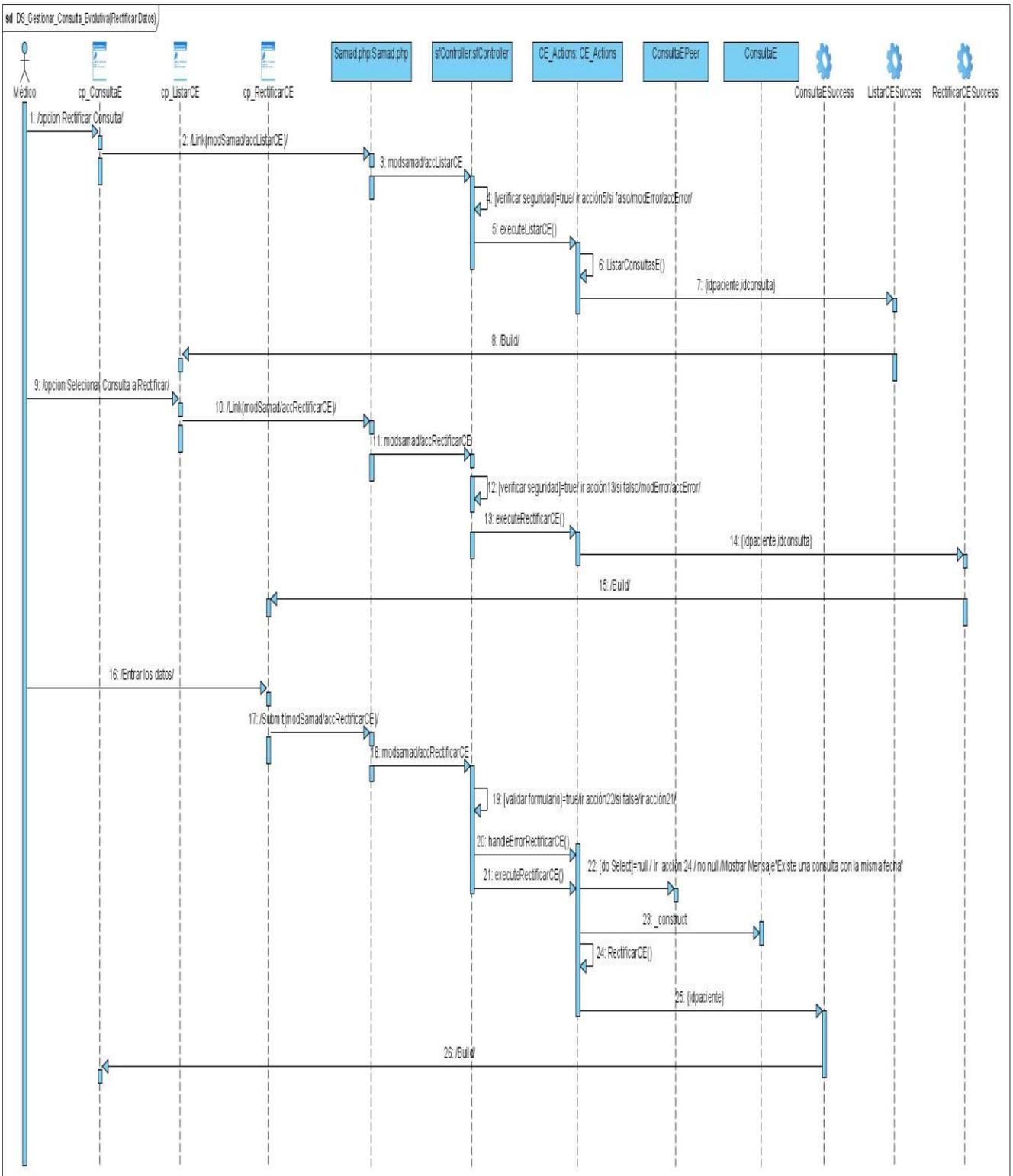


Fig.3.19 Diagrama de Secuencia: Gestionar Consulta Evolutiva {Rectificar Consulta}

➤ Diagrama de Secuencia: Gestionar Paciente {Insertar Datos Personales }

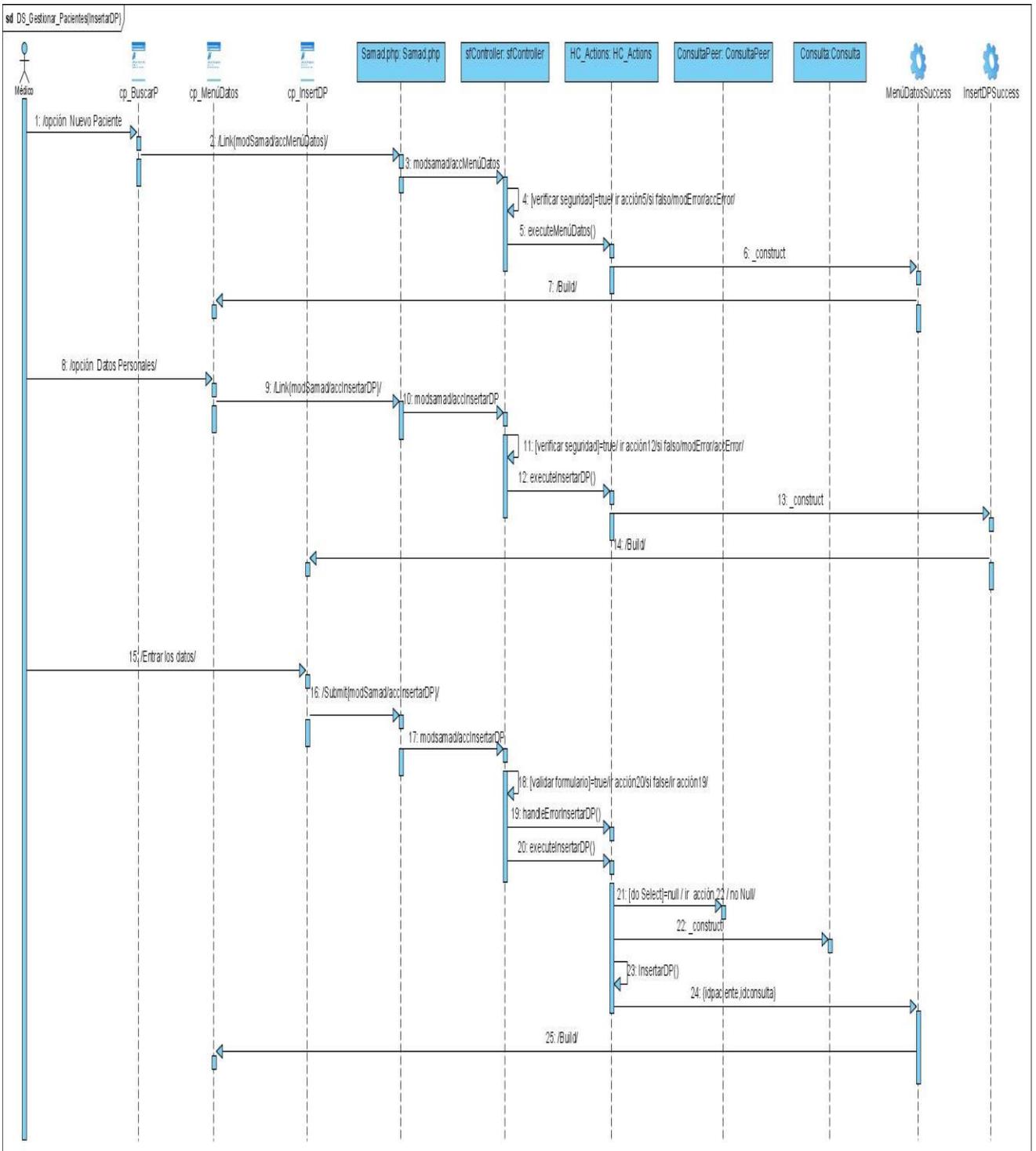


Fig.3.20 Diagrama de Secuencia: Gestionar Paciente {Insertar Datos Personales}

➤ **Diagrama de Secuencia: Buscar Paciente**

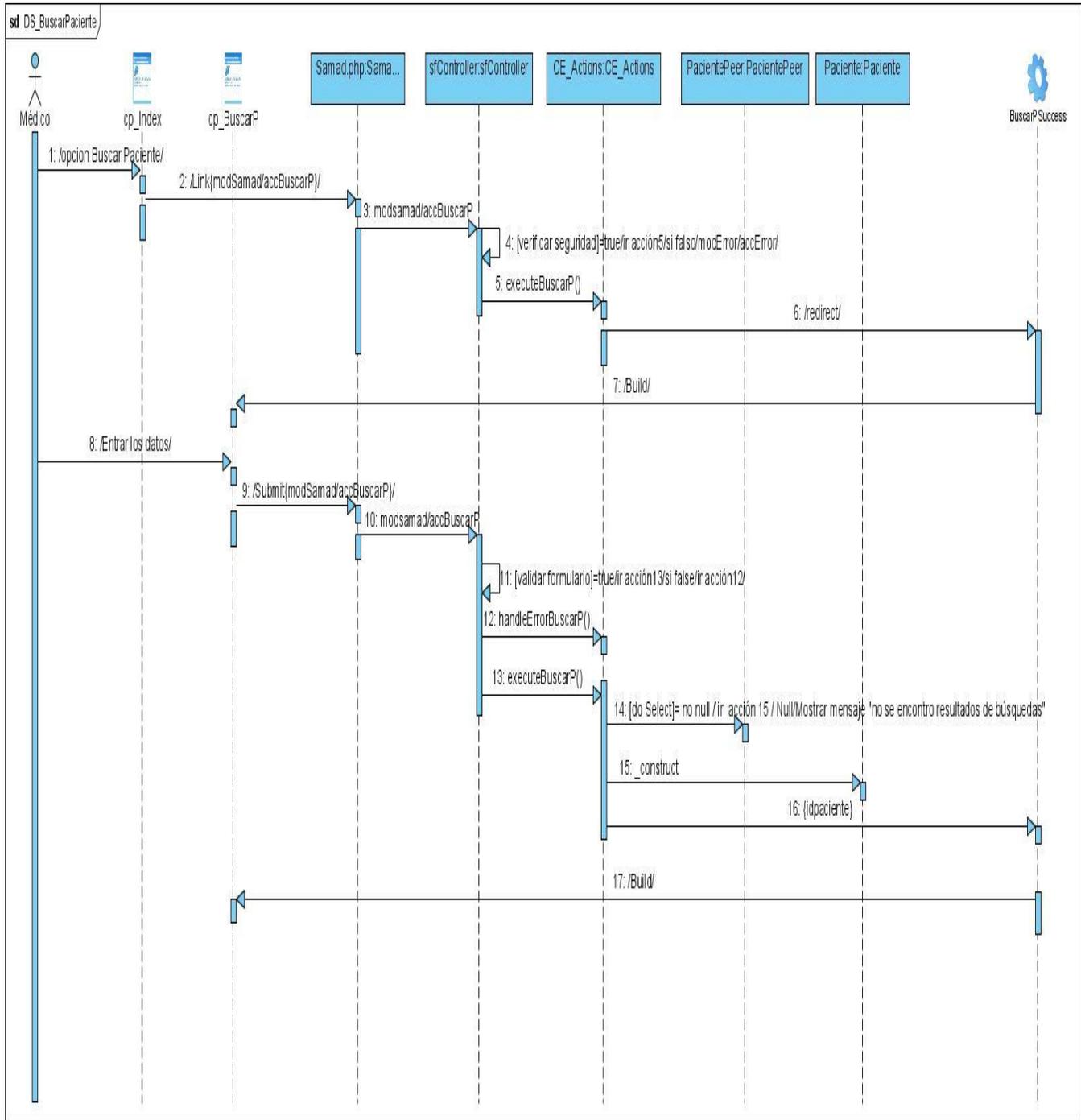


Fig.3.21 Diagrama de Secuencia: Buscar Paciente

➤ Diagrama de Secuencia: Generar Reportes {Generar Datos Personales}

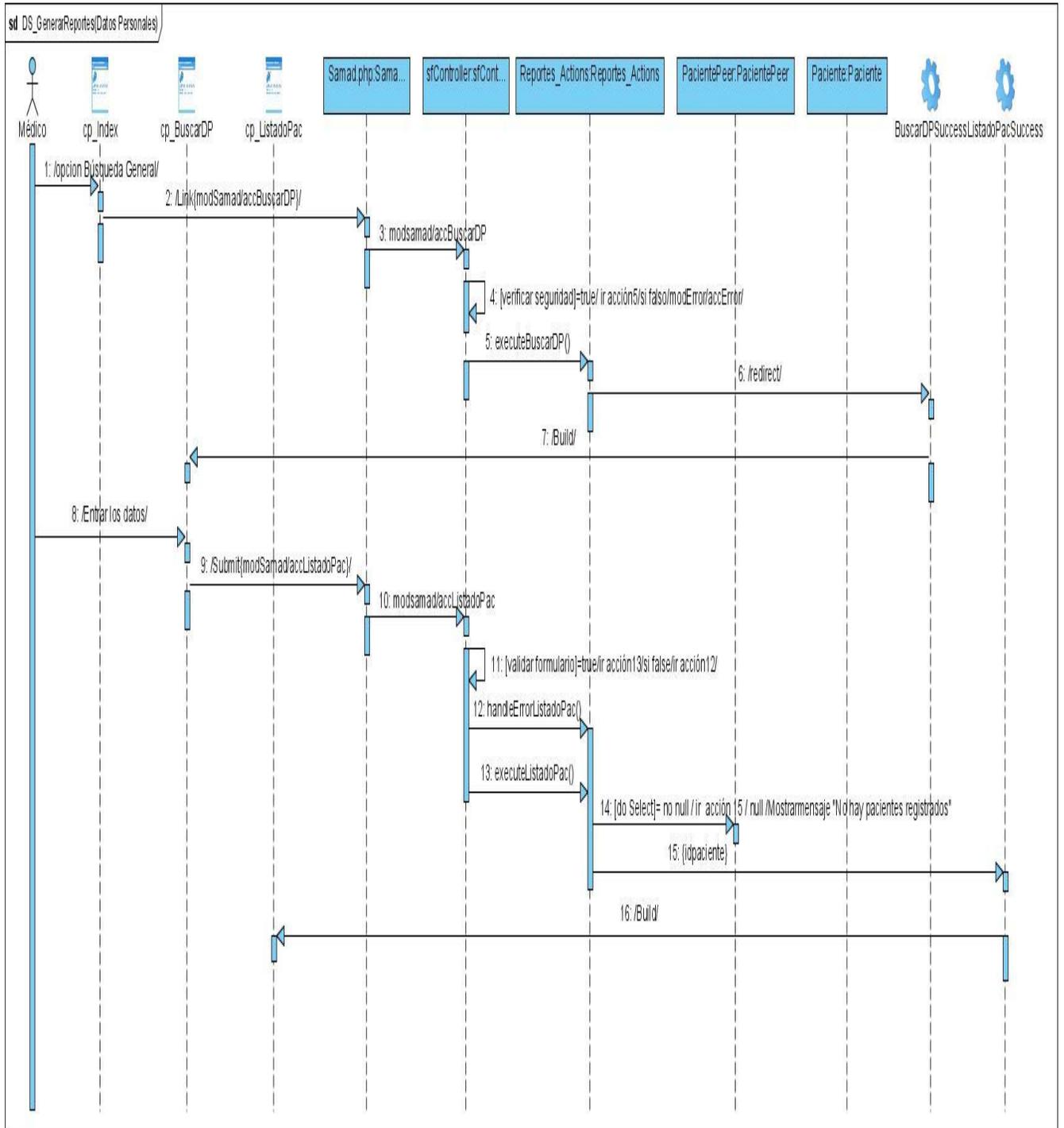


Fig.3.22 Diagrama de Secuencia: Generar Reportes

➤ Diagrama de Secuencia: Autenticar Usuario

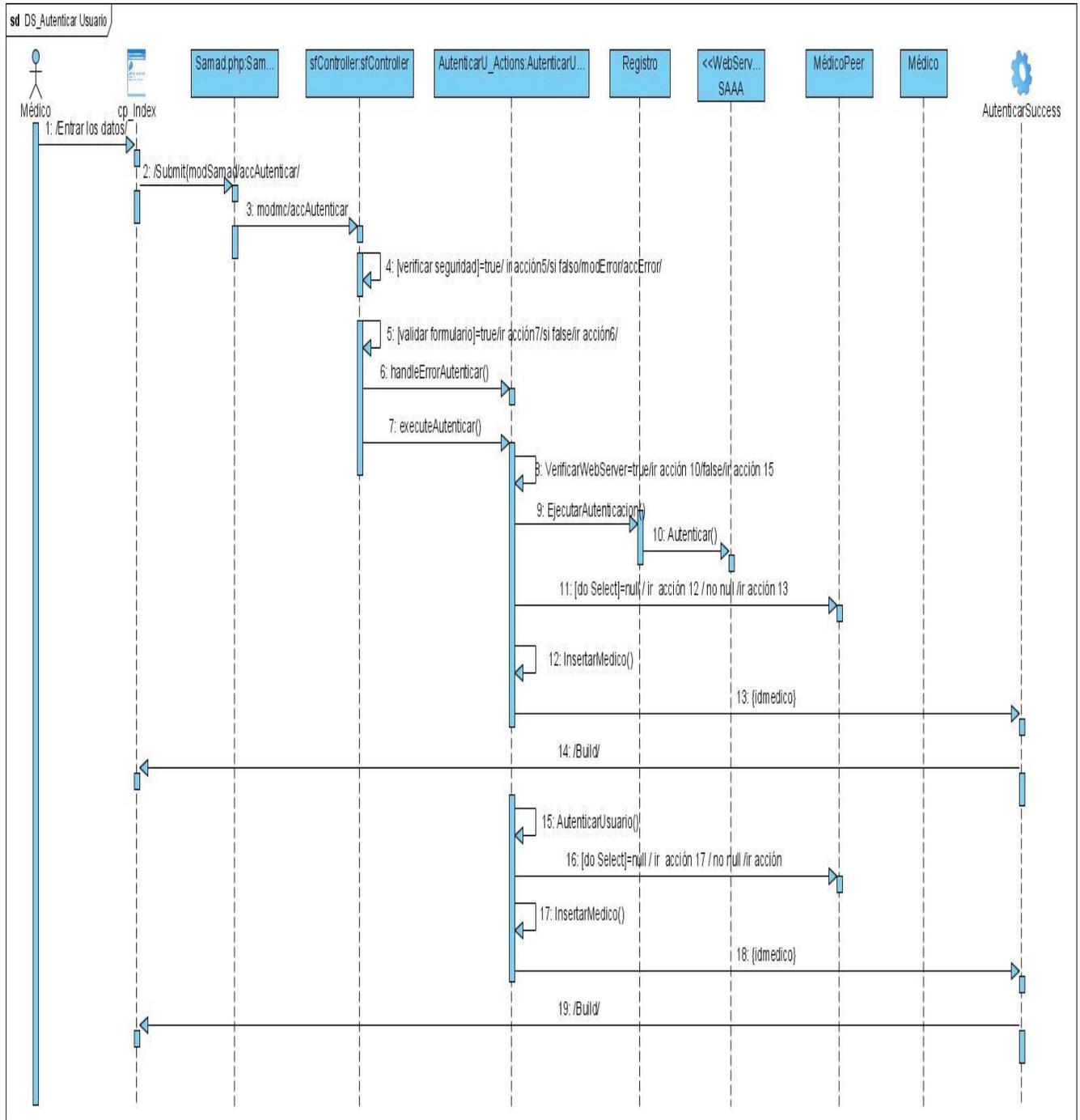


Fig.3.23 Diagrama de Secuencia: Autenticar Usuario

Los diagramas de las clases de secuencia de los CUS tales como Cálculo de Peso Ideal, Cálculo de dietas, Actualización de Fármacos se encuentra en los anexos, el resto en el expediente de proyecto.

3.4 DISEÑO DE LA BASE DE DATOS

En el desarrollo de un sistema informático, el diseño de la BD es de gran importancia, ya que en ella se almacenan todos los datos que son necesarios en la modelación del problema que se desea resolver, además ésta es la fuente de obtención de toda la información que se quiera recuperar del sistema.

Las bases de datos necesitan de una definición de su estructura que le permitan almacenar datos, reconocer el contenido, y recuperar la información. La puesta en práctica de la base de datos es el paso final en el desarrollo de aplicaciones de soporte del negocio. Se conforman con los requisitos del proceso del negocio, que son la primera abstracción de la vista de la base de datos.

La tarea de diseñar la base de datos es antecedida por la realización del diagrama de clases persistentes y le sigue la generación del modelo de datos.

El término clase persistente se refiere a la capacidad de las clases de mantener su valor en el espacio y en el tiempo (persistent) (7).

El diagrama de clases persistente, es un diagrama de clases al igual que otros, posee las relaciones definidas entre las clases (asociación, agregación/composición y generalización/especialización) pero sólo tiene clases que se hayan definido como persistentes. El modelo de datos se refiere a cómo se relacionan las tablas de la futura base de datos.

El diagrama de clases persistentes propuesto para SAMAD es el que se muestra a continuación:

3.4.1 DIAGRAMA DE CLASES PERSISTENTES

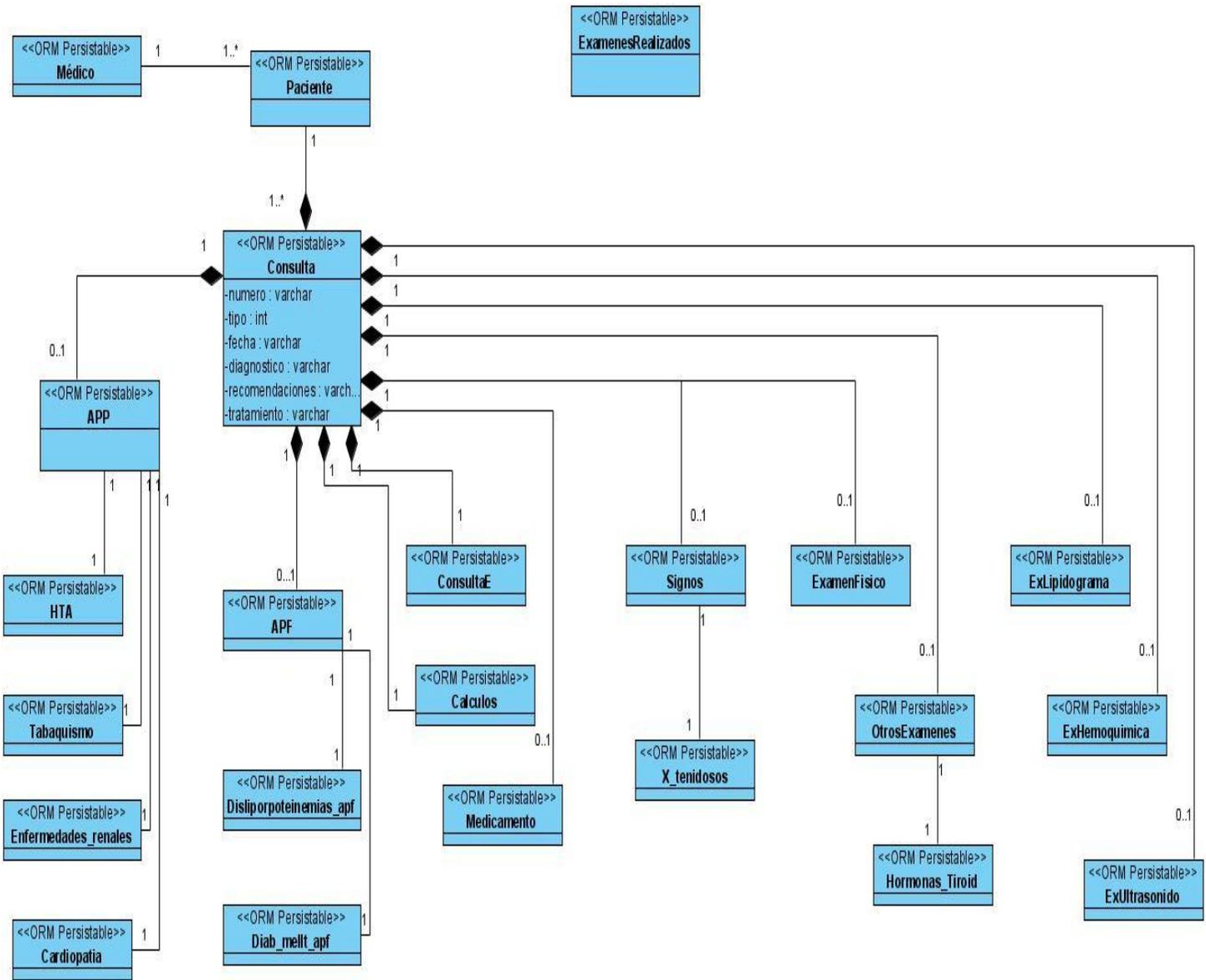


Fig. 3.24 Diagrama de clases persistentes

3.5 MODELO DE DESPLIEGUE

“El Modelo de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo.”

(5)

Características del Modelo de Despliegue:

- Los nodos representan recursos de cómputo, procesador o dispositivo hardware.
- Los nodos poseen relaciones que representan medios de comunicación entre ellos, tales como Internet, Intranet, BUS y otros.
- Representa la correspondencia entre la arquitectura del software y la del hardware.

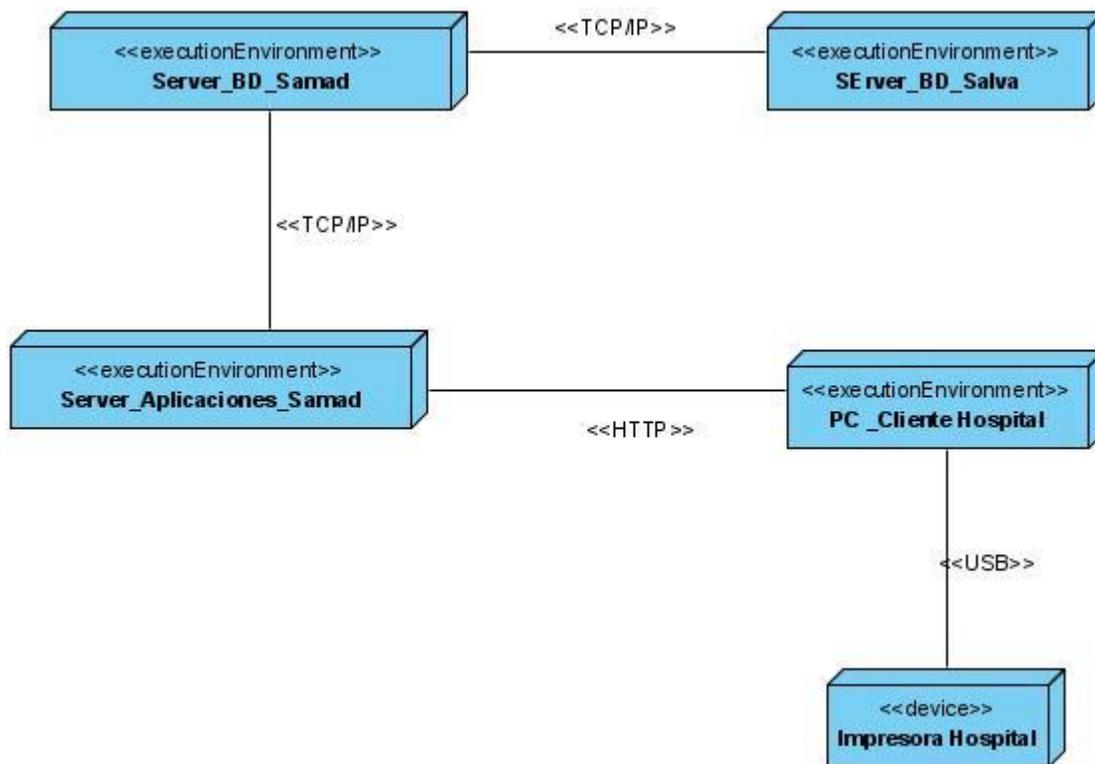


Fig.3.26 Diagrama del Modelo de Despliegue.

CONCLUSIONES

Como resultados de este capítulo, se realizaron los Diagramas de Clases del Análisis y los diagramas de Clases del Diseño de los casos de usos críticos. Se describieron las clases del diseño, además del diagrama de base de datos y el diagrama físico. El resto de los diagramas de clases de análisis, las descripciones de las clases del diseño, así como los diagramas de clases del diseño se encuentran en los **Anexos**.

CONCLUSIONES GENERALES

En la presente investigación se realizó el refinamiento del análisis y diseño del Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias el cual cumple con todos los requerimientos funcionales del usuario. Con dicho resultado se contribuye a la prevención y atención en todo el país de las Dislipoproteinemias. Se diseñaron 32 prototipos de interfaz no funcionales, entre los cuales se encuentran el módulo de cálculos auxiliares, atención de paciente en seguimiento y gestión de pacientes del Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias con el objetivo de automatizar todo el proceso de diagnóstico y tratamiento de esta enfermedad. Se utilizó Symfony como framework permitiendo una mejor integración con la arquitectura establecida por Softel constituyendo una herramienta que en gran medida satisface los requerimientos planteados.

SAMAD ha sido presentado en varios eventos, tal es el caso de la Jornada Científica Estudiantil a nivel de facultad y en la Jornada Científica desarrollado en el Hospital Ernesto Che Guevara de la UCI.

RECOMENDACIONES

La investigación debido a la complejidad del diagnóstico y tratamiento de las Dislipoproteinemias, requiere del estudio de algunas funcionalidades en conjunto con los especialistas del hospital "Hermanos Ameijeiras" para su posterior desarrollo. Por lo que se exhorta lo siguiente:

- Análisis detallado de los algoritmos utilizados para contribuir al diagnóstico y tratamiento de las Dislipoproteinemias que permita al equipo de programadores en conjunto con especialistas informáticos vinculados a la rama de bioinformática un mejor aprovechamiento de las funcionalidades realizadas en el análisis y diseño del sistema propuesto.
- Implementar la aplicación web que fue diseñada en la investigación para que finalmente sea de utilidad en el Sistema Nacional de Salud Cubano, con el objetivo de promover el control y prevención de las Dislipoproteinemias.
- Futuros estudios de gestores de Base de datos compatibles con la arquitectura definida por Softel con la visión de no tener que comenzar desde cero sí se cambia de gestor de base de datos.

REFERENCIAS BIBLIOGRAFICAS

REFERENCIAS BIBLIOGRAFICAS

1. **Licata, Lic. Marcela.** Zonadiet. *Zonadiet*. [Online] Zonadiet.com, 1999-2007. [Cited: Octubre 20, 2007.] <http://www.zonadiet.com/salud/dislipidemia.htm>.
2. **Ing. Alfredo Sánchez Rodríguez, Ing. Frank Pompa Sourd.** *Arquitectura, normas y tecnologías para el desarrollo de aplicaciones informáticas para la Salud Pública en Cuba*. Junio/2007. [Cited: Marzo 5, 2008.].
3. **Jacobson, Ivar, Rumbaugh, James and Booch, Grady.** *El Lenguaje Unificado de Modelado. Manual de Referencia*. s.l. : Addison Wesley, 2000. [Cited: febrero 10, 2008.].
4. *Ayuda extendida del Rational Rose Enterprise Edition*. 2003.
5. **Ivar Jacobson, Rumbaugh, James and Booch, Grady.** *El proceso unificado de desarrollo del software*. s.l. : Addison Wesley, 2000. [Cited: enero 10, 2008.].
6. **Fabien Potencier, François Zaninotto.** *Guia definitiva de Symfony* <http://www.librosweb.es/symfony>.
7. Base de Datos.
[Online]<http://www.telefonica.net/web2/todobi/Pixfolder/Comparativa-Bases-de-Datos-Open-Source-Vs-Comerciales.doc>.

BIBLIOGRAFIA

BIBLIOGRAFIA

1. **Licata, Lic. Marcela.** Zonadiet. *Zonadiet*. [Online] Zonadiet.com, 1999-2007. [Cited: Octubre 20, 2007.] <http://www.zonadiet.com/salud/dislipidemia.htm>.
2. **Médica., Víctor Rolando Avila Cruz. Profesor Asistente de Bioestadística e Informática.** Medicina y Computación: una integración necesaria. *Medicina y Computación: una integración necesaria*. [Online] 1999. [Cited: octubre 20, 2007.] <http://www.compendium.com.ar/neuroc99/text/medicinacomputacion.htm>.
3. **Lic. Bárbara Lázara Hernández González, MSc y .Lic. Elsa Rodríguez Leon, MSc.** Revista Cubana de Informática Médica. *Universidad Médica y sociedad: su vinculación a la luz de la Informática Médica*. [Online] Lic. Bárbara Lázara Hernández González, Lic. Elsa Rodríguez Leon (MSc)., 2005. [Cited: Febrero 19, 2008.] http://www.cecam.sld.cu/pages/rcim/revista_7/articulo_hm/univmesoc.htm#5. ISSN:1684-1859.
4. **Porta, Dr. Carlos Alberto.** Clínica Virtual Ginecológica. *Informática Médica*. [Online] Agosto 20, 1999. [Cited: enero 15, 2008.] <http://members.tripod.com/~gineco/INFOMED.HTM>.
5. **Jacobson, Ivar, Rumbaugh, James and Booch, Grady.** *El Lenguaje Unificado de Modelado. Manual de Referencia*. s.l. : Addison Wesley, 2000. 0-201-30998-X.
6. Base de Datos. [Online] <http://www.telefonica.net/web2/todobi/Pixfolder/Comparativa-Bases-de-Datos-Open-Source-Vs-Comerciales.doc>.
7. *Ayuda extendida del Rational Rose Enterprise Edition*. 2003.
8. **Ivar Jacobson, Rumbaugh, James and Booch, Grady.** *El proceso unificado de desarrollo del software*. s.l. : Addison Wesley, 2000. 0-201-57169-2.
9. **Moreno Martínez, Gerardo.** *Ingeniería de Software UML*, <http://www.monografia.com/trabajos5/insof/insof.htm> [Cited: noviembre 20 , 2007.]
10. **Fabien Potencier, François Zaninotto.** *Guía definitiva de Symfony*. <http://www.librosweb.es/symfony>.
11. **Ing. Aislein Blanco González, Madelayne Muñoz Morejón.** *Trabajo de Diploma: Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias*. Junio 2007. [Cited: Octubre 15 , 2007.]

ANEXOS

ANEXOS.

Anexo_Cap 2 Descripción de los casos de uso del sistema

CU 6: Calcular Peso ideal.

Nombre del Caso de Uso	Calcular Peso Ideal.	
Actores	Usuario(inicia)	
Propósito	Obtener el rango de peso ideal según el índice de masa corporal...	
Resumen	El caso de uso se inicia cuando un usuario sin registrarse en el sistema solicita la opción de calcular el Peso Ideal y concluye cuando el sistema muestra resultados y una serie de recomendaciones.	
Referencias	R4	
Precondiciones	Poder acceder a los servidores de infomed.	
Poscondiciones	Mostrar resultados del cálculo.	
Curso Normal de los Eventos		
Acciones del Actor	Respuesta del sistema.	
1. El usuario selecciona la opción del menú Calcular Peso Ideal.	2. El sistema muestra una interfaz para realizar dicho cálculo.	
3. El usuario introduce los datos necesarios e indica al sistema realizar el cálculo.	4. El sistema valida que los datos sean correctos.	
	5. El sistema realiza el cálculo a partir de los datos insertados.	

ANEXOS

	6. El sistema muestra el resultado del cálculo y una serie de recomendación que puede hacer una persona para mantener una vida más saludable, finalizando así en caso de uso.
Flujo alternativo 4 a“Si los datos no son permitidos”	
Acciones del Actor	Respuesta del Sistema
	4.1 Si los datos introducidos no son permitidos se indican donde están los errores que se cometieron para que sean corregidos. Retorna a 3.
Prioridad:	Normal.

Tabla 2. 10 Descripción del Caso de Uso Calculo de Peso Ideal

CU 7: Calcular Dietas

Nombre del Caso de Uso	Calcular Dietas
Actores	Usuario(inicia)
Propósito	Obtener el cálculo de dietas según calorías.
Resumen	El caso de uso se inicia cuando el usuario solicita la opción de cálculo de dietas según calorías y concluye cuando el sistema muestra resultados y algunas recomendaciones.
Referencias	R5
Precondiciones	Poder acceder a los servidores de infomed.
Poscondiciones	Mostrar resultados del cálculo.

ANEXOS

Curso Normal de los Eventos	
Acciones del Actor	Respuesta del sistema.
1. El usuario selecciona del menú principal la opción Calcular Dietas	2. El sistema muestra una interfaz para realizar dicho cálculo.
2. El usuario introduce los datos necesarios e indica al sistema realizar el cálculo.	4. El sistema valida que los datos sean correctos
	5. El sistema realiza el cálculo a partir de los datos insertados.
	6. El sistema muestra el resultado del cálculo y una serie de recomendación que puede hacer una persona para mantener una vida más saludable, finalizando así en caso de uso.
Flujo alternativo 4 .1“Si los datos no son permitidos”	
Acciones del Actor	Respuesta del Sistema
	4.1 Si los datos introducidos no son permitidos se indican donde están los errores que se cometieron para que sean corregidos. Retorna a 3.
Prioridad:	Normal.

Tabla 2. 11 Descripción del Caso de Uso Calculo de dietas según calorías.

ANEXOS

CU 8: Actualizar fármacos.

Nombre del Caso de Uso	Actualizar fármacos.
Actores	Administrador(inicia)
Propósito	Debido al desarrollo farmacéutico que se ha alcanzado en la actualidad han surgido muchos medicamentos que hace solo unos años no existían. Todo esto nos lleva a tener en presente la existencia de nuevos medicamentos que ayuden al tratamiento de las dislipoproteinemias. El sistema con este módulo no se verá afectado pues sin tener que tocar nada del código ya que permitirá actualizar los fármacos que se indican en el tratamiento.
Resumen	<p>El caso de uso se inicia cuando el administrador pulsa el vínculo para la actualización fármacos y el sistema ofrece la posibilidad de realizar algunas de las siguientes acciones:</p> <ul style="list-style-type: none"> • Insertar Fármacos. Cuando surge un nuevo medicamento para el tratamiento de la dislipoproteinemias. • Eliminar Fármacos. Cuando un medicamento ya existente resulta poco conveniente, ya sea por la existencia de uno nuevo o porque trae reacciones adversas para el paciente. <p>Nota: Esta acción deberá ser consultada y autorizada previamente por las instituciones correspondiente.</p>
Referencias	R15
Precondiciones	Entrar al sitio con los privilegios de administrador
Poscondiciones	Que se actualicen los medicamentos, para que el software no caduque.
Curso Normal de los Eventos	

ANEXOS

Acciones del Actor	Respuesta del sistema.
<p>1. El Administrador selecciona cuál acción desea realizar:</p> <p>d. Insertar Fármacos.</p> <p>e. Eliminar Fármacos.</p>	<p>2. El sistema, en dependencia de la acción solicitada por el Administrador, muestra la interfaz correspondiente:</p> <ul style="list-style-type: none"> • Insertar Fármacos (Ir a la sección "Insertar fármacos"). • Eliminar Fármacos (Ir a la sección "Eliminar Fármacos").
Sección "Insertar Fármacos."	
Acciones del Actor	Respuesta del Sistema
	<p>1. El sistema muestra el formulario seleccionado permitiéndole al Administrador insertar los datos del nuevo medicamento.</p>
<p>2. El Administrador inserta los datos del nuevo medicamento en el formulario y le indica al sistema guardar los datos.</p>	<p>3. El sistema verifica que todos los campos estén llenos.</p>
	<p>4. El sistema verifica que los datos introducidos sean correctos.</p>
	<p>5. El sistema registra el nuevo medicamento en la base de datos.</p>
	<p>6. El sistema emite un mensaje confirmando que los datos han sido guardados correctamente y finaliza así el caso de uso.</p>
Flujo alternativo 3.1 "Si existe algún campo vacío."	
Acciones del Actor	Respuesta del sistema.

	3.1 El sistema emite un mensaje “Todos los campos deben llenarse” y se indica en color rojo, el nombre de los campos vacíos y va a la acción 2 de esta Sección.
Flujo alternativo 4.1 “Si los datos no son correctos.”	
Acciones del Actor	Respuesta del sistema.
	4.1 El sistema emite un mensaje “Los datos deben estar en el rango establecido” y se indica en color rojo, el nombre de los campos incorrectos y va a la acción 2 de esta Sección.

Tabla 2. 12 Descripción del Caso de Uso Actualizar Fármacos.

Las descripciones de los casos de uso Cálculo de Índice de Masa Corporal, Cálculo de Factores de Riesgos, gestionar usuario y Cambios de Rangos se encuentran el expediente de proyectos de SAMAD.

Anexo_Cap 3 DIAGRAMAS DE CLASES DEL ANÁLISIS (DCA).

➤ **Diagrama de Clases del Análisis: Cálculos Auxiliares**

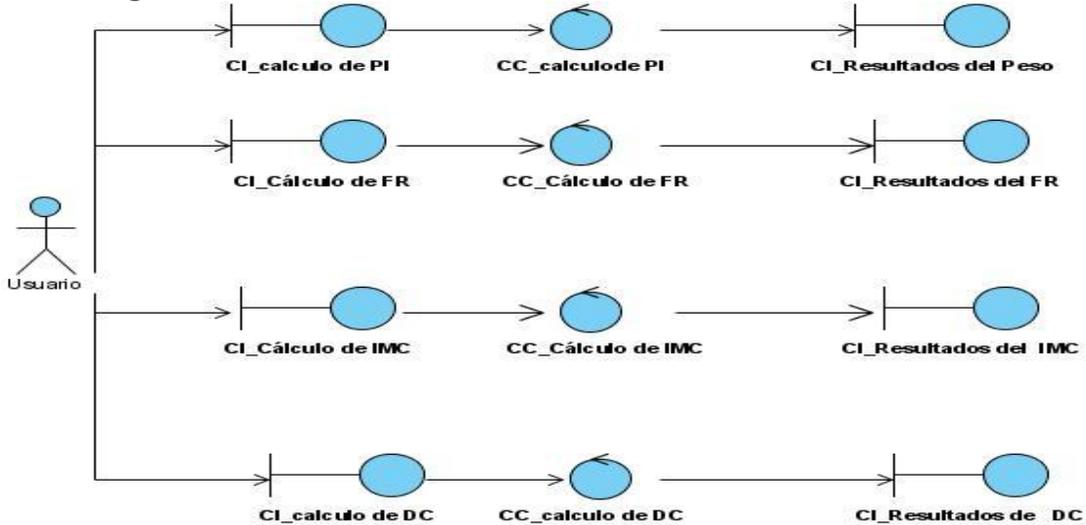


Fig.3.27 Diagrama de Clases del Análisis Cálculos Auxiliares asociado al Usuario.
 ➤ **Diagrama de Clases del Análisis: Actualizar Fármacos**



Fig. 3.28 Diagrama de Clases del Análisis Actualizar Fármacos asociado al Administrador

Los diagramas de las clases de análisis de los casos de uso Cálculo de Índice de Masa Corporal, Cálculo de Factores de Riesgos, gestionar usuario y Cambios de Rangos se encuentran en el expediente de proyectos de SAMAD.

DIAGRAMAS DE CLASES DEL DISEÑO (DCD).

➤ **Diagrama de Clases del Diseño: Calcular Dietas**

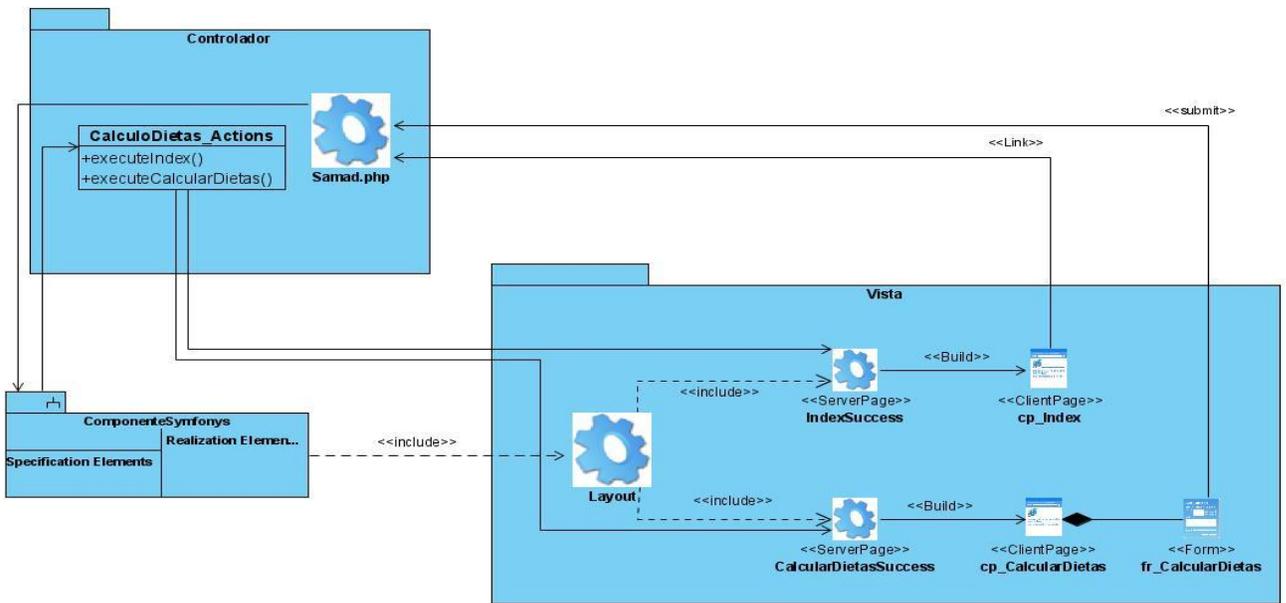


Fig.3.29 Diagrama de Clases del Diseño: Calcular Dietas

Diagrama de Clases del Diseño: Actualizar Fármacos

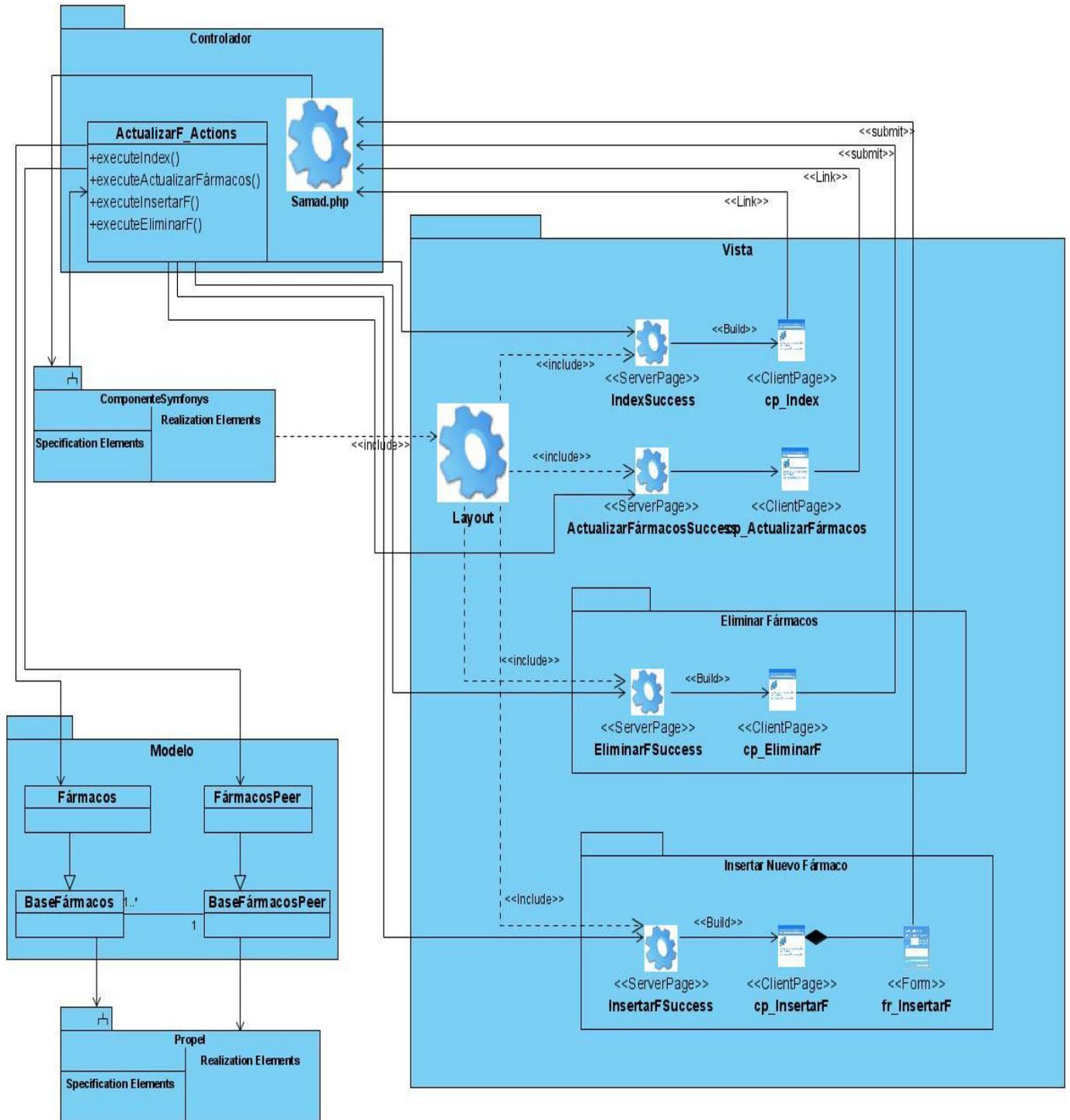


Fig.3.30 Diagrama de Clases del Diseño: Actualizar Fármacos

➤ Diagrama de Clases del Diseño: Calcular Peso Ideal

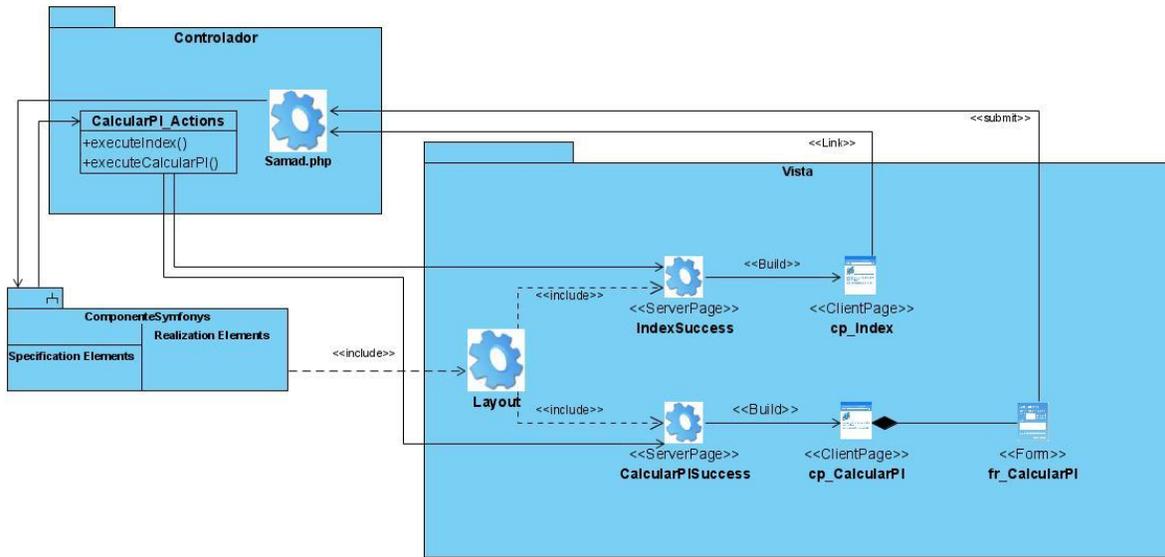


Fig.3.31 Diagrama de Clases del Diseño: Calcular Peso Ideal

Los diagramas de las clases de diseño de los casos de uso Cálculo de Índice de Masa Corporal, Cálculo de Factores de Riesgos, gestionar usuario y Cambios de Rangos se encuentran en el expediente de proyectos de SAMAD.

DESCRIPCIÓN DE CLASES DEL DISEÑO

Calcular Dietas

Nombre: CalcularDietas_Actions
Tipo: Clase (Esta clase contiene todas las funciones correspondientes al paquete de diseño Calcular Dietas según calorías. Este cálculo no se va a guardar en la base de datos del sistema, por lo cual no tiene interacción con el modelo ..)
Atributo-Tipo Peso-varchar
Responsabilidades:

ANEXOS

Nombre:	Descripción:
executeCalcularDietas()	Permite calcular la dieta de usuario. En la función CalcularDietas() se calcula la dieta entrando como datos el peso del usuario.

Tabla 3.5 CUS: Calcular Dietas

Actualizar Fármacos

Nombre: ActualizarF_Actions	
Tipo: Clase (Esta clase contiene todas las funciones correspondientes para la actualización de los fármacos, los cuales van a ser utilizar para emitir un tratamiento. Todas las funciones van a utilizar los métodos relacionados con el paquete Modelo).	
Atributo-Tipo Fechaentrad-varchar, nombre-varchar, familiamed-varchar	
Responsabilidades:	
Nombre:	Descripción:
executeActualizarFármacos()	Muestra un menú a partir del cual podrá insertar o eliminar fármacos en el sistema.
executeInsertarF()	Muestra un formulario permitiendo llenar todos los datos de un nuevo fármaco. Para la implementación de este método se van a utilizar algunos métodos de la clase Peer tales como: DoSelect, construct, Set, Save. Los cuales van a permitir la creación e inserción del nuevo fármaco en la Base de Datos.

ANEXOS

executeEliminarF()	Esta función va a buscar y mostrar todos los medicamentos existentes hasta el momento y a través de un llamado a la función EliminarFármacos() se obtiene el fármaco seleccionado y se elimina de la base de datos. Para la implementación de esta función se van a utilizar métodos de las clases Peer tales como: DoSelect, Criterias, Save, DoDelete.
--------------------	--

Tabla 3.6 CUS: Actualizar Fármacos

Calcular Peso Ideal (PI)

Nombre: CalcularPI_Actions	
Tipo: Clase (Esta clase contiene todas las funciones correspondientes al paquete del diseño Calcular PI. Este cálculo no se va a guardar en la base de datos del sistema, por lo cual no tiene interacción con el modelo).	
Atributo-Tipo Talla-varchar	
Responsabilidades:	
Nombre:	Descripción:
executeCalcularPI()	Permite calcular el peso ideal de un usuario. En la función CalcularPI() se calcula el peso entrando como dato la talla del usuario.

Tabla 3.7 CUS: Calcular Peso ideal

Las descripciones de las clases de diseño de los casos de uso Cálculo de Índice de Masa Corporal, Cálculo de Factores de Riesgos, gestionar usuario y Cambios de Rangos se encuentran en el expediente de proyectos de SAMAD.

➤ Diagrama de Secuencia: Actualizar Fármacos {Insertar Fármacos}

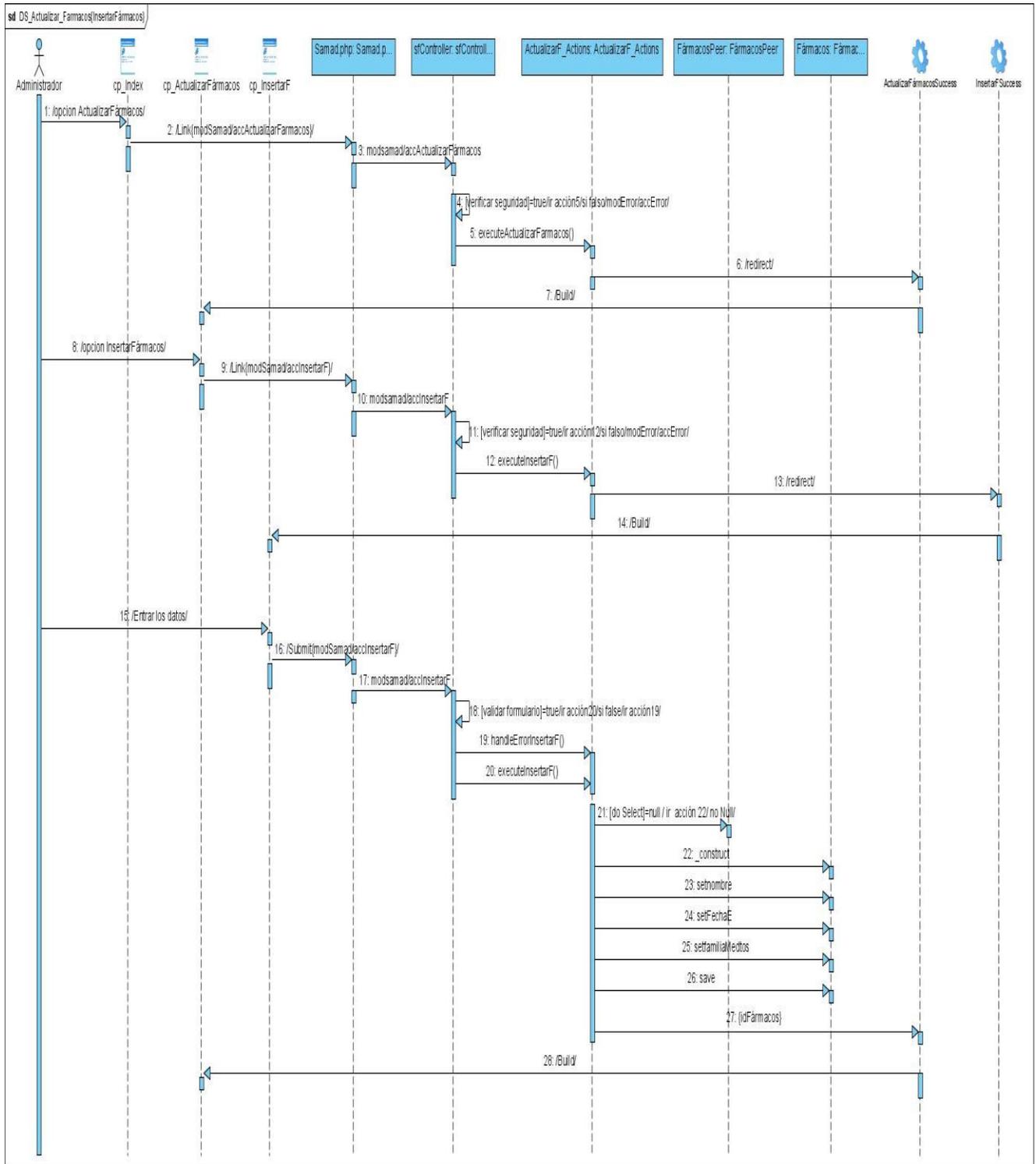


Fig.3.32 Diagrama de Secuencia: Actualizar Fármacos {Insertar Fármacos}

➤ Diagrama de Secuencia: Calcular Dietas

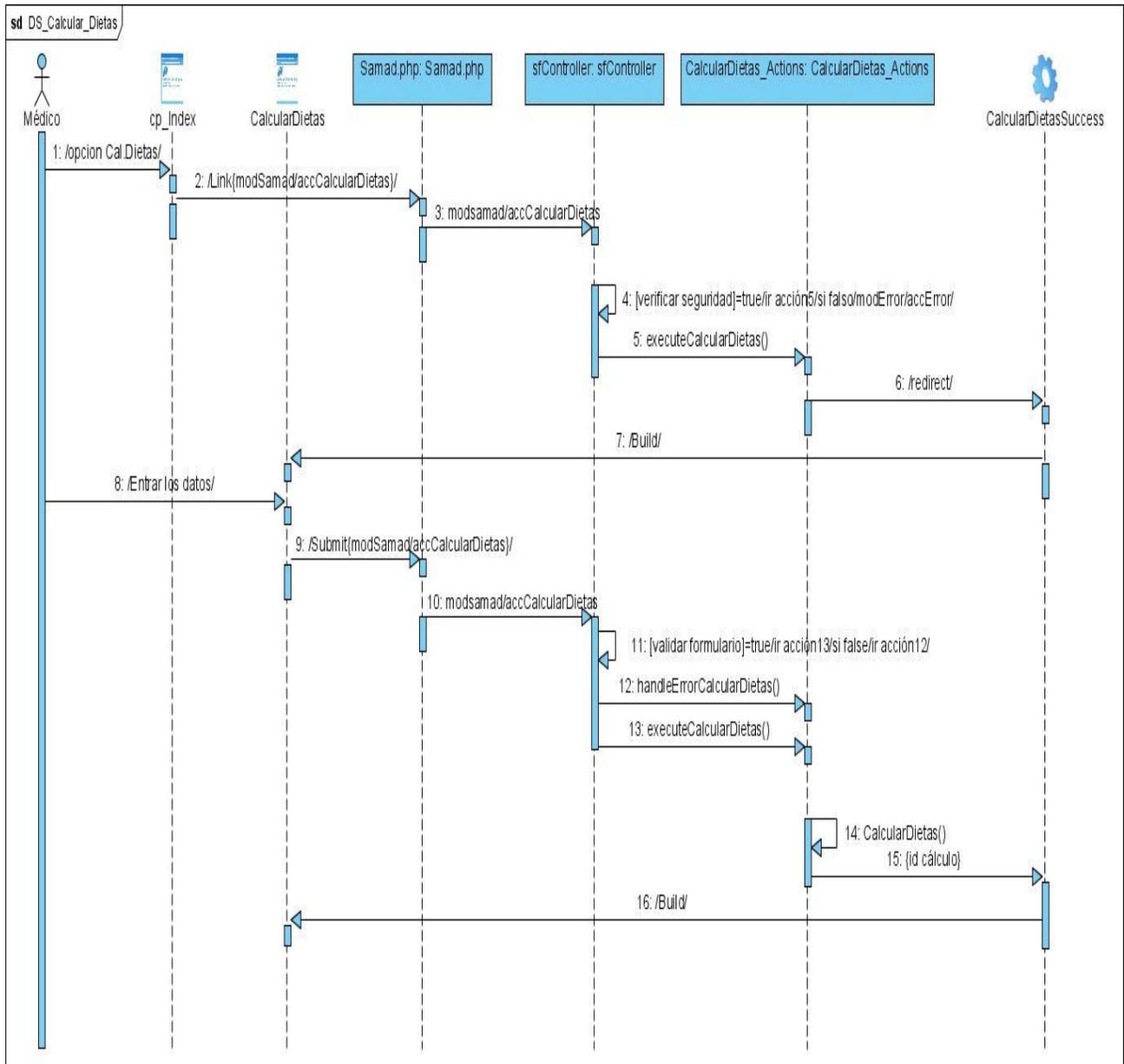


Fig.3.33 Diagrama de Secuencia: Calcular Dietas

➤ Diagrama de Secuencia: Calcular Peso Ideal

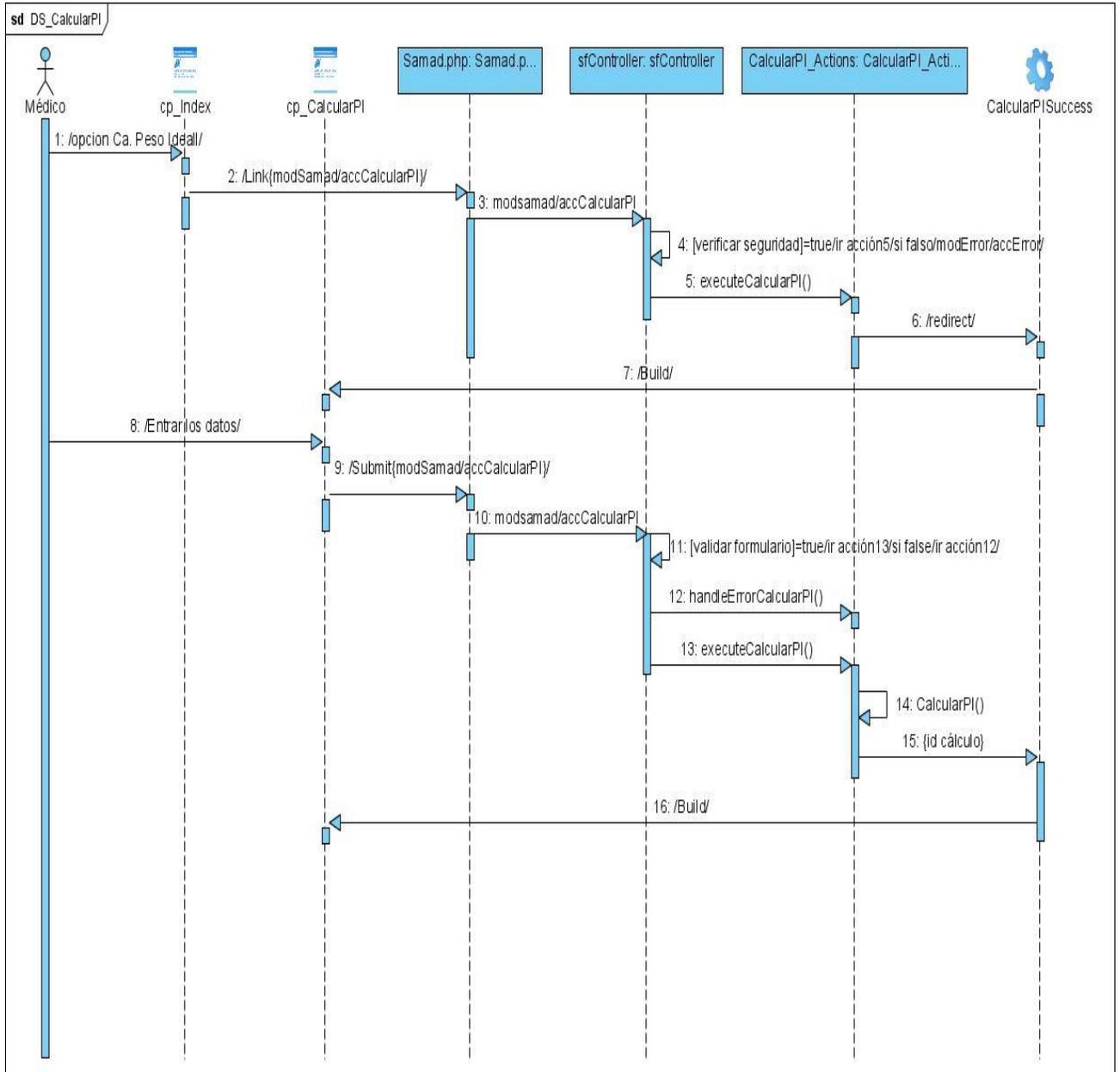


Fig.3.34 Diagrama de Secuencia: Calcular Peso Ideal

Las diagramas de secuencia de los casos de uso Cálculo de Índice de Masa Corporal, Cálculo de Factores de Riesgos, gestionar usuario y Cambios de Rangos se encuentran en el expediente de proyectos de SAMAD.

PROTOTIPO DE INTERFAZ NO FUNCIONAL

SISTEMA DE AYUDA MÉDICA PARA LA ATENCIÓN DE LAS DISLIPOPROTEINEMIAS.

SAMAD
Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias

Bienvenido a la Página Inicial

Hacer dieta pensando en tu salud

Cinco normas a seguir.

1. Distribuir la comida del día en : desayuno, media mañana, comida, merienda y cena; siendo la comida y la cena ligeramente superiores al resto en cantidad, y el desayuno más abundante de lo habitual.
2. Comer de todo, salvo frituras y salsas. Lo mejor es cocer la carne, el pescado y las verduras. También se pueden asar o hacer a la plancha. Igualmente se aconseja utilizar leche desnatada en lugar de la entera.

Hipertensión Arterial

SAMAD (CON ACCESO AUTORIZADO)

SAMAD
Sistema de Ayuda Médica para la Atención de las Dislipoproteinemias

Bienvenido a la Página Inicial

Hacer dieta pensando en tu salud

Cinco normas a seguir.

1. Distribuir la comida del día en : desayuno, media mañana, comida, merienda y cena; siendo la comida y la cena ligeramente superiores al resto en cantidad, y el desayuno más abundante de lo habitual.
2. Comer de todo, salvo frituras y salsas. Lo mejor es cocer la carne, el pescado y las verduras. También se pueden asar o hacer a la plancha. Igualmente se aconseja utilizar leche desnatada en lugar de la entera.

Hipertensión Arterial

Menu principal

- Inicio
- Cal. Factores de Riesgos
- Cal. Peso Ideal
- Cal. Ind de Masa Corporal
- Cal. Dietas
- Ayuda
- Foro

Acceso de usuarios

Hola, admin

Salir

ANEXOS

BUSCAR PACIENTE

 **MÉDICO :** Alexander Rodriguez

 Nuevo Paciente
  Consultas
  Modificar y Completar
  Mostrar HC

Consulta del Dr/Dra Alexander Rodriguez



Buscar paciente

Nombre

Primer Apellido:

Segundo Apellido:

Carnet de Identidad:

Buscar

No.	Nombre y Apellidos	Hist. Clínica	Consulta	Modificar
1	SAIDEL SAADS ADSASD	20071124111111		
2	SAIDEL WWW QQ	20080424111111		

Completado No completado

GESTIONAR PACIENTE

- Menu principal**
- Inicio
 - Cal.Factores de Riesgos
 - Peso Ideal
 - Cal. Dietas
 - Cal. Ind de Masa Corporal
 - Ayuda
 - Foro
 -
 - SAMAD**
 -
 - Buscar Paciente
 - Búsqueda General
 -
 - ADMINISTRAR**
 -
 - Actualizar Fármacos
 - Administración
 - Rangos y Valores
- Acceso de usuarios**
- Hola, admin
 - Salir
- Usuarios en línea**
- admin

 **PACIENTE :** Ninguno

 **MÉDICO :** Alexander Rodriguez

 Nuevo Paciente
  Consulta
  Modificar y Completar
  Mostrar HC

Consulta del Dr/Dra Alexander Rodrig

- Datos personales**
- APP
- APF
- Exámen general**
- Exámen físico
- Signos asociados
- Medicamentos tomados**
- Exámenes de Lípidos y Lipoproteinémias
- Otros Exámenes
- Diagnóstico
- Cálculos Auxiliares

Datos personales

Nombre

Apellido1

Apellido2

Pais

Provincia

Municipio Localidad

Calle Entre

No. Apto

Fecha Nac. Año Mes Dia

Sexo

Telefono:

ANEXOS

GESTIONAR CONSULTA EVOLUTIVA

Consulta del Dr/Dra {DOCTOR}

- Insertar Consulta
- Rectificar Consulta
- Cálculos Auxiliares

Consulta Evolutiva

Fecha:

Peso:

Presión Arterial:

Colesterol:

Trigliceridos:

HDL:

LDL:

VLDL:

Otros:

Diagnóstico del Paciente:

Tratamiento del Paciente:

Recomendaciones:

Cálculos AUXILIARES

Consulta del Dr/Dra {DOCTOR}

- Insertar Consulta
- Cálculos Auxiliares

Calcular Factor de Riesgo Cardiovascular

Edad:

Colesterol: 100 y 800 mg/dL

Sexo:

Es Fumador?

cHDL: 10 y 100 mg/dL

BP:

TG: 10 y 500 mg/dL

Cumple tratamiento para la hipertensión?

Resultado:

Calcular Indice de Masa Corporal

Talla:

Peso:

Resultado:

Calcular Peso Ideal

Talla:

Resultado:

Calcular Dietas Según Calorias

Peso Ideal:

Resultado:

CALCULAR DIETAS

Inicio ► Cal. Dietas 09-06-2008

Menu principal

- Inicio
- Cal. Factores de Riesgos
- Cal. Peso Ideal
- Cal. Dietas
- Cal. Ind de Masa Corporal
- Ayuda
- Foro
-
- SAMAD**
-
- Buscar Paciente
- Búsqueda General
-
- ADMINISTRAR**
-
- Actualizar Fármacos
- Administración
- Busqueda Médico
- Rangos y Valores

Acceso de usuarios

Hola, admin

Salir

Calcule su Dieta según sus calorías

Peso Ideal

LDL Cholesterol (red arrow down 31%)

HDL Cholesterol (green arrow up 29%)

CALCULAR PESO IDEAL

Calcule su peso ideal.

Talla

GENERAR REPORTES

SAMAD

Inicio ► Búsqueda General 09

Menu principal

- Inicio
- Cal. Factores de Riesgos
- Peso Ideal
- Cal. Dietas
- Cal. Ind de Masa Corporal
- Ayuda
- Foro
-
- SAMAD**
-
- Buscar Paciente
- Búsqueda General
-
- ADMINISTRAR**
-
- Actualizar Fármacos
- Administración
- Rangos y Valores

Acceso de usuarios

Hola, admin

Salir

Datos Personales
APP
Otros

Nombre

Apellido1

Apellido2

Pais

Provincia

Municipio

Sexo

ACTUALIZAR FÁRMACOS

Consulta del Dr./Dra. {DOCTOR}

Insertar Fármacos
 Eliminar Fármacos

Insertar Fármacos

Fecha de Entrada:

Nombre:

Familia:



GLOSARIO DE TÉRMINOS

1. **Dislipoproteinemias:** Las dislipoproteinemias son trastornos que pueden ser ocasionados por defectos en el transporte de los lípidos. Dermatológicamente se pueden manifestar como xantomas que son característicos para el dermatólogo quien es el que generalmente hace el diagnóstico. La oportunidad del diagnóstico y su tratamiento puede evitar complicaciones.
2. **Lípidos:** son un conjunto de moléculas orgánicas, la mayoría biomoléculas, compuestas principalmente por carbono e hidrógeno y en menor medida oxígeno, aunque también pueden contener fósforo, azufre y nitrógeno, que tienen como característica principal el ser hidrofóbicas o insolubles en agua y sí en disolventes orgánicos como el benceno.
3. **Aplicación Web:** es un sistema informático que los usuarios utilizan accediendo a un servidor web a través de Internet o de una Intranet
4. **Base de datos:** Una base de datos es un conjunto de datos que pertenecen al mismo contexto almacenado sistemáticamente para su uso posterior. En este sentido, una biblioteca puede considerarse una base de datos compuesta en su mayoría por documentos y textos impresos en papel e indexados para su consulta.
5. **Factores de Riesgo:** Características de una persona que aumentan sus probabilidades de desarrollar una enfermedad.
6. **Colesterol:** una sustancia similar a la grasa que se encuentra en la sangre, los músculos, el hígado, el cerebro y en otros tejidos del cuerpo.
7. **Triglicéridos:** Es grasa. Está formado por tres moléculas de ácido graso y una molécula de glicerol. Parte de todos los lípidos de la sangre, especialmente los de tipo baja densidad, que son dañinos para el cuerpo.
8. **Lipoproteínas:** Las lipoproteínas son conjugados de proteínas con lípidos, especializadas en el transporte de estos últimos y se dividen en varios grupos según su densidad
9. **Lipoproteínas de baja densidad (LDL):** son habitualmente de unas dos terceras partes del colesterol plasmático total.

10. **Lipoproteínas de alta densidad (HDL):** Lipoproteínas de alta densidad. Estas se conocen como las protectoras. Ya que no permiten que las otras lipoproteínas que son las agresoras se peguen a las células y nos provoque daños en nuestro cuerpo.
11. **Hipercolesterolemia:** Aumento del colesterol en sangre, generalmente se refiere al LDL (malo).
12. **Hipertrigliceridemia:** La hipertrigliceridemia se caracteriza por el aumento de los triglicéridos plasmáticos por encima de 200 miligramos por cada decilitro de sangre.
13. **Aterosclerosis:** La aterosclerosis es un síndrome caracterizado por el depósito de sustancias lipídicas, llamado placa de ateroma, en las paredes de las arterias de mediano y grueso calibre.
14. **Metodologías:** Se refiere a los métodos de investigación en una ciencia. Aun cuando el término puede ser aplicado a las artes cuando es necesario efectuar una observación o análisis más riguroso o explicar una forma de interpretar la obra de arte.
15. **Algoritmos:** Un algoritmo es un conjunto de operaciones y normas que deben seguirse para resolver un problema. Los algoritmos se codifican en sistemas comprensibles para los motores de búsqueda. El lenguaje algorítmico, es por tanto, más cercano a ser entendido por una máquina que por las personas.
16. **Software:** también conocido como programático y aplicación informática- es la parte lógica del ordenador, esto es, el conjunto de programas que puede ejecutar el hardware para la realización de las tareas de computación a las que se destina. Es el conjunto de instrucciones que permite la utilización del equipo.
17. **Refinamiento:** Se refiere a tener que realizar un nuevo análisis detallado a la aplicación que ya fue desarrollada pero que todavía tiene problemas en cuanto a su uso.