



**UNIVERSIDAD DE LAS CIENCIAS INFORMÁTICAS**

**FACULTAD 7**

**Trabajo de Diploma para Optar por el Título de  
Ingeniero en Ciencias Informáticas**

**Título: Sistema Nacional de Rehabilitación**

**Autores:** Yalinet Padilla Lee

Alejandro Martínez Brito

**Tutor:** Ing. Daniel Miranda Calzadilla

**Ciudad de la Habana, Mayo 2009**

**“Año del 50 aniversario del triunfo de la Revolución”**



*“Todos y cada uno de nosotros paga puntualmente su cuota de sacrificio consciente de recibir el premio de la satisfacción del deber cumplido, conscientes de avanzar con todos hacia el Hombre Nuevo que se vislumbra en el horizonte.”*

*Ernesto Guevara de la Serna*

## **DECLARACIÓN DE AUTORÍA**

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 firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

**Yalinet Padilla Lee**

\_\_\_\_\_  
Firma del Autor

**Alejandro Martínez Brito**

\_\_\_\_\_  
Firma del Autor

**Daniel Miranda Calzadilla**

\_\_\_\_\_  
Firma del Tutor

## AGRADECIMIENTOS

*De Yalinet:*

*A mis padres por estar siempre guiándome para que siga el mejor camino.*

*A mi hermana y a mi padrastro por sus consejos y su apoyo.*

*A mi novio Eliurkís, gracias por estar siempre a mi lado.*

*A todos mis compañeros por compartir conmigo estos cinco años de universidad.*

*A mis amigos Yanísel, Javier, Solainy, Juenlís, Darían, por estos años inolvidables. ¡Nunca me olvidaré de ustedes!*

*A mi tutor por confiar en mí y aguantar mis nerviosismos en todo este tiempo.*

*A toda mi familia por apoyarme en esta dura etapa de estudio.*

*A mis compañeros del proyecto gracias por su apoyo.*

*A todas las personas que contribuyeron para que este sueño se cumpla.*

*¡Muchas Gracias!*

*De Alejandro:*

*A aquellos amigos que comenzaron conmigo, continuaron junto a mí y hoy nos graduamos juntos. Nunca los olvidaré.*

*A mis compañeros de aula por compartir conmigo y soportarme durante estos maravillosos cinco años. Sé que no fue tarea fácil.*

*A todos los que pusieron su granito de arena y contribuyeron a que yo cumpla mi sueño.*

*A mis compañeros del proyecto por su apoyo y preocupación.*

*A todos ¡Muchas Gracias!*

## DEDICATORIA

### *De Yalinet:*

*A nuestro Comandante Fidel Castro por ser guía de esta revolución que ha hecho realidad este sueño tan bonito.*

*A mis padres María Antonia y Abel por ser guías y ejemplo en todo momento.*

*A mi padrastro Jesús por ser tan bueno conmigo y por su gran apoyo.*

*A Yalíen (yayí) por ser no sólo hermana sino amiga.*

*A mi novio Eliurkís que con su amor convirtió los momentos tristes en momentos felices.*

*A mi compañero de tesis Alejandro por su confianza y apoyo.*

*A mis tíos Gregorio y María por ser mi apoyo estando tan lejos de mi hogar.*

### *De Alejandro:*

*A mamá ¡VIEJA TE EXTRAÑO!*

*A papí, por tener siempre el consejo exacto en el momento más oportuno.*

*A Tata, por ser madre, hermana y amiga.*

*A Yisel, por estar siempre a mi lado brindándome su apoyo incondicional.*

*Al equipo 5.*

*A todos los que anduvieron junto a mí en este largo maratón de cinco años.*

*A mi compañera de tesis, Yalinet, por su apoyo y por soportarme durante todo este año.*

*A la Revolución y al Comandante en Jefe por hacer posible que se materialice mi mayor sueño*

### RESUMEN

El Ministerio de Salud Pública y la Universidad de las Ciencias Informáticas se han propuesto informatizar los centros hospitalarios del país, obteniendo en el año 2008 una aplicación web para las salas de rehabilitación. Esta no satisface las necesidades actuales del cliente ya que la información no fluye de igual forma en todos los niveles del Sistema Nacional de Salud.

El presente trabajo tiene como objetivo desarrollar una nueva versión del “Sistema Automatizado para la Gestión de la Información en Rehabilitación”, que garantice un manejo correcto del flujo de información en todos los niveles del Sistema Nacional de Salud en el área de rehabilitación.

Para el desarrollo de la aplicación se emplean: el lenguaje de programación PHP5, el Framework CodeIgniter que utiliza la arquitectura Modelo-Vista-Controlador (MVC), el Sistema Gestor de Bases de Datos PostgreSQL y la metodología de Desarrollo el Proceso Unificado de Desarrollo (RUP). Además; para especificar, construir y documentar el sistema se hace uso del Lenguaje Unificado de Modelado (UML) y Visual Paradigm for UML 2.0 para la creación de los artefactos que se generan durante el ciclo de vida del software.

Esta nueva versión de la aplicación web “Sistema Automatizado para la Gestión de la Información en Rehabilitación” permitirá al país contar con un sistema que facilite la conexión de las diferentes clínicas de rehabilitación. Así como el acceso a la información desde una determinada área (municipio, provincia, nación y la utilización de un estándar de clasificación de enfermedades).

**Palabras Claves:** Rehabilitación, CodeIgniter, Framework, Software.

# TABLA DE CONTENIDOS

<b>INTRODUCCIÓN</b> .....	<b>1</b>
<b>Capítulo 1. Fundamentación Teórica</b> .....	<b>1</b>
1.1. Orígenes y Evolución de la Fisioterapia .....	1
1.2. Rehabilitación Integral a Nivel Internacional .....	2
1.3. Rehabilitación Integral en Cuba .....	3
1.4. Sistemas Automatizados Existentes .....	4
1.5. Tecnologías .....	6
1.6. Arquitectura .....	15
1.7. Metodología .....	15
1.8. Herramientas .....	16
<b>Capítulo 2. Características del Sistema</b> .....	<b>19</b>
2.1 Descripción del flujo de trabajo en las Salas de Rehabilitación .....	19
2.2 Descripción de los Procesos del Negocio .....	19
2.3 Análisis crítico de ejecución de los procesos actuales .....	20
2.4 Reglas del Negocio .....	21
2.5 Objeto de Automatización .....	21
2.6 Propuesta de Modelo de Negocio .....	22
2.6.1 Actores del Negocio .....	22
2.6.2 Trabajadores del Negocio .....	23
2.6.3 Diagrama de Casos de Uso del Negocio .....	24
2.6.4 Especificación de los Casos de Uso .....	25
2.6.5 Diagrama de Actividades .....	26
2.7 Especificación de los requisitos de software .....	29
2.7.1 Requerimientos Funcionales. ....	30
2.7.2 Requerimientos No Funcionales. ....	30
2.8 Definición de los casos de uso. ....	32
2.8.1 Definición de los actores. ....	32
2.8.2 Listado de Casos de Usos (CU) .....	34
2.8.3 Diagrama de casos de uso del sistema. ....	38

2.8.4 Casos de uso por ciclo. ....	38
2.8.5 Casos de uso expandidos. ....	40
<b>Capítulo 3. Análisis y Diseño .....</b>	<b>58</b>
3.1 Arquitectura.....	58
3.2 Modelo de Análisis.....	59
3.3 Modelo de Diseño.....	65
<b>Capítulo 4. Implementación .....</b>	<b>79</b>
4.1 Modelo de Datos.....	79
4.2 Modelo Implementación.....	88
4.3 Diagrama de Despliegue .....	88
4.4 Diagrama de Componentes .....	89
4.5 Tratamiento de Errores .....	92
4.6 Seguridad.....	93
4.7 Estrategias de Codificación. Estándares y estilos a utilizar. ....	94
<b>CONCLUSIONES .....</b>	<b>101</b>
<b>RECOMENDACIONES.....</b>	<b>102</b>
<b>REFERENCIAS BIBLIOGRÁFICAS .....</b>	<b>103</b>
<b>BIBLIOGRAFÍA.....</b>	<b>105</b>
<b>Anexo 1 Diagramas de Secuencia.....</b>	<b>107</b>
<b>Anexo 2 Diagrama de Clases del Diseño.....</b>	<b>115</b>
<b>GLOSARIO DE TÉRMINOS.....</b>	<b>121</b>



# INTRODUCCIÓN

En la actualidad, la Informática ha tomado gran auge a nivel mundial y como resultado de esto, cada vez se crean sistemas informatizados más fiables, oportunos, de gran calidad. Estos resultan asequibles para promover la capacitación, la enseñanza y la investigación continua en la medicina, así como mejorar la calidad de la atención al paciente. Los sistemas orientados a la salud varían sus características debido a la existencia de importantes diferencias entre una región y otra, de un país a otro.

Las primeras aplicaciones desarrolladas en el campo de la salud fueron dirigidas a resolver tareas orientadas a ciertos servicios o departamentos, usándose sobre todo con fines administrativos, análisis financieros o manejo de bases de datos. Estas aplicaciones han tenido un gran impacto desde el punto de vista organizativo, pero no están dirigidas a las tareas propiamente médicas, aunque han contribuido a reducir el tiempo de acceso a la información.

Una de las principales metas del gobierno cubano es desarrollar la Industria del Software en el sector de la salud, aprovechando las ventajas que aportan el uso de las Tecnologías de la Información y las Comunicaciones (TIC) e informatizar los servicios que brindan los centros hospitalarios del país, para así mejorar los procesos de gestión de información y ofrecer una atención cualitativamente superior a la población.

A finales del 2002 se crea la Universidad de las Ciencias Informáticas, esta marca el comienzo de una nueva etapa en el desarrollo de la industria cubana del software. La misma está conformada por diferentes facultades que se identifican con perfiles correspondientes con diversos sectores de la sociedad. La facultad 7 se dedica al desarrollo de software para la salud, para ello tiene en cuenta un conjunto de métodos, técnicas, procedimientos y actividades gerenciales dirigidas al manejo de la información en esta área.

La complejidad del sistema de salud cubano está dada por sus características de universalidad, cobertura, gratuidad y accesibilidad. Estas permiten que un ciudadano se atienda por diferentes instituciones en función de su salud, ya sea en un Consultorio Médico, Policlínico, Hospital, Clínica Estomatológica, Farmacia, Clínica de Rehabilitación u otro tipo de institución que genere información para la toma de decisión clínica o para el proceso de dirección en los diferentes niveles.

Para que la Atención Primaria cumpla un papel fundamental en la elevación del estado de salud de la población, no solo es necesario desarrollar a ese nivel acciones de prevención y curación. Sino que es indispensable lograr acciones de rehabilitación concretas que, en última instancia, determinan el grado de incorporación o reincorporación del individuo a la vida social, que con una expectativa de vida de 78 años es hoy el más alto de Latinoamérica.

La rehabilitación es un proceso global y continuo de duración limitada, con objetivos definidos, encaminados a lograr niveles de independencia física y las habilidades funcionales de las personas con discapacidades, así como su ajuste psicológico, social y vocacional que le permitan llevar de forma libre e independiente su propia vida. [1]

La necesidad de estos servicios de rehabilitación ha conllevado a la apertura de clínicas a lo largo de todo el país. La primera sala de rehabilitación en 1964 se inauguró en Santiago de Cuba y actualmente estas se perfeccionan con nuevos servicios en todas las provincias. Además, se trabaja para elevar la formación del personal calificado que asume el trabajo asistencial en estos centros con el deber de atender de forma eficiente al paciente, lo que depende en gran medida de la buena gestión de la información en dichos centros.

Debido a la gran demanda de pacientes con necesidades de rehabilitación física y el gran cúmulo de documentos que deben ser llenados manualmente, las clínicas de rehabilitación comienzan a presentar problemas con la gestión de la información, lo que dificulta la atención al paciente.

Para darle solución a estos problemas, en el 2008 se desarrolló la Aplicación Web, “Sistema Automatizado para la Gestión de la Información en Rehabilitación”. Al terminar su desarrollo esta aplicación no garantiza la conexión entre diferentes clínicas de rehabilitación, pues estas trabajan de forma independiente impidiendo obtener información de un paciente que se haya atendido en distintas clínicas. Tampoco garantiza el acceso a la información desde una determinada área (municipio, provincia, nación). Si se desea realizar un control de las clínicas en el nivel nacional hay que dirigirse a los niveles inferiores para obtener la información. Actualmente esta se manda a través de correos electrónicos o se realiza manualmente, además de no garantizar la utilización de un estándar de clasificación de enfermedades.

De lo antes planteado se puede definir el siguiente **problema**: ¿Cómo lograr que la aplicación web “Sistema Automatizado para la Gestión de la Información en Rehabilitación” garantice el intercambio de información entre los diferentes niveles de dirección del Sistema Nacional de Salud?

Para el desarrollo de la investigación se plantea como **objeto de estudio** los procesos de gestión de la información en las áreas de rehabilitación.

El **campo de acción** comprende los procesos que intervienen en el manejo del flujo de información en los diferentes niveles de dirección en las áreas de rehabilitación.

Para lograr la solución de la problemática se plantea como **objetivo general**: Desarrollar una nueva versión del “Sistema Automatizado para la Gestión de la Información en Rehabilitación” que garantice el intercambio de información entre los diferentes niveles de dirección del Sistema Nacional de Salud.

Para lograr el objetivo propuesto se trazaron las siguientes **tareas de la investigación**:

1. Realizar un análisis de los sistemas informáticos existentes utilizados a nivel nacional e internacional para dar seguimiento a los pacientes atendidos en las clínicas de rehabilitación.
2. Analizar cómo se gestiona la información en el “Sistema Automatizado para la Gestión de la Información en Rehabilitación”.
3. Analizar los mecanismos de envío de información entre los diferentes niveles de dirección del Sistema Nacional de Salud en el Área de Rehabilitación.
4. Asimilar las herramientas y tecnologías definidas en el Área Temática para el desarrollo de sus aplicaciones.
5. Refinar los requerimientos del “Sistema Automatizado para la Gestión de la Información en Rehabilitación”.
6. Analizar el funcionamiento de aplicaciones que consuman Servicios Web.
7. Analizar la información que se gestiona en el “Sistema Automatizado para la Gestión de la Información en Rehabilitación”.
8. Identificar los servicios a consumir brindados por el Sistema de Información para la Salud (SiSalud).
9. Realizar el análisis y diseño de la nueva versión del “Sistema Automatizado para la Gestión de la Información en Rehabilitación”.
10. Implementar la nueva versión del “Sistema Automatizado para la Gestión de la Información en Rehabilitación”.

El presente trabajo, está estructurado en 4 capítulos, distribuidos de la siguiente forma:

**Capítulo 1.** Fundamentación Teórica: descripción de los procesos en los centros de rehabilitación, problemas que motivan la investigación, sistemas existentes vinculados a los procesos que intervienen en el flujo de información en las salas de rehabilitación. Tendencias y tecnologías actuales a considerar: descripción de tendencias y tecnologías seleccionadas para el desarrollo de la propuesta de solución.

**Capítulo 2.** Características del Sistema: definición de los procesos, actores, trabajadores, casos de uso del negocio, diagramas de clases del modelo de objetos del negocio; requisitos funcionales y no funcionales; actores y casos de uso del sistema.

**Capítulo 3.** Análisis y Diseño del Sistema: descripción del diseño a través de diagramas de clases, análisis y diseño de la aplicación; en la realización de los diagramas de diseño se tienen en cuenta estereotipos web, que describen la relación entre las páginas.

**Capítulo 4.** Implementación: Trata los aspectos relacionados con la construcción de la solución propuesta, se modelan los diagramas de componentes y despliegue. Se obtiene el modelo de datos para generar a partir del mismo la base de datos, además de la estrategia de codificación, el tratamiento de errores y como se garantiza la seguridad en la solución del sistema.

### Capítulo 1. Fundamentación Teórica

La fisioterapia a lo largo de la existencia del hombre ha ido evolucionando en beneficio de la humanidad, así como por el desarrollo de la tecnología y la ciencia. En todo el mundo se han abierto salas de rehabilitación, impulsando a que la fisioterapia sea cada vez más utilizada para lograr la incorporación de un paciente a la vida social.

El presente capítulo aborda el tema del origen de la fisioterapia como ciencia, conceptos como rehabilitación, fisioterapia y medicina física. Además, se investiga acerca de los diferentes sistemas que existen a nivel internacional y nacional vinculados a la gestión de información en centros de rehabilitación. Se realiza la descripción de las tendencias y tecnologías seleccionadas, para determinar las que son utilizadas en el desarrollo de la solución propuesta.

#### 1.1. Orígenes y Evolución de la Fisioterapia

La palabra Fisioterapia proviene de la unión de las palabras griegas: *physis*, que significa naturaleza y *therapeia*, que quiere decir tratamiento. Por tanto, desde un punto de vista etimológico, Fisioterapia o *physis-therapeia* significa “Tratamiento por la Naturaleza”, o también “Tratamiento mediante Agentes Físicos”. [2]

La mayoría de los agentes físicos empleados en la fisioterapia moderna ya se emplearon en la antigüedad. Los primeros escritos de Grecia y Roma se refieren a los efectos beneficiosos del sol y del agua, y tanto el ejercicio como los masajes fueron utilizados por los antiguos. [3]

La fisioterapia como ciencia surge en Inglaterra a finales del siglo XIX. Entre las razones del gran aumento de la demanda de los servicios de fisioterapia estaban los excelentes resultados obtenidos en el tratamiento de los heridos durante la II Guerra Mundial y las guerras de Vietnam y Corea, los accidentes laborales, el aumento de las discapacidades crónicas consecuencia del número creciente de ancianos en la población, y el rápido desarrollo de los programas hospitalarios y de asistencia médica.

En la actualidad, gracias a los avances de la tecnología, la fisioterapia dispone a su alcance el uso de numerosos agentes físicos (*masaje, agua, sonido, electricidad, movimiento, luz, calor, frío...*), pero sin descuidar o abandonar el desarrollo e impulso de nuevas concepciones y métodos de terapia manual,

# Capítulo 1. Fundamentación Teórica

---

principal herramienta del fisiatra para la prevención, tratamiento, curación y recuperación de un gran número de patologías y lesiones.

Se podría decir que en el proceso de rehabilitación de un enfermo, interviene además de la fisioterapia, el especialista, la logopedia, la terapia ocupacional, la psicología, etc.

La **Fisioterapia y Rehabilitación** no significan lo mismo. La rehabilitación es la recuperación física, psíquica, social y laboral, constituye en general la recuperación global del paciente, mientras que la fisioterapia sólo se ocupa de la recuperación física.

## **Medicina Física y Rehabilitación**

Especialidad Médica que desarrolla una serie de medidas de promoción de salud, prevención, diagnóstico, investigación y tratamiento, para obtener la máxima reincorporación posible a la comunidad de los pacientes con secuelas congénitas o adquiridas. [4]

La especialidad utiliza esencialmente como terapéutica los agentes físicos, naturales y artificiales, métodos de reeducación funcional, métodos de terapia ocupacional además de los métodos de la ortopedia técnica. Se encarga de implementar, dirigir y controlar el desarrollo del proceso de rehabilitación.

## **1.2. Rehabilitación Integral a Nivel Internacional**

El tipo de discapacidad determina que apoyo puede necesitar el individuo en su proceso de rehabilitación integral, por lo tanto, es necesario manejar los conceptos básicos en el proyecto de creación de un centro de rehabilitación integral para los pacientes con discapacidades.

Discapacidad en términos genéricos significa déficit, limitación en la actividad y restricciones en la participación. Existen deficiencias temporales o permanentes; progresivas, regresivas o estáticas; intermitentes o continuas.

La rehabilitación de acuerdo a las Resoluciones 48/96 de la Asamblea General de las Naciones Unidas: *“...es un proceso encaminado a lograr que las personas con discapacidad estén en condiciones de alcanzar y mantener un estado funcional óptimo desde el punto de vista físico, sensorial, intelectual, psíquico o social, de manera que cuenten con medios para modificar su propia vida y ser más independientes. La rehabilitación puede abarcar medidas para proporcionar o restablecer funciones o para compensar la pérdida o falta de una función o limitación funcional. El proceso de Rehabilitación no*

# Capítulo 1. Fundamentación Teórica

---

*supone la prestación de atención médica preliminar. Abarca una amplia variedad de medidas y actividades, desde la rehabilitación más básica y general hasta las actividades de orientación específica, como por ejemplo la rehabilitación profesional". [5]*

La Asociación Dominicana de Rehabilitación, es una prestigiosa institución sin fines de lucro, fue fundada hace más de 40 años prestando servicios a más de siete millones de personas, la mayoría con escasos recursos económicos. Los resultados de este enorme esfuerzo sostenido durante casi cinco décadas, han dado sus frutos positivos, se ha mantenido inquebrantable la filosofía de "proporcionar el más eficiente servicio de rehabilitación para la atención integral a las personas con discapacidad física o intelectual, sin importar su condición socioeconómica". [6]

Los organismos internacionales recomiendan fortalecer la capacidad y eficiencia de las organizaciones que trabajan en la atención al sector de la discapacidad, desarrollar un sistema de estadísticas para formular programas y sobre todo un ambiente de comprensión de la sociedad hacia el problema.

Enrique Elissalde, quien ejerció la Presidencia de la Fundación Braille en Uruguay, encargada de posibilitar el acceso a la información a niños y adultos con discapacidad visual expresó: "Estoy cada vez más convencido que la humanidad no puede darse el lujo de prescindir de las personas con discapacidad, las necesita para construir un mundo justo y feliz, necesita el trabajo y la energía de estas personas porque la tarea es demasiado grande y solo habrá un mundo mejor si todos, absolutamente todos, participamos en su construcción". [7]

## **1.3. Rehabilitación Integral en Cuba**

En Cuba, el servicio de rehabilitación integral es una estrategia del sistema de salud para lograr la atención adecuada y la reincorporación social del discapacitado, así como garantizar una mejor calidad de vida para la población.

Su objetivo primordial es disminuir el impacto de la discapacidad por medio de la ampliación de coberturas y la integración de las personas teniendo en cuenta que el 80 por ciento de la población en algún momento de su vida sufre de un proceso de discapacidad.

En la actualidad, el país cuenta con más de 400 servicios de rehabilitación integral, dotados de modernos equipos. En zonas montañosas donde es difícil el acceso también existen estos servicios, desarrollándose con recursos propios y personal muy calificado, que brindan una atención adecuada. Se

# Capítulo 1. Fundamentación Teórica

---

han formado unos 17 mil rehabilitadores en más de 35 especialidades médicas, que incluye, fisioterapia, ultrasonido, logopedia, podología, defectología y terapia ocupacional, entre otras.

## 1.4. Sistemas Automatizados Existentes

Los sistemas de rehabilitación que se desarrollen deben tener características comunes y cumplir con objetivos generales tales como:

- Proporcionar a las clínicas un sistema de información homogéneo, abierto y tecnológicamente avanzado.
- Proporcionar una mayor calidad asistencial a los pacientes de las clínicas
- Facilitar el acceso a la información por parte de todos los implicados.
- Facilitar el acomodamiento de las clínicas a diversos procedimientos, fruto del ajuste de las mismas a sistemas de calidad.

A nivel internacional existen diferentes aplicaciones que han sido elaboradas y diseñadas con la finalidad de gestionar todo lo relacionado con la información en clínicas y salas de rehabilitación. Estas no son utilizadas en Cuba porque no están en correspondencia con las características del sistema de salud que responde al proyecto socialista que construye el país. Por lo tanto no es factible utilizarlos en las clínicas cubanas, ya que estos sistemas fueron creados para centros donde el paciente tiene que pagar por los servicios prestados. Además, son sistemas propietarios que en ocasiones no cobran el uso de los mismos, pero si por darle mantenimiento online y por adquirir las actualizaciones posteriores.

- ✓ **FisioGest (Software de Gestión para Clínicas de Fisioterapia):** ha sido concebido como una herramienta imprescindible para la gestión moderna y eficaz de centros clínicos de rehabilitación. Posibilita una gestión integral de información en centros de fisioterapia a través de una interfaz novedosa y atractiva. Permite actualizaciones en-línea periódicamente, donde se van añadiendo nuevas funcionalidades. Proporciona un nivel de seguridad avanzado mediante su sistema de autenticación de usuarios, clasificándolos por perfiles. La aplicación gestiona pacientes, genera informes y certificados; gestiona citas, y además la facturación.

Este software fue desarrollado por Orion Network Communication S.L. (ORIONET), el cual solicita el pago de 470 € para adquirir la primera licencia del producto, 150 € por cada licencia adicional, 15 € por el mantenimiento mensual y 5 € por el mantenimiento mensual por cada licencia .[8]



## Capítulo 1. Fundamentación Teórica

---

- ✓ **FisioSalus v6.5:** desarrollado por la empresa QSOFT desde 1995, permite la gestión integral de centros de fisioterapia. En un solo programa se gestiona toda la información relativa a las 3 grandes áreas de gestión del centro que son Agenda para la planificación de las visitas y el resto de las actividades del centro, Historia Clínica para el almacenamiento y consulta de los datos clínicos, emitir informes, etc., Administración para gestionar la facturación y cobro a pacientes, deudas, proveedores.

Está limitado a 50 accesos e instala una base de datos con clientes y visitas ficticias para ver cómo funciona el programa. En el sitio de descargas Downloads-portal.com (<http://spanish.downloads-portal.com/>) se puede encontrar la descarga de la versión Demo de FisioSalus por un precio de 1200 €.

- ✓ **ALC Fisio:** fue creado por la empresa ALC Desinco Sistemas, permite administrar y gestionar de forma totalmente integral un centro de fisioterapia, controlando cada detalle y a través de una interfaz agradable al usuario. El programa es fácil de usar y cuenta con un sistema de bases de datos potente y eficaz, prácticamente sin límites de registros. Permite configurar todos los datos del centro en que se está usando, dar de alta a todos los clientes y llevar un exhaustivo control de citas programadas, además de la facturación del centro. En varios sitios de descargas se puede obtener una versión de ALC Fisio pero esta tiene como desventaja que permite introducir como máximo 10 pacientes. [9]

### **Cambios recientes en ALC Fisio:**

- Añadida la búsqueda "Citas entre fechas".
  - Incluido el proceso "Exportar las citas de la lista", la opción "Importar citas VCalendar", la opción "Pacientes aviso cumpleaños" y opción de diferenciación según los distintos estilos de usuarios.
- ✓ **Fisio Office 2008:** gerenciador completo para clínicas y consultorios de fisioterapia. Realiza búsqueda avanzada y completa para la administración de oficinas. Incluye innovadoras y únicas herramientas que ayudan a simplificar el trabajo en estos centros, brinda información clínica detallada y completa.

Este software fue creado por Rh! Software Sociedad Limitada, dicha empresa radica en Brasil y los precios están dados en Reales (R\$), moneda oficial de Brasil, el cual solicita el pago de R\$ 265

# Capítulo 1. Fundamentación Teórica

---

para adquirir la versión estándar del producto, R\$ 650 para la versión profesional y por la licencia de redes R\$ 250 cada una. [10]

- ✓ **IPFisio:** constituye una solución informática para la gestión integral de la información en clínicas de fisioterapia. La aplicación permite gestionar todos los aspectos de su clínica: historiales clínicos, documentación, facturación, entidades, etc., de una manera rápida y sencilla. La solución incluye una aplicación para gestionar información. [11]

Brinda servicios como: actualizaciones de la aplicación, copias de seguridad, soporte técnico 24 horas al día y todo el año. El sistema IPFisio fue creado por iProyectos, empresa creadora de software para diferentes sistemas, que solicita el pago de 49 € mensualmente por su adquisición.

En estos momentos existe un sistema para rehabilitación en el país específicamente en el hospital CIMEQ (Centro de Investigaciones Médico Quirúrgicas) el cual se encarga de procesar toda la información en este centro, es manipulado por la recepcionista. Está desarrollado en FoxBase, presenta una interfaz alfanumérica y el almacenamiento se realiza en ficheros. Presenta las siguientes limitaciones: escasa documentación, no está diseñado para su uso en otras clínicas y la información de toda la sala de rehabilitación es procesada por una sola persona, lo que implica un doble trabajo y gasto de tiempo en el procesamiento de la información.

Basándose en las limitaciones de la aplicación anteriormente descrita, con vista a mejorar el proceso de flujo de información y la necesidad de la informatización en esta área de la salud, la Universidad de las Ciencias Informáticas (UCI) desarrolló en el año 2008, la aplicación web “Sistema Automatizado para la Gestión de Información en Rehabilitación”. Esta aplicación, teniendo en cuenta los sistemas existentes, da soporte a los procesos relacionados con el flujo de información médica aunque las funcionalidades que tiene implementada deben ser llevadas a las necesidades actuales del cliente.

## 1.5. Tecnologías

Teniendo en cuenta las necesidades vistas y las características del entorno donde se aplicará la solución propuesta, se realizó un estudio de las tecnologías actuales posibles a emplear, descritas a continuación.

### Internet

Internet consiste en el conjunto de redes: redes de ordenadores y equipos físicamente unidos mediante cables que conectan puntos de todo el mundo. Gigantesca Red que se difumina en ocasiones porque los

# Capítulo 1. Fundamentación Teórica

---

datos pueden transmitirse vía satélite, o a través de servicios como la telefonía celular, o porque a veces no se sabe muy bien a dónde está conectada. Al contrario de lo que se piensa comúnmente, Internet no es sinónimo de World Wide Web (WWW, o "la Web"). [12]

La WWW consiste desplegar gráficos y usar el mouse para "navegar" por lugares en Internet. Anteriormente el acceso era complicado y poco interactivo: en nuestras pantallas sólo se mostraban textos y se debían usar instrucciones complicadas o programas manejados con el teclado.

Internet incluye aproximadamente 5000 redes en todo el mundo y más de 100 protocolos distintos basados en (Protocolo de Transferencia y Control) TCP/IP, que se configura como el protocolo de la red. Los servicios disponibles en la red mundial de PC, han avanzado mucho gracias a las nuevas tecnologías de transmisión de alta velocidad, como DSL (Línea de Abonado Digital) y Wireless (Comunicación Inalámbrica).

## Aplicaciones Web

Una Aplicación Web es una especialización de las aplicaciones Cliente/Servidor, están comúnmente estructuradas como una aplicación en tres capas, son conocidas como aplicaciones servidor. En su forma más general, el navegador web es la primera capa, la segunda capa realiza la función de intermediaria entre la primera y tercera capa y se caracteriza por el uso de tecnología web dinámica ejemplos: CGI (Common Gateway Interface o Interfaz de Entrada Común), PHP (Hypertext Preprocessor o Procesador de Hipertexto), ASP (Active Server Pages o Servidor de Páginas Activas), Java y la tercera capa se conoce como capa de base de datos. [13]

Ventajas:

- ✓ Permite a los clientes o usuarios migrar de sistema operativo o cambiar de hardware libremente sin afectar el funcionamiento de las aplicaciones servidor.
- ✓ No se requieren complicadas combinaciones Hardware/Software para utilizar estas aplicaciones, los requerimientos mínimos consisten en un cómputo con un buen navegador web.
- ✓ Facilita el trabajo a larga distancia, permite trabajar desde cualquier PC con conexión a Internet.

Desventajas:

- ✓ Necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.

- ✓ Elementos de interacción muy limitados.

## **Servidor Web Apache**

Servidor Web hecho por excelencia, su configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. Apache muestra los beneficios del trabajo cooperativo dentro de Internet. Apache, siendo un programa desarrollado en un ambiente software libre, es reconocido en muchos ámbitos empresariales y tecnológicos, por las siguientes razones: [14]

- ✓ Corre en una multitud de Sistemas Operativos, por lo que es prácticamente universal.
- ✓ Tecnología de código fuente abierta y gratuita. El hecho de ser gratuita es importante pero no tanto como el hecho de que se trate de código fuente abierto.
- ✓ Es un servidor altamente configurable de diseño modular. Es muy sencillo ampliar las capacidades del servidor Web Apache. Actualmente existen muchos módulos para este servidor, que son adaptables.
- ✓ Trabaja con Perl, PHP y otros lenguajes de script.
- ✓ Permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor.

## **Yahoo User Interface (YUI)**

Yahoo User Interface Library o Librería de Interfaz de Usuarios de Yahoo es un conjunto de utilidades y controles, escrito en JavaScript para desarrollar aplicaciones Web interactivas usando técnicas DOM (Document Object Model o Modelo en Objetos para la representación de Documentos), DHTML (Dynamic Hypertext Markup Language o Lenguaje de Marcado de Hipertexto Dinámico) y AJAX (Asynchronous JavaScript And XML o JavaScript Asíncrono y XML). Este incluye también un conjunto de fuentes Cascading Style Sheets u Hojas de Estilo en Cascada (CSS). Todos los componentes YUI han sido desarrollados bajo Licencia BSD Open Source y están disponibles para todo tipo de uso.

Características: Están disponibles dos tipos de componentes diferentes: Utilidades y Controles: Las YUI Utilidades simplifican el desarrollo para la compatibilidad entre Navegadores basados en técnicas DOM, DHTML y AJAX. Los controles de YUI proporcionan elementos visuales altamente interactivos del diseño para sus aplicaciones Web. Estos elementos se crean y se manejan íntegramente del lado del cliente (usuario) y nunca requieren de una recarga de página. [15]

## Servicios Web

El concepto "Servicios Web" se refiere de forma genérica a la técnica que permite a los servicios que utilizan distintos sistemas operativos y se ejecutan en diferentes plataformas "comunicarse" entre ellos. La idea es que todo tipo de aplicaciones ejecutadas en cualquier ordenador puedan llegar a compartir información e interactuar con otras aplicaciones ejecutadas en otro ordenador.

Mediante los Servicios Web, las compañías pueden integrar los procesos empresariales más allá de los cortafuegos con sus proveedores, socios tecnológicos y clientes, sin necesidad de conocer los sistemas de TI de dichos proveedores, socios tecnológicos y clientes. Cada parte conserva el control de su propio entorno, pero puede comunicarse con las demás con total facilidad.

## Protocolo Simple de Accesos a Datos (SOAP)

Existen diferentes protocolos creados para facilitar la comunicación entre aplicaciones, es decir, protocolos para la comunicación utilizando Servicios Web, los más conocidos son: RPC (Remote Procedure Call o Llamada a Procedimiento Remoto) de Sun Microsystems, DCE (Distributed Computing Environment o Sistema de Software para Computación Distribuida) de Microsoft, RMI (Java Remote Method Invocation o Método Invocación Remota de Java) de Java y SOAP.

En el núcleo de los Servicios Web se encuentra el protocolo simple de acceso a datos SOAP, que proporciona un mecanismo estándar de empaquetar mensajes. Este ha recibido gran atención debido a que facilita una comunicación del estilo RPC entre un cliente y un servidor remoto. Algunas de sus ventajas son: [16]

- ✓ No se asocia a ningún lenguaje.
- ✓ No se encuentra fuertemente asociado a ningún protocolo de transporte.
- ✓ No está atado a ninguna infraestructura de objeto distribuido.
- ✓ Aprovecha los estándares existentes en la industria.

## Lenguaje Unificado de Modelado (UML)

UML es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra una gran cantidad de software. Está compuesto por diversos elementos gráficos que se combinan para conformar diagramas. Debido a que el UML es un lenguaje, cuenta con reglas para

# Capítulo 1. Fundamentación Teórica

---

combinar tales elementos. En lugar de indicarle a usted cuáles son los elementos y las reglas, véase directamente los diagramas ya que se utiliza para hacer el análisis del sistema. Es un lenguaje que permite la modelación de sistemas con tecnología orientada a objetos. [17]

## Lenguajes de Programación

Al mundo encontrarse inmerso en un desarrollo continuo, son muchos los lenguajes para programar Aplicaciones Web que han ido surgiendo, en la actualidad estos lenguajes se dividen en: [18]

- ✓ Lenguajes del lado del Servidor: son los lenguajes que son reconocidos, ejecutados e interpretados por el propio servidor y son enviados al cliente en un lenguaje comprensible.
- ✓ Lenguajes del lado del Cliente: independiente del servidor, lo cual permite que la página pueda ser albergada en cualquier sitio.

Entre los diferentes lenguajes de programación se encuentra el JavaScript, utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página web. Su uso se basa fundamentalmente en la creación de efectos especiales en las páginas y la definición de interactividades con el usuario.

Los lenguajes del lado del servidor más utilizados en el mundo por los desarrolladores de aplicaciones Web son: PHP, ASP, PERL (Practical Extracting and Reporting Lenguaje o Lenguaje de Programación para Extraer Información de Archivos de Texto) y JSP (JavaServer Pages). En el caso de las Aplicaciones que están integradas a Servicios Web, el lenguaje más usado es el XML, constituyendo el formato estándar para los datos que se va a intercambiar.

## Hypertext Preprocessor (PHP)

Es más conocido en el mundo entero por las siglas PHP, tecnología de código abierto muy útil para diseñar de forma rápida y eficaz Aplicaciones Web conectadas a Bases de Datos. Potente lenguaje de secuencia de comandos diseñado específicamente para permitir a los programadores crear Aplicaciones Web con distintas prestaciones de forma rápida.

Una de sus características más potentes es el soporte para varios gestores de base de datos entre las que se pueden mencionar InterBase, MicrosoftSQL, MySQL, Oracle, Informix, PostgreSQL, entre otras. Es Software Libre, por lo que es menos costoso utilizarlo. Lenguaje multiplataforma, funciona tanto para Unix

# Capítulo 1. Fundamentación Teórica

---

(con Apache) como para Windows (con Microsoft Internet Information Server) de forma que el código que se haya creado para una de ellas no tiene porqué modificarse al pasar a la otra.

El código PHP es transparente al usuario, es ejecutado en el servidor y el resultado es enviado al navegador. El resultado es normalmente una página HTML (Hypertext Markup Language). Por lo que al usuario le parecerá que está visitando una página HTML que cualquier navegador puede interpretar. Es soportado por cualquier navegador, pero para que sus páginas funcionen tiene que ser soportado por el servidor.

## **HyperText Markup Language (HTML)**

Es un lenguaje de programación para crear páginas web conocido como código HTML, consiste en un lenguaje de marcas de hipertexto. Permite la inclusión de textos, imágenes fijas y móviles, vídeo, archivos, entre otros. Lenguaje en el que se escriben los documentos que se visualizarán a través de WWW.

## **JavaScript**

JavaScript es un lenguaje de programación utilizado para crear pequeños programas encargados de realizar acciones dentro del ámbito de una página web. Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga de procesamiento.

Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado. Con JavaScript se pueden crear efectos especiales en las páginas y definir interactividades con el usuario. El navegador del cliente es el encargado de interpretar las instrucciones JavaScript y ejecutarlas para realizar estos efectos e interactividades, de modo que el mayor recurso, y tal vez el único, con que cuenta este lenguaje es el propio navegador.

## **Extensible Markup Language (XML)**

Lenguaje Extensible de Marcado (XML), es el formato estándar para los datos que se vayan a intercambiar entre mediante el consumo de Servicios Web. Es un lenguaje muy similar a HTML pero su función principal es describir datos y no mostrarlos como es el caso de HTML. Lenguaje estandarizado, y posibilita trabajar desde cualquier plataforma. Admite la comunicación entre dos aplicaciones sin importar en el lenguaje en que están desarrolladas.

## Web Services Description Languages (WSDL)

Lenguaje de interfaz pública para los Servicios Web. Es una descripción basada en XML, permite la descripción de los servicios web desplegados. WSDL se utiliza también para la localización y ubicación de estos servicios en Internet. Un documento WSDL no es más que un documento XML que describe ciertas características propias de un servicio web, así como su localización y aquellos parámetros y métodos que soporta.

## Framework

El Framework es una aplicación genérica incompleta y configurable a la que se pueden añadir las últimas piezas para construir una aplicación concreta.

Los objetivos principales que persigue un Framework son: acelerar el proceso de desarrollo, reutilizar código ya existente y promover buenas prácticas de desarrollo como el uso de patrones. Un Framework Web, se define como un conjunto de componentes por ejemplo clases en java, descriptores y archivos de configuración en XML que componen un diseño reutilizable que facilita y agiliza el desarrollo de Aplicaciones Web. Tipos de Framework Web: [19]

- ✓ Orientados a la interfaz de usuario, como Java Server Faces
- ✓ Orientados a aplicaciones de publicación de documentos, como Cocoon.
- ✓ Orientados a la parte de control de eventos, como Struts y algunos que incluyen varios elementos como Tapestry.

## CodeIgniter

CodeIgniter es un Framework para desarrollo de aplicaciones en PHP. Es de código abierto, tiene una interfaz simple y un acceso a sus librerías bien estructurado.

Es liviano, bastante fácil de utilizar, altamente configurable y cuenta con un gran grupo de desarrollo en Internet.

CodeIgniter PHP es adecuado si:

- ✓ Necesitas un rendimiento excepcional.
- ✓ Necesita amplia compatibilidad con el estándar de las cuentas de hosting que ejecutar una gran variedad de versiones y configuraciones de PHP.



# Capítulo 1. Fundamentación Teórica

---

- ✓ Necesita un marco que exija casi cero configuraciones.
- ✓ Necesita un marco que no requiera que usted use líneas de comandos.
- ✓ Necesita un marco que no le pida que se adhiera a la codificación de normas restrictivas.
- ✓ Usted necesita evitar complejidad, favoreciendo soluciones simples.
- ✓ Necesita claro, documentación exhaustiva.

## Sistemas Gestores de Base de Datos (SGBD)

Un Sistema Gestor de Base de Datos (SGBD) está definido como el conjunto de programas que permiten crear y mantener una base de datos, asegurando su integridad, confidencialidad y seguridad. Por tanto debe permitir:

- ✓ Definir una base de datos: especificar tipos, estructuras y restricciones de datos.
- ✓ Construir la base de datos: guardar los datos en algún medio controlado por el mismo SGBD.

Algunas de las características deseables en un Sistema Gestor de Base de Datos son:

- ✓ Control de la redundancia.
- ✓ Restricción de los accesos no autorizados: cada usuario debe tener permisos de acceso y autorización.
- ✓ Cumplimiento de las restricciones de integridad: el SGBD debe ofrecer recursos para definir y garantizar el cumplimiento de las restricciones de integridad.

Entre los SGBD comúnmente utilizados en el mundo están: Oracle, MySQL, Microsoft SQL Server, PostgreSQL, InterBase, entre otros. Todos estos presentan un enfoque relacional con un buen basamento matemático centrado en el Álgebra Relacional.

- ✓ **Oracle:** requiere de una licencia para poderlo utilizar, es decir, es necesario pagar para su utilización.
- ✓ **Microsoft SQL Server:** no es multiplataforma, solo puede ser utilizado con el sistema operativo Windows que está patrocinado por la compañía Microsoft, no es gratuito.
- ✓ **MySQL:** soporta un subconjunto de SQL92, multiplataforma.
- ✓ **PostgreSQL:** soporta un subconjunto de SQL92, es gratuito.

## PostgreSQL.

PostgreSQL es un sistema de gestión de bases de datos Objeto-Relacionales (ORDBMS) libre, liberado bajo la licencia BSD (Berkeley Software Distribution). Es una alternativa a otros sistemas de bases de datos de código abierto (como Firebird y MaxDB), así como sistemas propietarios como Oracle y SQLServer. El mismo ha sido desarrollado de varias formas desde 1977.

En 1994, Andrew Yu y Jolly Chen añadieron un intérprete de lenguaje SQL a Postgres. Postgres95 fue lanzada a continuación a la Web para que encontrara su propio hueco en el mundo como un descendiente de dominio público y código abierto del código original Postgres de Berkeley.

En 1996, debido a un nuevo esfuerzo de código abierto y a la incrementada funcionalidad del software, Postgres fue renombrado a PostgreSQL, tras un breve periplo como Postgres95. El proyecto PostgreSQL sigue actualmente un activo proceso de desarrollo a nivel mundial gracias a un equipo de desarrolladores y contribuidores de código abierto. PostgreSQL está ampliamente considerado como el sistema de bases de datos de código abierto más avanzado del mundo. Posee muchas características que tradicionalmente sólo se podían ver en productos comerciales de alto calibre.

Las principales mejoras en PostgreSQL incluyen: [20]

- ✓ Los bloqueos de tabla han sido sustituidos por el control de concurrencia multiversión, el cual permite a los accesos de sólo lectura continuar leyendo datos consistentes durante la actualización de registros, y permite copias de seguridad en caliente desde `pg_dump` mientras la base de datos permanece disponible para consultas.
- ✓ Se han implementado importantes características del motor de datos, incluyendo subconsultas, valores por defecto, restricciones a valores en los campos (constraints) y disparadores (triggers).
- ✓ Se han añadido características adicionales que cumplen el estándar SQL92, incluyendo claves primarias, identificadores entrecomillados, forzado de tipos cadenas literales, conversión de tipos y entrada de enteros binarios y hexadecimales.
- ✓ Los tipos internos han sido mejorados, incluyendo nuevos tipos de fecha/hora de rango amplio y soporte para tipos geométricos adicionales.
- ✓ La velocidad del código del motor de datos ha sido incrementada aproximadamente en un 20-40%, y su tiempo de arranque ha bajado el 80% desde que la versión 6.0 fue lanzada.

## 1.6. Arquitectura

### Modelo Cliente/Servidor

La arquitectura Cliente/Servidor es la integración distribuida de un sistema en red, con los recursos, medios y aplicaciones que, definidos modularmente en los servidores, administran, ejecutan y atienden las solicitudes de los clientes; todos interrelacionados física y lógicamente, compartiendo datos, procesos e información. Se establece así un enlace de comunicación transparente entre los elementos que conforman la estructura. Entre las principales características de la arquitectura Cliente/Servidor, se pueden destacar las siguientes:

- ✓ El cliente no necesita conocer la lógica del servidor, sólo su interfaz externa.
- ✓ El cliente no depende de la ubicación física del servidor, ni del tipo de equipo físico en el que se encuentra, ni de su sistema operativo.
- ✓ Los cambios en el servidor implican pocos o ningún cambio en el cliente.

Ventajas del Modelo:

- ✓ El servidor no necesita potencia de procesamiento, parte del proceso se reparte a los clientes.
- ✓ Se reduce el tráfico de red considerablemente.

### Modelo-Vista-Controlador (MVC)

Es una de las guías más usadas para el diseño de arquitecturas de aplicaciones que ofrezcan una fuerte interactividad con usuarios. Este patrón organiza la aplicación en tres modelos separados:

- ✓ Modelo: representa los datos de la aplicación y las reglas de negocio.
- ✓ Vistas: representa a los formularios de entrada y salida de la información.
- ✓ Controlador: representa al conjunto de controladores o páginas servidoras que procesan las peticiones de los usuarios y controla el flujo de ejecución del sistema.

## 1.7. Metodología

### Proceso Unificado de Desarrollo (RUP)

El Proceso Unificado de Desarrollo del Software más conocido por sus siglas RUP fue publicado en 1988 como resultado de varios años de experiencia. Es un proceso para el desarrollo de un software que define

# Capítulo 1. Fundamentación Teórica

---

claramente *quién, cómo, cuándo y qué debe hacerse* en el proyecto. RUP es dirigido por casos de uso, centrado en la arquitectura e iterativo e incremental.

RUP cubre el ciclo de vida de desarrollo de un proyecto y toma en cuenta las mejores prácticas a utilizar en el modelo de desarrollo de software. Es un proceso basado en componentes, que utiliza el Lenguaje Unificado de Modelado (UML) para preparar todos los esquemas de un sistema software.

## 1.8. Herramientas

### Diseño Gráfico

Existen muchas herramientas que permiten realizar el Diseño Gráfico de una Aplicación Web, entre ellas se encuentra el Macromedia Fireworks, herramienta de fácil maniobrar para cualquier tipo de gráficos, por estar integrado al Dreamweaver se hace más fácil su uso, por lo que simplifica el trabajo con imágenes en páginas Web. El Adobe Photoshop CS3 durante cualquier tipo de desarrollo permite al usuario desplegar sus conocimientos con total libertad.

### Diseño de Interfaz

Macromedia MX es uno de los editores de desarrollo Web más utilizado a nivel profesional para la creación de sitios Web. Su amplio abanico de herramientas permite crear desde la más simple página Web personal hasta el sitio Web más completo y complejo para una gran empresa y utilizar casi todos los recursos de la Web.

Consiste en un editor de HTML para el diseño, codificación y desarrollo de páginas, sitios y aplicaciones Web; permite entre otras cosas la edición visual, es decir, crear páginas rápidamente sin tener la necesidad de escribir una línea de código. Entre otros códigos que permite teclear se encuentran: JavaScript, PHP, CSS, XML y más. Soporta tecnologías como Microsoft ASP, Microsoft ASP.NET, JSP y PHP.

Es importante conocer que actualmente el paquete completo de Macromedia fue comprado por la compañía Adobe. Lanzando este año la última versión del Dreamweaver dentro de la Suite Adobe CS3, con el nombre de Adobe Dreamweaver CS3.

### Zend Studio

Ambiente de Desarrollo Integrado o Integrated Development Environment (IDE) disponible para los desarrolladores profesionales, agrupando los diferentes componentes para el desarrollo de aplicaciones

# Capítulo 1. Fundamentación Teórica

---

PHP. Contiene un comprensivo conjunto de herramientas de edición, depurado, análisis, optimización y Base de Datos.

## Visual Paradigm

Visual Paradigm es una herramienta Case Visual con UML 2.0 que ofrece:

- ✓ Entorno de creación de diagramas para UML 2.0.
- ✓ Diseño centrado en casos de uso y enfocado al negocio que generan 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 (versión profesional) e inversa.
- ✓ Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- ✓ Disponibilidad de múltiples versiones, para cada necesidad.
- ✓ Disponibilidad en múltiples plataformas.

## HelpMaker

HelpMaker es un editor de archivos de ayuda que permite crear este tipo de documentos en tres formatos diferentes: WinHelp, RTF y HTML-Help. El procesador de textos incluido cuenta con diversas herramientas para dar formato al texto y configurar un documento limpio y ordenado: alineación, negrita, cursiva, subrayado, tachado, índice y subíndice, viñetas, sangría, etcétera.

La información es almacenada en un único fichero y puede imprimirse en un documento en formato PDF de varias páginas. HelpMaker cuenta con una interfaz sencilla que permite una organización efectiva de los temas que se van a incluir y que siempre está al alcance de la mano por si es necesario añadir un nuevo elemento.

## Capítulo 1. Fundamentación Teórica

---

En este capítulo se hizo referencia al surgimiento y principales conceptos relacionados con la rehabilitación y la fisioterapia. Se analizaron los principales software utilizados en el manejo del flujo de la información en las salas de rehabilitación a nivel nacional e internacional.

Luego de un análisis detallado de las diferentes tecnologías y herramientas a utilizar en el desarrollo de la aplicación, se proponen: RUP como metodología de desarrollo de software, pues se adapta a proyectos de largo plazo donde existe una forma organizada de asignar tareas, y UML como lenguaje de modelado. Además, se determinó como lenguaje de programación PHP5 y PostgreSQL como gestor de base de datos.

## Capítulo 2. Características del Sistema

La calidad de un producto de software, está estrechamente relacionada con la calidad que se inicie su proceso de desarrollo. Se basa en comprender la estructura de la organización en la cual se va a implantar el producto, comprender los problemas actuales y asegurar que los consumidores, usuarios finales y desarrolladores tengan un entendimiento común de la organización.

En el presente capítulo se abordan características del sistema como: definición de los procesos, actores, trabajadores, casos de uso del negocio, diagramas de clases del modelo de objetos del negocio; requisitos funcionales y no funcionales; actores y casos de uso del sistema.

### 2.1 Descripción del flujo de trabajo en las Salas de Rehabilitación

El análisis del flujo de trabajo en las salas de rehabilitación integral permite conocer su funcionamiento para producir uno o varios resultados. Este revela problemas potenciales tales como: la circulación de información doble, pasos innecesarios, pérdida y acumulación de documentos, entre otros.

En las salas de rehabilitación, el flujo comienza cuando el paciente llega a la clínica donde es atendido por la recepcionista que es la encargada de planificarle las consultas y los tratamientos. En caso que el paciente vaya a una consulta la recepcionista lo remitirá con el especialista que es el encargado de orientarle los tratamientos y según la evolución que tenga podrá darle de alta. Si el paciente acude a la clínica para recibir los tratamientos la recepcionista lo remite con los técnicos que son los encargados de aplicárselos y seguir su evolución.

### 2.2 Descripción de los Procesos del Negocio

El día de la consulta del paciente, el especialista le llena una hoja de cargo, donde recoge una serie de datos entre los que se encuentran: impresión diagnóstica, problemas encontrados y le indica cuales son los tratamientos que se debe aplicar. Al finalizar la consulta, es remitido con la recepcionista, que será la encargada de hacer la planificación de los tratamientos, teniendo en cuenta la disponibilidad de los turnos. Se lleva a cabo un seguimiento al paciente tomando su asistencia a los tratamientos, conformando así la hoja de evolución.

Al terminar el ciclo de tratamientos, el especialista unido a los técnicos valoran la evolución del paciente, si es satisfactoria, este es dado de alta, si tiene un resultado adverso o negativo, el especialista le realizará

## Capítulo 2. Características del Sistema

---

una re-consulta. El paciente solo es dado de alta cuando se encuentra completamente recuperado, si un tratamiento en el transcurso de su aplicación afecta su salud, se le retirará y es remitido urgentemente con un especialista.

### **2.3 Análisis crítico de ejecución de los procesos actuales**

En Cuba, existen problemas con la gestión de información en las salas de rehabilitación integral, entre ellos sobresalen: la pérdida de información, la acumulación de documentos y la falta de un control estadístico eficiente sobre la información del paciente. Todos estos problemas se deben a que la información es manipulada manualmente.

Se estudiaron aplicaciones internacionales que se han desarrollado para gestionar la información en centros de rehabilitación, determinándose que no se ajustan al sistema de salud cubano. Al no disponer el país de algún medio o herramienta que le permita realizar estas tareas de forma más sencilla, se propuso la elaboración de un sistema informático que solucione los problemas que se presentan con la gestión de la información de dichos centros.

La elaboración del software comenzó en el año 2008 por el proyecto “Fisioterapia” de la Universidad de las Ciencias Informáticas. Se le incluyó el servicio de Administración que permite gestionar toda la información con la que se trabaja (estructura organizativa, permisos de usuarios, roles del sistema, módulos de la aplicación, información sobre los usuarios, etc.). Se desarrolló el servicio de agenda, donde se realizan las planificaciones de consultas, tratamientos y la realización de la consulta médica, entre otras actividades.

Al terminar su desarrollo esta aplicación no garantiza la conexión entre diferentes clínicas de rehabilitación, pues estas trabajan de forma independiente impidiendo obtener información de un paciente que se haya atendido en distintas clínicas, no garantiza el acceso a la información desde una determinada área (municipio, provincia, nación), si se desea realizar un control de las clínicas en el nivel nacional hay que dirigirse a los niveles inferiores para obtener la información que actualmente se manda a través de correos electrónicos o se realiza manualmente, además de no utilizar un estándar de clasificación de enfermedades.



### 2.4 Reglas del Negocio

Se identificaron las siguientes reglas que debe seguir la aplicación que se desarrolle, a fin de respetar y garantizar las restricciones que existen en el negocio:

- ✓ La secretaria es la encargada de manipular toda la información de los pacientes, se encarga de registrar a los pacientes, planificarle las consultas y los tratamientos.
- ✓ El médico fisioterapeuta del centro de rehabilitación es el encargado de realizarle la consulta al paciente y orientarle los tratamientos.
- ✓ El paciente es consultado por un médico fisioterapeuta del centro de rehabilitación, donde le prescribe los tratamientos que deberá aplicarse.
- ✓ Para poder ser recibido por un médico fisioterapeuta del centro de rehabilitación deberá tener la consulta planificada.
- ✓ Para poder recibir un tratamiento el paciente tiene que ser consultado antes por el médico fisioterapeuta del centro de rehabilitación que es el encargado de orientarle los tratamientos.
- ✓ Se necesita solicitar consulta para que el médico fisioterapeuta del centro de Rehabilitación pueda darle de alta de los tratamientos que le orientó.
- ✓ Para poder dar el alta, el paciente tiene que haber terminado de aplicarse todos los tratamientos orientados por el médico fisioterapeuta del centro de rehabilitación.
- ✓ Si el paciente falta a 3 sesiones del tratamiento consecutivamente, se le eliminan los turnos restantes.

### 2.5 Objeto de Automatización

Para minimizar los problemas existentes en la aplicación web “Sistema Automatizado para la Gestión de la Información en Rehabilitación” se propone la creación una nueva versión que garantice la conexión entre las diferentes clínicas de rehabilitación, el acceso a la información desde una determinada área (municipio, provincia, nación) y utilice un estándar de clasificación de enfermedades. Para lograr esto se hará uso de los servicios que brinda el Sistema de Información para la Salud (SiSalud) que permitirá integrar la información de las diferentes áreas de manera sencilla.

## Capítulo 2. Características del Sistema

---

- ✓ Registro del Personal de Salud (RPS): se encuentra información detallada del personal que trabaja para la salud en el país, dígame médicos, enfermeras, estadísticos, en general todo el personal médico y no médico.
- ✓ Sistema de Autenticación, Autorización y Auditoría (SAAA): módulo que se encarga de la gestión de la seguridad de todo el sistema y de los demás módulos y brinda servicios de autenticación, autorización y auditoría.
- ✓ Registro de Ciudadanos (RC): a través de este sistema se canaliza y gestiona todo el flujo de la información, ya sea de creación o modificación de los datos de los ciudadanos.
- ✓ Registro de la Clasificación Internacional de Enfermedades y Problemas Relacionados con la Salud (RCIE): listado de enfermedades y problemas de salud registrados en la Clasificación Estadística Internacional de Enfermedades y Problemas Relacionados con la Salud en su Décima Revisión (CIE-10), además de dar la posibilidad de gestionar o visualizar la información de los Capítulos, Grupos, Categorías, Subcategorías y Problemas de Salud.

Este sistema permite adoptar un lenguaje unificado y estandarizado para la descripción precisa de las enfermedades, causas de muerte u otros trastornos o condiciones de salud

### 2.6 Propuesta de Modelo de Negocio

Los sistemas por muy pequeños que sean generalmente son complejos, por lo que se hace necesario dividirlo en piezas para su mejor comprensión y para reducir su complejidad. Se realizó un estudio detallado de cada uno de los procesos que ocurren en estos centros de rehabilitación, analizando críticamente la ejecución y las actividades que se desarrollan. A partir de las necesidades que plantean el cliente, los problemas que se presentan y las deficiencias que existen, se ha realizado una propuesta de negocio.

#### 2.6.1 Actores del Negocio

Un actor del negocio representa un individuo, grupo, entidad, organización, máquina o sistema de información externos con los que interactúa el negocio [21]. Lo que se modela como actor es el rol que se juega cuando se interactúa con el negocio para beneficiarse de sus resultados.

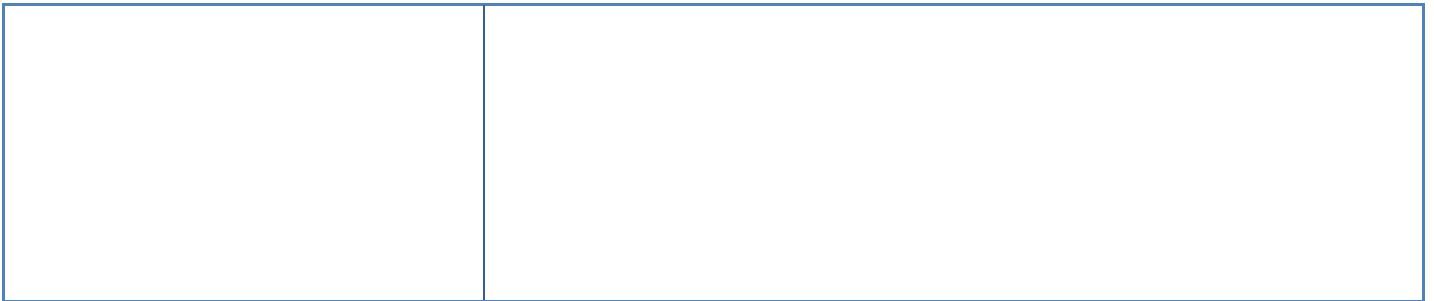
## Capítulo 2. Características del Sistema

Actores del Negocio	Descripción
<b>Paciente</b>	El paciente es el principal beneficiado con los resultados del negocio. El más interesado en recibir un servicio con calidad.

### 2.6.2 Trabajadores del Negocio

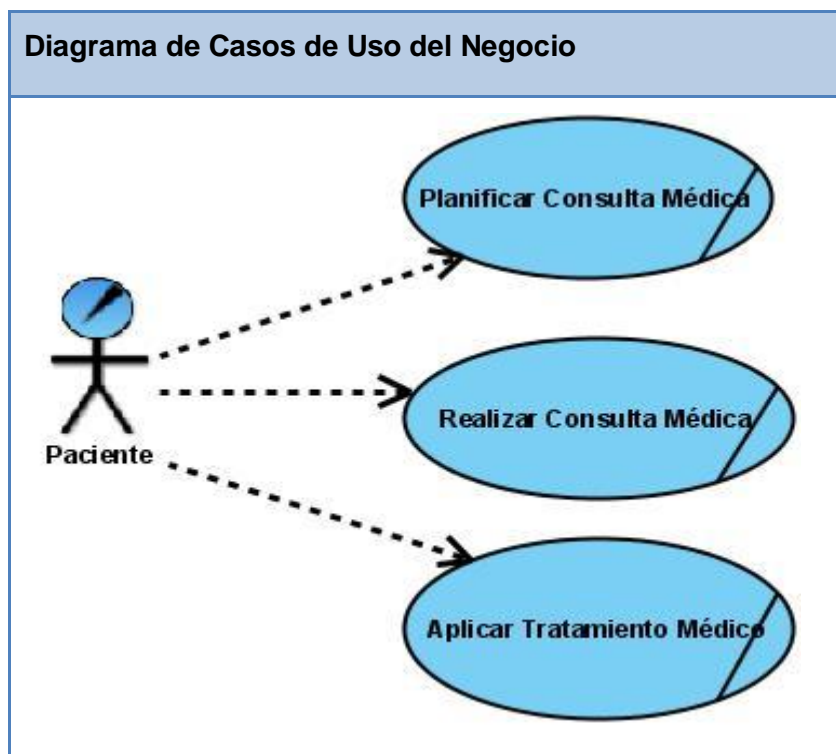
Un trabajador (rol), define un comportamiento o responsabilidades de un individuo o grupo de individuos trabajando en equipo, en el contexto de una organización de ingeniería de software. [22]

Trabajadores del Negocio	Descripción
<b>Recepcionista</b>	Es la encargada de registrar al paciente, comprobar si existe su historia clínica en el hospital y si no la tiene, le toma los datos para confeccionarle una. Se encargará de planificarle los turnos y los tratamientos. Entrega diariamente al servicio el modelo registro de pacientes atendidos en el Centro Integral de Rehabilitación.
<b>Médico</b>	Es el encargado de atender al paciente, diagnosticar la enfermedad y orientar los tratamientos en dependencia del diagnóstico. Es el único que puede dar el alta. Planifica las acciones de salud. Habilita y actualiza sistemáticamente el Registro de Pacientes en Rehabilitación.
<b>Técnico</b>	Se encarga de aplicar al paciente los tratamientos orientados por el especialista, y tiene la obligación de informar sobre la evolución del mismo. Habilita y entrega diariamente el(los) modelo(s) de Registro de Pacientes atendidos en Centro Integral de Rehabilitación al Departamento de Estadística.



### 2.6.3 Diagrama de Casos de Uso del Negocio

Un diagrama de casos de uso del negocio representa gráficamente a los procesos del negocio y su interacción con los actores del negocio.



## Capítulo 2. Características del Sistema

---

### 2.6.4 Especificación de los Casos de Uso

#### Caso de Uso “Solicitar Consulta Médica”

<b>Caso de Uso:</b>	<b>Planificar Consulta Médica</b>
<b>Actores:</b>	Paciente
<b>Trabajadores:</b>	Recepcionista
<b>Resumen:</b>	El caso de uso se inicia cuando el paciente llega a la sala de rehabilitación solicitando atención médica, este es atendido por la recepcionista, la cual le toma los datos necesarios, si tiene historia clínica se la actualiza y si no le hace una, además de crearle el turno médico para que sea atendido por el especialista.

#### Caso de Uso “Realizar Consulta Médica”

<b>Caso de Uso:</b>	<b>Realizar Consulta Médica</b>
<b>Actores:</b>	Paciente
<b>Trabajadores:</b>	Médico
<b>Resumen:</b>	El caso de uso se inicia cuando el paciente entra a la sala donde va a ser atendido por el especialista, este revisará los datos de la historia clínica y le hará previamente un diagnóstico para emitir una evaluación y orientarle los tratamientos.

## Capítulo 2. Características del Sistema

---

### Caso de Uso “Aplicar Tratamientos Médicos”

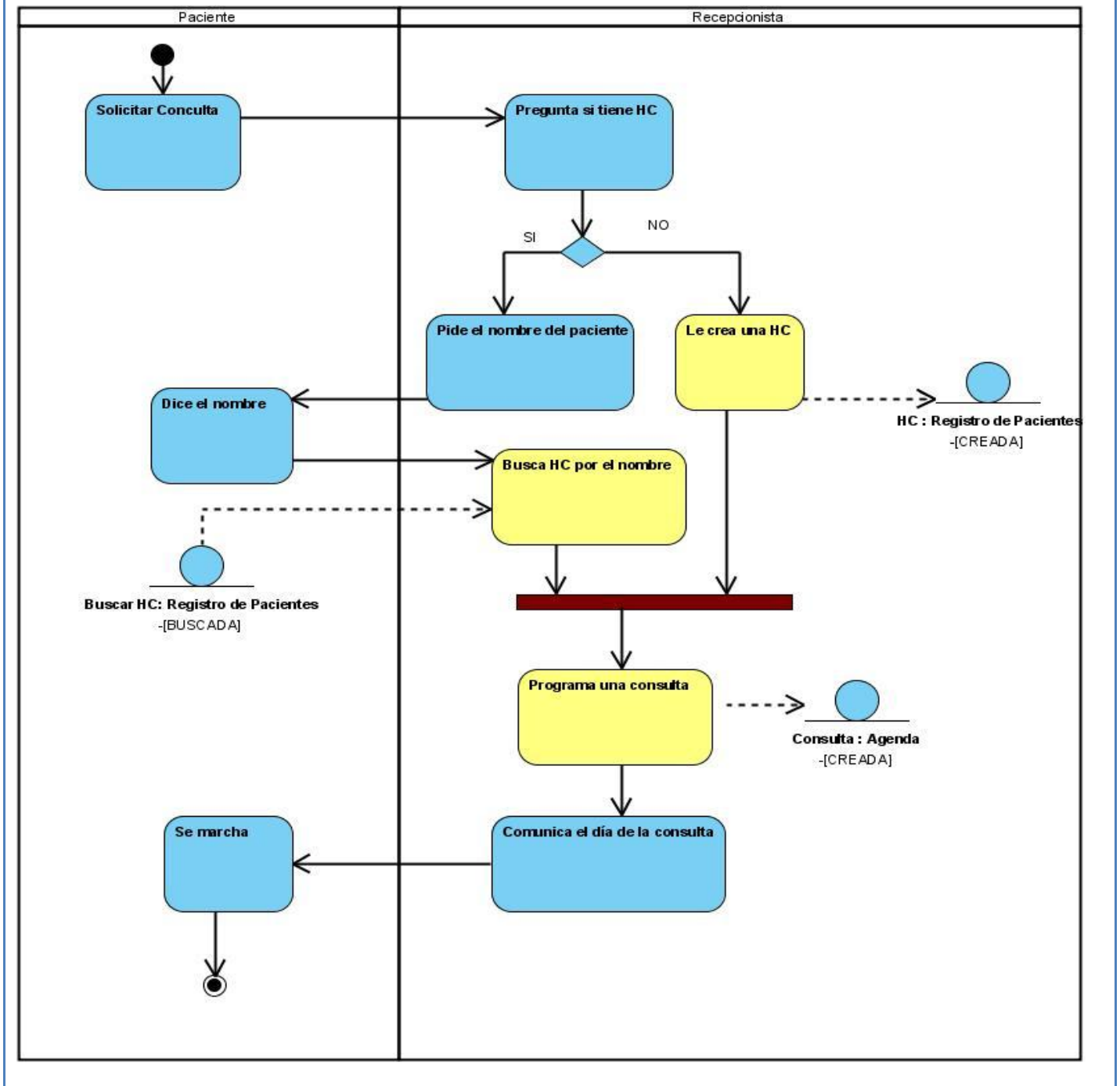
<b>Caso de Uso:</b>	<b>Aplicar Tratamientos Médicos</b>
<b>Actores:</b>	Paciente
<b>Trabajadores:</b>	Técnico
<b>Resumen:</b>	El caso de uso se inicia cuando el paciente entra a la sala donde va a ser atendido por los técnicos los cuales le aplicarán el o los tratamientos médicos indicados por el especialista que lo atendió. Estos le informarán al especialista sobre la evolución del paciente en la sala. Se evaluará el progreso de los tratamientos aplicados, para luego determinar si se le debe dar de alta al paciente o cambiarle el tratamiento.

### 2.6.5 Diagrama de Actividades

Los casos de uso del negocio consisten en la descripción de la secuencias de actividades que, en conjunto, producen algo observable para el actor del negocio. El proceso consiste en un flujo básico de una o más alternativas. La estructura del mismo se describe gráficamente con la ayuda de un diagrama de actividad. [23]

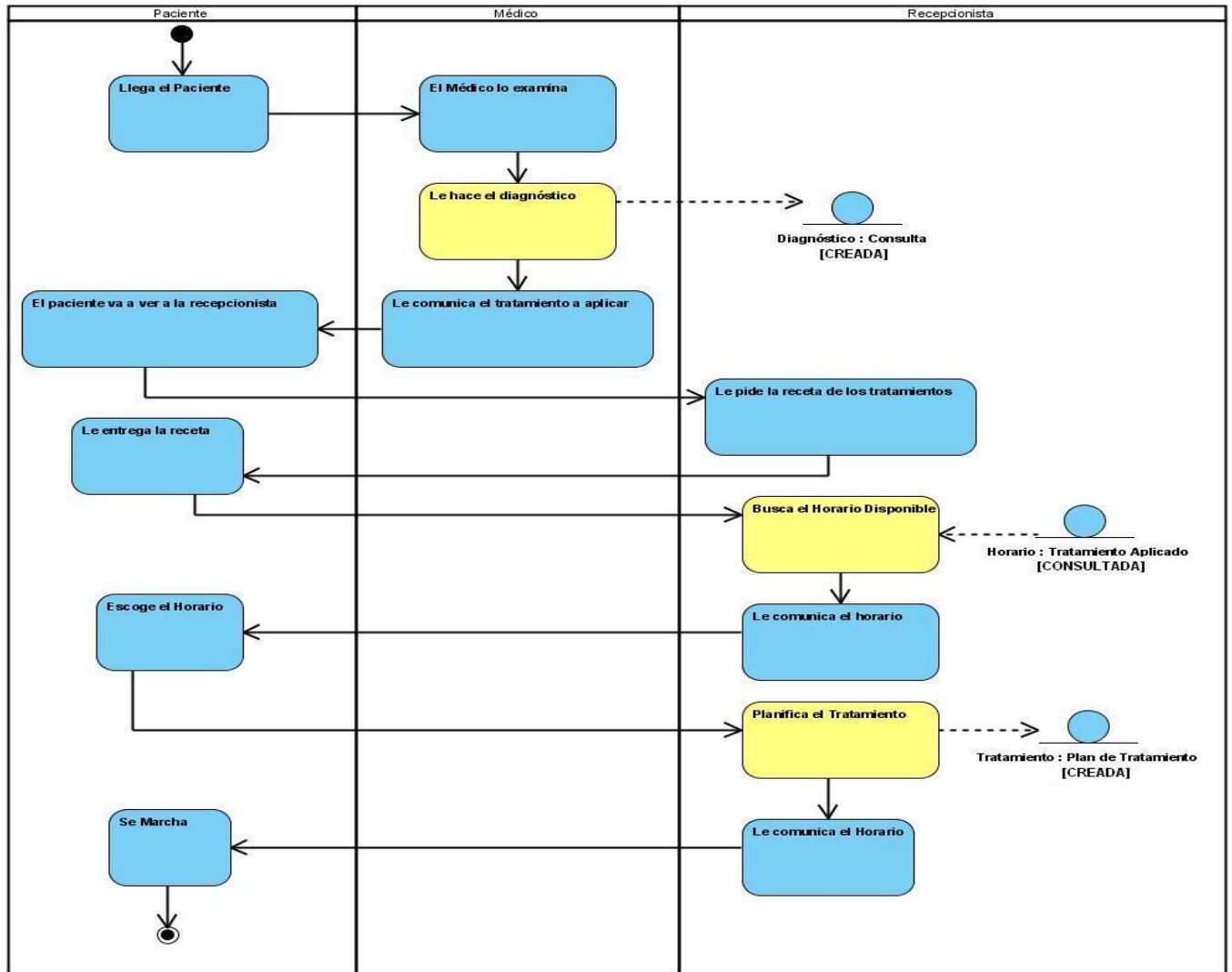
## Capítulo 2. Características del Sistema

Diagrama de Actividades - Caso de Uso Planificar Consulta Médica



## Capítulo 2. Características del Sistema

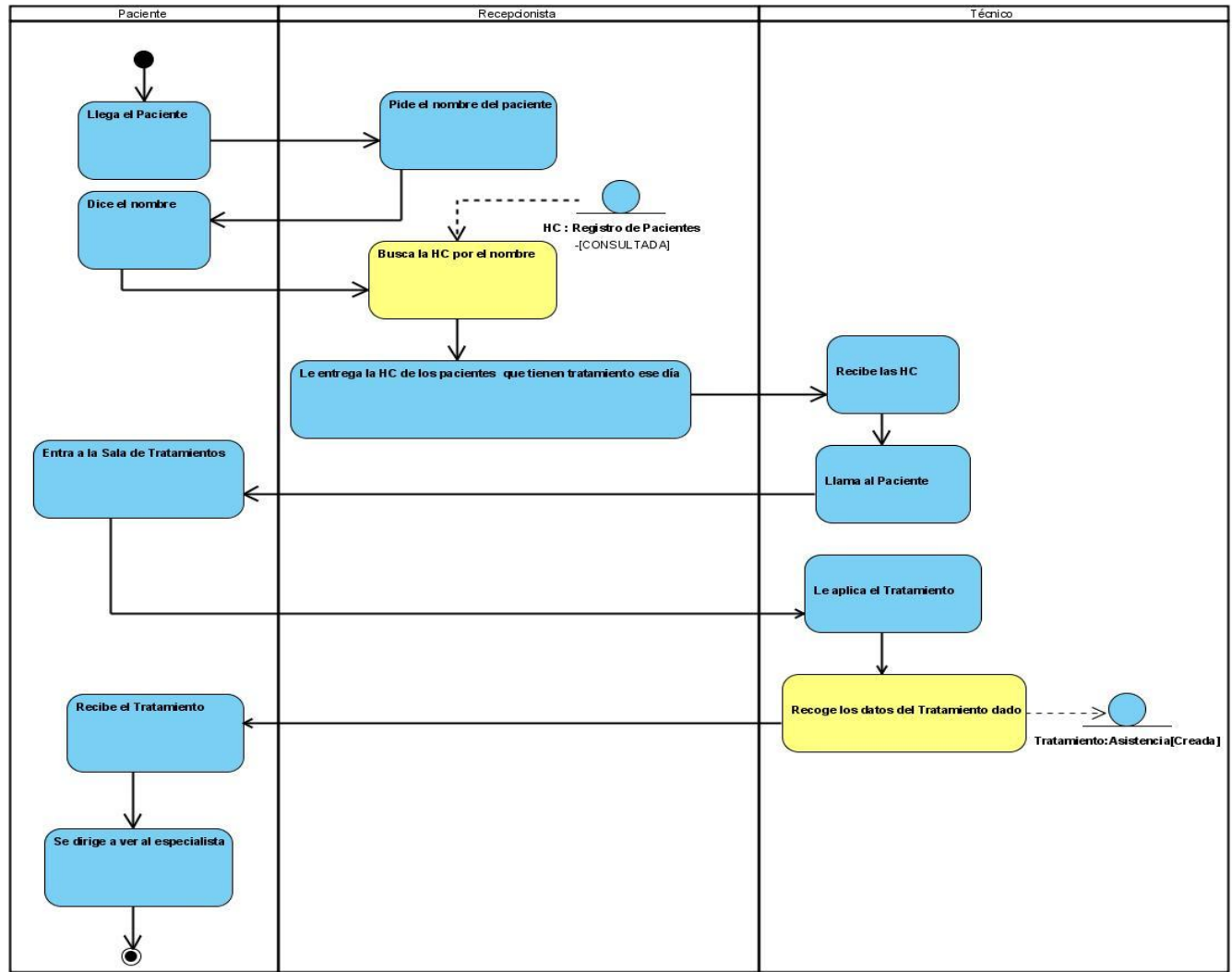
Diagrama de Actividades - Caso de Uso Realizar Consulta Médica





## Capítulo 2. Características del Sistema

Diagrama de Actividades - Caso de Uso Aplicar Tratamiento Médico



### 2.7 Especificación de los requisitos de software.

Los requerimientos de software son condiciones o capacidades que debe tener el sistema para satisfacer las necesidades de un cliente, estos deben ser especificados por escrito.

### 2.7.1 Requerimientos Funcionales.

Los requerimientos funcionales son aquellos requisitos que, desde el punto de vista de las necesidades del usuario, debe cumplir el sistema y que están fuertemente ligados a las opciones del programa.

RF1 - Autenticar Usuario

RF2 - Registrar Personal

RF3 - Buscar Pacientes

RF4 - Planificar Consulta

RF5 - Realizar Consulta

RF6 - Planificar Tratamientos

RF7 - Interrumpir Tratamiento

RF8 - Dar Asistencia a Tratamiento

RF9 - Gestionar Egreso

RF10-Buscar Ciudadano.

### 2.7.2 Requerimientos No Funcionales.

Los requerimientos no funcionales son características que describen alguna forma o restricción para la realización de algún requerimiento (funcionalidad) o conjunto de ellas e inclusive todos los requerimientos. Se consideran los atributos del sistema, propiedades que debe tener el producto. A continuación se muestran los requerimientos no funcionales:

#### **Fiabilidad**

La herramienta de implementación a utilizar tiene soporte para recuperación ante fallos y errores.

#### **Seguridad**

Utilización de un mecanismo de seguridad basado en el Sistema de Autenticación, Autorización y Auditoría (SAAA).

## Capítulo 2. Características del Sistema

---

### **Integridad**

El sistema se encargará de controlar los diferentes niveles de acceso en sitio, se identificará al usuario antes de que pueda realizar cualquier acción sobre el sistema. Se realizarán validaciones de la información tanto en el cliente como en el servidor, no obstante los usuarios acceden de manera rápida y operativa al sistema sin que los requerimientos de seguridad se conviertan en un retardo para ellos.

### **Soporte**

Se requiere que el producto reciba mantenimiento ante cualquier fallo que ocurra y que sea de fácil instalación.

### **Diseño**

Aplicación web se desarrollará con tecnología para creación de páginas web dinámicas PHP5 y con gestor de base de datos PostgreSQL.

### **Requisito no funcional para la documentación de usuarios en línea y ayuda del sistema.**

El sistema brindará a los usuarios una buena ayuda en línea, de modo que si el usuario presenta algún problema pueda acudir al mismo, así como una documentación apropiada para el mejor trabajo con el mismo.

### **Interfaz**

#### **Interfaces de usuario**

La interfaz no contendrá muchas imágenes para no demorar las respuestas al usuario. El diseño de la interfaz es sencillo y claro de usar con reconocimiento visual a través de elementos visibles que identifiquen cada una de sus acciones. Es formal, serio y con una navegación sugerente, todo esto teniendo en cuenta el fin con el que se desarrolla la aplicación.

#### **Interfaces Hardware**

Se necesitarán como requerimientos mínimos una PC con procesador Pentium II o superior, una memoria RAM de 128 MB o superior, disco duro de 20 GB o más, admite cualquier velocidad de red.

## Capítulo 2. Características del Sistema

---

### Interfaces Software

Para el funcionamiento del sistema en el servidor es necesario el Sistema Operativo Windows o Linux, Windows 98 o superior, y para Linux alguna distribución del Debían, admite cualquier navegador, preferentemente Mozilla Firefox en cualquiera de sus versiones

### Restricciones

Se utilizará UML para lograr una mejor documentación del sistema y como herramienta de apoyo Visual Paradigm, como lenguaje de programación PHP5 y gestor de base de datos PostgreSQL.

### Requisitos Legales, de Derecho de Autor y otros.

El sistema se basa en un estándar que se rige por normas internacionales y cumple con las normas y leyes establecidas en Cuba.

### Portabilidad

No es necesario especificar ningún sistema operativo para trabajar con la aplicación, ya que este podrá ser accesible desde cualquiera. El producto correrá sobre una plataforma web, codificada en PHP5, sus sistemas de gestor de bases de datos PostgreSQL y consume varios Servicios Web.

## 2.8 Definición de los casos de uso.

### 2.8.1 Definición de los actores.

Actor	Descripción
Usuarios	Va a ser todo aquel personal que va a tener algún rol en el sistema, es decir, los que van a poder trabajar con la aplicación.
Jefe de Departamento	Es el encargado de realizar las configuraciones del sistema, tiene acceso a todas las funcionalidades del sistema. En general va a ser el administrador de la aplicación.

## Capítulo 2. Características del Sistema

---

<b>Recepcionista</b>	Es la encargada de planificar las consultas, los tratamientos, la doble sesión de los tratamientos a los pacientes.
<b>Técnico</b>	Son los encargados de registrar en el sistema las asistencias de los pacientes al tratamiento, así como emitir la evolución de un paciente en el tratamiento.
<b>Médico</b>	Es el encargado de consultar al paciente, diagnosticarlo y ponerle un tratamiento en dependencia de la enfermedad. Es el único que puede darle de alta a un paciente.
<b>RCIE</b>	Este rol representa el intercambio de información con el sistema externo Registro de la Clasificación Internacional de Enfermedades.
<b>RC</b>	Este rol representa el intercambio de información con el sistema externo Registro de Ciudadanos.
<b>SAAA</b>	Este rol representa el intercambio de información con el sistema externo Sistema Autenticación, Autorización y Auditoría
<b>RPS</b>	Este rol representa el intercambio de información con el sistema externo Registro de Personal de la Salud.

## Capítulo 2. Características del Sistema

---

### 2.8.2 Listado de Casos de Usos (CU)

<b>CU - 1</b>	Autenticar Usuario
<b>Actor</b>	Usuario (Recepcionista, Técnico, Médico, Jefe de Departamento)
<b>Descripción</b>	Permite que solamente puedan acceder a la aplicación todos aquellos que trabajarán directamente con la misma, dándole acceso a operar en el sistema, según las funcionalidades a las que tiene permiso.
<b>Referencia</b>	<b>RF1</b>

<b>CU - 2</b>	Registrar Personal
<b>Actor</b>	Jefe de Departamento
<b>Descripción</b>	Permitirá registrar un personal médico en la clínica, el administrador del sistema entrará el usuario y contraseña del médico, luego se confirmará a través del servicio SAAA, obteniéndose los datos del médico a través del servicio RC.
<b>Referencia</b>	<b>RF2</b>

<b>CU - 3</b>	Buscar Paciente
<b>Actor</b>	Usuario (Recepcionista, Técnico, Médico, Jefe de Departamento)
<b>Descripción</b>	Se permitirá buscar a un paciente y visualizar sus datos. Se buscarán los pacientes que se requieran, ya sea para planificarle un

## Capítulo 2. Características del Sistema

---

	tratamiento, realizarle una consulta, darle asistencia a un tratamiento, interrumpirle un tratamiento o darle de alta.
<b>Referencia</b>	<b>RF3</b>

<b>CU - 4</b>	Planificar Consulta
<b>Actor</b>	Recepcionista
<b>Descripción</b>	Una vez que el paciente llegue al centro de rehabilitación solicitando ser atendido por un médico la recepcionista buscará los datos de mismo en el Registro de Ciudadanos (RC) y procederá a planificarle la consulta.
<b>Referencia</b>	<b>RF4</b>

<b>CU - 5</b>	Realizar Consulta
<b>Actor</b>	Médico, Jefe de Departamento
<b>Descripción</b>	El sistema mostrará un listado de los pacientes planificados para ese día y el médico seleccionará el paciente y entrará los datos para la realización de la consulta así como una descripción de los tratamientos.
<b>Referencia</b>	<b>RF5</b>

## Capítulo 2. Características del Sistema

---

<b>CU - 6</b>	Planificar Tratamientos
<b>Actor</b>	Recepcionista
<b>Descripción</b>	La planificación de tratamientos se realizará después que el paciente haya sido atendido por el médico en la consulta. El sistema mostrará un listado de los tratamientos del paciente y la recepcionista accederá a planificar el tratamiento verificando la disponibilidad en el día.
<b>Referencia</b>	<b>RF6</b>

<b>CU - 7</b>	Interrumpir Tratamiento
<b>Actor</b>	Médico, Jefe de Departamento.
<b>Descripción</b>	Permitirá a un especialista interrumpirle un tratamiento a un paciente, por dos razones fundamentales:  - El paciente se encuentra restablecido y no necesita más sesiones de ese tratamiento.  - El tratamiento está empeorando la situación del paciente, por lo que es necesario cambiarle el tratamiento.
<b>Referencia</b>	<b>RF7</b>

<b>CU - 8</b>	Dar Asistencia al Tratamiento
<b>Actor</b>	Técnico



## Capítulo 2. Características del Sistema

---

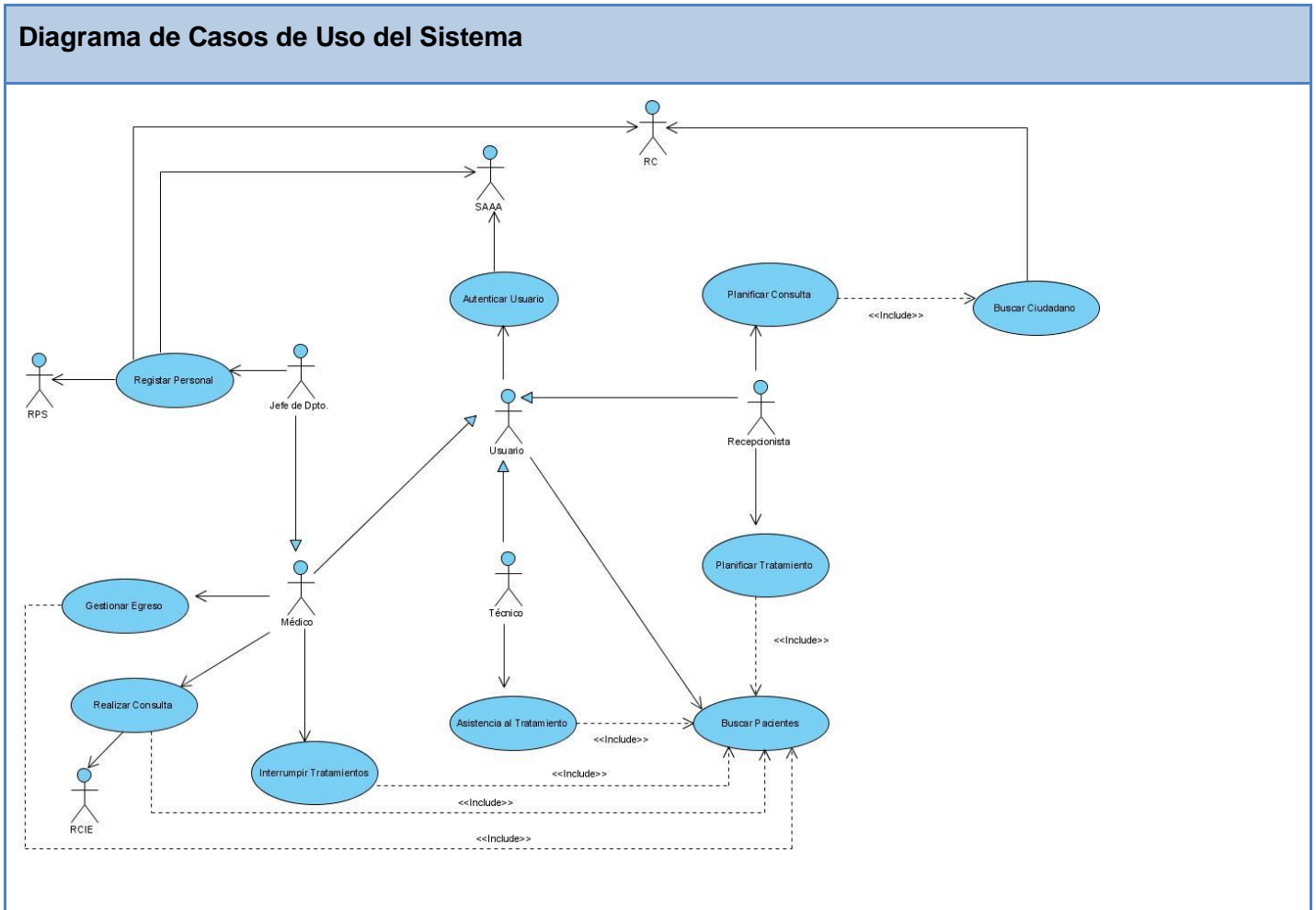
<b>Descripción</b>	El técnico debe tomar la asistencia diaria del paciente y dar una evaluación de la evolución del mismo.
<b>Referencia</b>	<b>RF8</b>

<b>CU - 9</b>	Gestionar Egreso
<b>Actor</b>	Médico, Jefe de Departamento
<b>Descripción</b>	El egreso consiste en darle de baja a un paciente, es decir, cuando el paciente haya terminado con las sesiones de los tratamientos que se le han orientado, formará parte del listado de egresados de la clínica automáticamente, cuando se encuentre en este listado el especialista determinará si el paciente ya se encuentra listo para ser egresado.
<b>Referencia</b>	<b>RF 9</b>

<b>CU - 10</b>	Buscar Ciudadanos
<b>Actor</b>	Recepcionista
<b>Descripción</b>	Cuando un paciente solicite ser atendido por un médico se buscará en el RC para obtener sus datos y planificarle la consulta.
<b>Referencia</b>	<b>RF10</b>

## Capítulo 2. Características del Sistema

### 2.8.3 Diagrama de casos de uso del sistema.



### 2.8.4 Casos de uso por ciclo.

Código	Caso de Uso	Ciclo	Descripción
CU-1	Autenticar Usuario	1	Se decide desarrollar en el primer ciclo debido a la importancia que tiene el manejo de la seguridad del sistema con el objetivo de evitar la pérdida o alteración de información por

## Capítulo 2. Características del Sistema

---

			personal no autorizado al mismo.
CU-2	Registrar Personal	1	Se decide desarrollar en el primer ciclo pues constituye una prioridad inmediata en la aplicación.
CU-3	Buscar Pacientes		Se decide desarrollar en el primer ciclo por ser una actividad fundamental a la hora de atender a los pacientes.
CU-4	Planificar Consulta	1	Se decide desarrollar en el primer ciclo por ser una actividad fundamental a la hora de atender a los pacientes.
CU-5	Realizar Consulta	1	Se decide desarrollar en el primer ciclo ya que otros casos de usos dependen de este para poderse realizar, además de constituir una prioridad para el sistema.
CU-6	Planificar Tratamiento	1	Se decide desarrollar en el primer ciclo por ser una actividad fundamental a la hora de atender a los pacientes.
CU-7	Interrumpir Tratamiento	1	Se decide desarrollar en el primer ciclo por ser una actividad fundamental a la hora de atender a los pacientes.
CU-8	Dar Asistencia a Tratamiento	1	Se decide desarrollar en el primer ciclo pues constituye una prioridad inmediata en la aplicación.

## Capítulo 2. Características del Sistema

CU-9	Gestionar Egreso	1	Se decide desarrollar en el primer ciclo pues constituye una prioridad inmediata en la aplicación.
CU-10	Buscar Ciudadano	1	Se decide desarrollar en el primer ciclo por ser una actividad fundamental a la hora de planificar una consulta a un paciente.

### 2.8.5 Casos de uso expandidos.

#### Descripción de Casos de Uso Autenticar Usuario

<b>Caso de Uso:</b>	Autenticar Usuario		
<b>Actores:</b>	Usuario (que puede ser Recepcionista, Técnico, Médico, Jefe de Departamento)		
<b>Resumen:</b>	Permite que solamente puedan acceder a la aplicación todos aquellos que trabajarán directamente con la misma, dándole acceso a operar en el sistema, según los vínculos a los que tienen permiso.		
<b>Precondiciones:</b>	El servicio brindado por SiSalud (SAAA) debe estar disponible y actualizado.		
<b>Referencias</b>	RF1		
<b>Prioridad</b>	Importante		
<b>Flujo Normal de Eventos</b>			
<b>Sección “Autenticar Usuario”</b>			
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>		
1. El usuario entra usuario y contraseña.	2. El sistema comprueba si los datos		

## Capítulo 2. Características del Sistema

	entrados son o no correctos.
	3. Si los datos son correctos el sistema muestra una página de bienvenida, con todas las funcionalidades a las que tiene acceso el usuario.
<b>Prototipo de Interfaz</b>	
<b>Flujo Alternativo de Eventos</b>	
<b>Sección “Autenticar Usuario”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	2.1 Si los datos son incorrectos el sistema muestra un mensaje de error <b>“Usuario o contraseña incorrecta”</b> .
<b>Poscondiciones</b>	El usuario queda autenticado en el sistema.

### Descripción de Casos de Uso Registrar Personal

<b>Caso de Uso:</b>	Registrar Personal
<b>Actores:</b>	Jefe de Departamento
<b>Resumen:</b>	Permitirá registrar un personal médico en la clínica, el médico entrará con su usuario y contraseña, luego en esto se confirmará a través del servicio SAAA, luego se obtendrán los datos del médico a través de los servicios RC y RPS.
<b>Precondiciones:</b>	La información de los diferentes registros que se relacionan con el proceso debe

## Capítulo 2. Características del Sistema

	estar disponible y actualizada.
<b>Referencias</b>	<b>RF2</b>
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Registrar Personal ”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El Jefe de Departamento accede la opción <i>Registrar Personal</i> .	2. El sistema muestra una pantalla para que entre usuario y contraseña.
3. El Jefe de Departamento introduce usuario y contraseña.	4. El sistema valida los datos.
	5. Si los datos son correctos el sistema realiza la búsqueda y muestra los resultados.
6. El Jefe de Departamento escoge el rol y selecciona la opción “Acción”.	7. El sistema registra los datos y termina el caso de uso.
<b>Prototipo de Interfaz</b>	
<b>Flujo Normal de Eventos</b>	
<b>Sección “Registrar Personal ”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	4.1 Si los datos son incorrectos el sistema muestra un mensaje de error “ <b>Acceso</b>

## Capítulo 2. Características del Sistema

	<b>Denegado”.</b>
	7.1 Si los datos entrados ya se encuentran en la base de datos el sistema muestra un mensaje de error <b>“Esta persona ya se encuentra registrada”.</b>
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar cualquier acción descrita con anterioridad.

### Descripción de Casos de Uso Buscar Pacientes

<b>Caso de Uso:</b>	Buscar Paciente
<b>Actores:</b>	Usuario (Recepcionista, Técnico, Médico, Jefe de Departamento)
<b>Resumen:</b>	Se permitirá buscar a un paciente y visualizar sus datos. Se buscarán los pacientes que se requieran, ya sea para planificarle un tratamiento, realizarle la consulta, gestionar un egreso, darle asistencia a algún tratamiento o interrumpirle un tratamiento.
<b>Precondiciones:</b>	La información de los diferentes registros que se relacionan con el proceso debe estar disponible y actualizada.
<b>Referencias</b>	<b>RF3</b>
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Buscar Pacientes”</b>	

## Capítulo 2. Características del Sistema

Acción del Actor	Respuesta del Sistema
1. El usuario accede a una de las opciones relacionadas con el <i>Buscar Paciente</i> .	2. El sistema muestra una pantalla para que el usuario entre el número de Carnet de Identidad o el Nombre del paciente.
3. El usuario introduce el número de Carnet de Identidad o el Nombre y da clic en el botón "Buscar".	4. El sistema valida los datos.
	5. El sistema realiza la búsqueda con los datos entrados, muestra los resultados de la búsqueda y termina el caso de uso.
<b>Prototipo de Interfaz</b>	
<b>Flujo Normal de Eventos</b>	
<b>Sección "Buscar Pacientes"</b>	
Acción del Actor	Respuesta del Sistema
	5.1 Si no existen pacientes en la base de datos el sistema muestra el mensaje " <b>No Existen Registros</b> ".
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar cualquier acción descrita con anterioridad.



## Capítulo 2. Características del Sistema

### Descripción de Casos de Uso Planificar Consulta

<b>Caso de Uso:</b>	Planificar Consulta	
<b>Actores:</b>	Recepcionista	
<b>Resumen:</b>	Una vez que el paciente llegue al centro de rehabilitación se busca en el Registro de Ciudadanos (RC) y se procederá a planificarle la consulta.	
<b>Precondiciones:</b>	La información del registro RC debe estar disponible y actualizada.	
<b>Referencias</b>	RF4	
<b>Prioridad</b>	Crítica	
<b>Flujo Normal de Eventos</b>		
<b>Sección “Planificar Consulta”</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. La Recepcionista accede a la opción “Planificar Consulta”	2. El sistema muestra una pantalla donde permite al usuario buscar los datos de los pacientes.	
3. La Recepcionista introduce el número de Carnet de Identidad o el Nombre y da clic en el botón “Buscar”.	4. El sistema valida los datos.	
	5. El sistema muestra una pantalla con los resultados de la búsqueda realizada.	
6. La Recepcionista selecciona al paciente dando clic en la opción de “Acción”.	7. El sistema muestra una pantalla para que se le planifique la consulta al paciente.	

## Capítulo 2. Características del Sistema

8. La Recepcionista entra los datos, selecciona la opción "Aceptar"	9. El sistema valida los datos.
	10. El sistema registra los datos y termina el caso de uso.
<b>Prototipo de Interfaz</b>	
<b>Flujo Alternativo de Eventos</b>	
<b>Sección "Planificar Consulta"</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	5.1 Si los datos son incorrectos el sistema muestra un mensaje de error " <b>No existen registros</b> ".
8.1 Si la recepcionista desea cancelar la planificación selecciona la opción "Cancelar"	
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar cualquier acción descrita con anterioridad.

### Descripción de Casos de Uso Realizar Consulta

<b>Caso de Uso:</b>	Realizar Consulta
<b>Actores:</b>	Médico, Jefe de Departamento
<b>Resumen:</b>	Se entrarán los datos que requiera el médico para llegar a un diagnóstico final para orientarle los tratamientos al paciente.
<b>Precondiciones:</b>	El paciente tiene que estar registrado en el sistema y la información de los

## Capítulo 2. Características del Sistema

	diferentes registros que se relacionan con el proceso debe estar disponible y actualizada.
<b>Referencias</b>	RF5
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Realizar Consulta”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El médico accede a la opción “ <i>Realizar Consulta</i> ”.	2. El sistema muestra una pantalla para que el Médico entre el número de Carnet de Identidad o el Nombre del paciente, así como muestra un listado de los pacientes que tienen consulta para ese día.
3. Selecciona al paciente haciendo clic en el botón “Acción”.	4. El sistema muestra una pantalla para que el médico entre los datos que se obtengan de la consulta realizada al paciente, y le orienta los tratamientos que se debe aplicar.
5. El médico entra los datos y da clic en el botón “Aceptar”.	6. El sistema valida los datos.
	7. El sistema muestra una pantalla para que se entre una descripción del (los) tratamiento(s) que se orientó al paciente si es necesario.

## Capítulo 2. Características del Sistema

8. El Médico Fisioterapeuta entra la descripción del tratamiento y da clic en el botón “Enviar”	9. El sistema registra los datos enviados y termina el caso de uso.
<b>Prototipo de Interfaz</b>	
<b>Flujo Alternativo de Eventos</b>	
<b>Sección “Realizar Consulta”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	2.1 Si no existe ningún paciente con consulta para ese día, muestra un mensaje: <b>“No existen registros”</b>
5.1 El médico da clic en el botón “Cancelar”, cancelando la consulta que le realizó al paciente.	6.1 Si los datos entrados son incompletos el sistema muestra un mensaje de error <b>“Todos los campos son obligatorios”</b>
<b>Prototipo de Interfaz</b>	
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar las acciones descritas.

### Descripción de Casos de Uso Planificar Tratamiento

<b>Caso de Uso:</b>	Planificar Tratamientos
<b>Actores:</b>	Recepcionista
<b>Resumen:</b>	La planificación de tratamientos se realizará después que el paciente haya sido atendido por el médico en la consulta. El primer paso consistirá en observar la disponibilidad para el tratamiento, es decir verificar que exista disponibilidad el día

## Capítulo 2. Características del Sistema

	que se va a planificar el tratamiento.
<b>Precondiciones:</b>	El paciente tiene que haber sido consultado por un médico con anterioridad.
<b>Referencias</b>	<b>RF6</b>
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Planificar Tratamiento”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. La recepcionista accede a la opción “ <i>Planificar Tratamiento</i> ”.	2. El sistema muestra una pantalla para que la recepcionista busque al paciente que desea planificarle el tratamiento.
3. La recepcionista inserta el número de Carnet de Identidad o el Nombre del paciente.	4. El sistema muestra una pantalla con un listado de todos los pacientes que ya han recibido consulta médica y se les orientó un tratamiento médico.
5. La recepcionista selecciona un paciente dando clic en la opción “Acción”.	6. El sistema muestra una pantalla con el listado de tratamientos que se le orientaron al paciente, donde tiene la opción “Planificar Tratamiento”.
7. La recepcionista selecciona la opción “Planificar Tratamiento”.	8. El sistema muestra una pantalla para que se entren los datos necesarios para planificar un tratamiento.
9. La recepcionista entra los datos que se requieren para planificar un tratamiento y	10. El sistema valida los datos

## Capítulo 2. Características del Sistema

selecciona la hora de la planificación.	
	11. El sistema registra los datos y termina el caso de uso.
<b>Flujo Alternativo de Eventos</b>	
<b>Sección “Planificar Tratamiento”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	4.1 Si no existe ningún paciente con consulta para ese día, muestra un mensaje: <b>“No existen registros”</b>
9.1 Si la hora de la planificación ya está ocupada saldrá un mensaje <b>“Este técnico/equipo está ocupado para el día aaaa-mm-dd a las hh:mm:ss”</b>	
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar las acciones descritas anteriormente.

### Descripción de Casos de Uso Interrumpir Tratamiento

<b>Caso de Uso:</b>	Interrumpir Tratamiento
<b>Actores:</b>	Médico, Jefe de Departamento
<b>Resumen:</b>	Permitirá a un especialista interrumpirle un tratamiento a un paciente, por dos razones fundamentales:  - El paciente se encuentra restablecido y no necesita más sesiones de ese tratamiento.

## Capítulo 2. Características del Sistema

	- El tratamiento está empeorando la situación del paciente, por lo que es necesario cambiarle el tratamiento.
<b>Precondiciones:</b>	El paciente tiene que tener planificado al menos un tratamiento.
<b>Referencias</b>	<b>RF7</b>
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Interrumpir Tratamiento”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El Especialista accede a la opción “ <i>Interrumpir Tratamiento</i> ”.	2. El sistema muestra una pantalla para que el especialista busque al paciente que desea interrumpirle el tratamiento.
3. El especialista inserta el número de Carnet de Identidad o el nombre del paciente.	4. El sistema muestra una pantalla con el resultado de la búsqueda.
5. El especialista selecciona el paciente al que le desea interrumpir el tratamiento dando clic a la opción “Acción”.	6. El sistema muestra una pantalla con el listado de tratamientos que se le orientaron al paciente.
7. El especialista selecciona el tratamiento que desea interrumpir.	8. El sistema muestra una pantalla con los datos a entrar para poder interrumpir un tratamiento.
9. El especialista entra los datos, en este caso pondría si el paciente se encuentra rehabilitado	10. El sistema valida los datos.

## Capítulo 2. Características del Sistema

o no y las causas y selecciona la opción “Aceptar”.	
	11.El sistema registra los nuevos datos y termina el caso de uso.
<b>Prototipo de Interfaz</b>	
<b>Flujo Alternativo de Eventos</b>	
<b>Sección “Interrumpir Tratamiento”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
	4.1 Si no existe ningún paciente con consulta para ese día, muestra un mensaje: <b>“No existen registros”</b>
9.1 En caso que no desee ya interrumpir el tratamiento puede dar clic en el botón “Cancelar”	
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar las acciones descritas anteriormente.

### Descripción de Casos de Uso Dar Asistencia a Tratamiento

<b>Caso de Uso:</b>	Dar Asistencia a Tratamiento
<b>Actores:</b>	Técnico
<b>Resumen:</b>	El técnico debe tomar la asistencia diaria del paciente y dar una evaluación de la



## Capítulo 2. Características del Sistema

	evolución del mismo.
<b>Precondiciones:</b>	El paciente tiene que tener al menos un tratamiento planificado.
<b>Referencias</b>	<b>RF8</b>
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Asistencia a Tratamiento”</b>	
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>
1. El técnico accede a la opción <i>“Dar Asistencia a Tratamiento”</i> .	2. El sistema muestra una pantalla para que seleccione el tratamiento que desea buscar para verificar la asistencia de un paciente.
3. El técnico escoge el tratamiento y da clic en el botón “Buscar”.	4. El sistema muestra en pantalla un listado con los pacientes que tienen planificados tratamientos para ese día.
5. El técnico selecciona al paciente que quiere darle la asistencia al tratamiento dando clic en la opción “Acción”.	6. El sistema muestra una pantalla para que entre los datos que se requieren para darle la asistencia de un tratamiento a un paciente.
7. El técnico entra los datos y da clic en el botón “Enviar”	8. El sistema valida los datos.
	9. El sistema registra los datos y termina el caso de uso.

## Capítulo 2. Características del Sistema

<i>Prototipo de Interfaz</i>	
Flujo Alternativo de Eventos	
Sección "Asistencia a Tratamiento"	
Acción del Actor	Respuesta del Sistema
	4.1 Si no existe ningún paciente con consulta para ese día, muestra un mensaje: <b>"No existen registros"</b>
7.1 En caso que desee cancelar la acción que está realizando puede dar clic en el botón "Cancelar"	8.1 Si los datos entrados son incorrectos el sistema muestra un mensaje de error <b>"Error en los datos"</b>
Poscondiciones	El sistema queda actualizado luego de realizar las acciones descritas anteriormente.

### Descripción de Casos de Uso Gestionar Egreso

<b>Caso de Uso:</b>	Gestionar Egreso
<b>Actores:</b>	Médico, Jefe de Departamento
<b>Resumen:</b>	Egreso consiste en darle de baja a un paciente, es decir, cuando el paciente haya terminado con las sesiones de los tratamientos que se le han orientado, formará parte del listado de egresados de la clínica automáticamente, cuando se encuentre en este listado se brindarán dos opciones fundamentales al especialista, la primera teclear la "Evolución Final" del paciente y darle de "Egreso o Baja" al paciente.

## Capítulo 2. Características del Sistema

<b>Precondiciones:</b>	El paciente tiene que haber terminado todas las sesiones del tratamiento	
<b>Referencias</b>	RF9	
<b>Prioridad</b>	Crítica	
<b>Flujo Normal de Eventos</b>		
<b>Sección “Gestionar Egreso”</b>		
<b>Acción del Actor</b>	<b>Respuesta del Sistema</b>	
1. El médico o Jefe de Departamento accede a la opción “Egreso”.	2. El sistema muestra una pantalla con un listado de todos los pacientes que están listos para el egreso, para que el especialista decida o no darle el egreso.	
3. El médico o Jefe de Departamento selecciona al paciente que va a darle de alta, seleccionando la opción “Acción”.	4. El sistema muestra una pantalla para que el médico o Jefe de Departamento inserte la evolución final del paciente al terminar los tratamientos.	
5. El médico o Jefe de Departamento inserta los datos sobre la evolución final del paciente y da clic en el botón “Enviar”	6. El sistema valida los datos.	
	7. El sistema registra los datos y termina el caso de uso.	
<b>Prototipo de Interfaz</b>		
<b>Flujo Alternativo de Eventos</b>		

## Capítulo 2. Características del Sistema

Sección “Gestionar Egreso”	
Acción del Actor	Respuesta del Sistema
	2.1 Si no existe ningún paciente, el sistema muestra un mensaje: <b>“No existen registros”</b>
5.1 Si desea cancelar el egreso del paciente solo debe dar clic en el botón “Cancelar”.	
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar las acciones descritas anteriormente.

### Descripción de Casos de Uso Buscar Ciudadanos

<b>Caso de Uso:</b>	Buscar Ciudadanos
<b>Actores:</b>	Recepcionista
<b>Resumen:</b>	Se permitirá buscar a un ciudadano y visualizar sus datos. Se buscarán los pacientes que se requieran obteniendo la información del Registro de Ciudadanos (RC), para planificar una consulta.
<b>Precondiciones:</b>	La información de los diferentes registros que se relacionan con el proceso debe estar disponible y actualizada.
<b>Referencias</b>	<b>RF10</b>
<b>Prioridad</b>	Crítica
<b>Flujo Normal de Eventos</b>	
<b>Sección “Buscar Ciudadanos”</b>	

## Capítulo 2. Características del Sistema

Acción del Actor	Respuesta del Sistema
1. El usuario accede a la opción relacionada con el <i>Buscar "Ciudadanos"</i> .	2. El sistema muestra una pantalla para que el usuario entre el número de Carnet de Identidad, o el Nombre del ciudadano.
3. El usuario introduce el número de Carnet de Identidad o el Nombre y da clic en el botón "Buscar".	4. El sistema realiza la búsqueda con los datos entrados, muestra los resultados de la búsqueda y termina el Caso de Uso
<b>Prototipo de Interfaz</b>	
<b>Poscondiciones</b>	El sistema queda actualizado luego de realizar cualquier acción descrita con anterioridad.

En este capítulo se ha realizado un análisis del flujo de la información en los centros de rehabilitación del país. Se realizó la modelación del negocio identificándose: los actores, trabajadores y casos de uso del mismo, así como las reglas a tener en cuenta durante todo el proceso.

Se obtuvo un listado de los requisitos funcionales y no funcionales a cumplir durante el desarrollo de la aplicación. Además de dar una explicación de los casos de uso y los actores del sistema, dando una vista global de cómo estará concebida la aplicación.

### Capítulo 3. Análisis y Diseño

Cada paso durante el ciclo de desarrollo del proyecto cumple un importante papel para obtener el producto final, en este caso se desarrolla el Flujo de Trabajo de Análisis y Diseño. Se explicará la estructura del sistema, la arquitectura a utilizar, así como la representación gráfica de los diagramas de clases del análisis, de interacción y de clases del diseño con estereotipos web.

#### 3.1 Arquitectura

##### Modelo-Vista-Controlador (MVC)

Fue diseñada para reducir el esfuerzo de programación necesario en la implementación de sistemas múltiples y sincronizados de los mismos datos. Sus características principales son que el modelo, las vistas y los controladores se tratan como entidades separadas; esto hace que cualquier cambio producido en el modelo se refleje automáticamente en cada una de las vistas.

Este modelo de arquitectura presenta varias ventajas:

- ✓ Hay una clara separación entre los componentes de un programa; que permite implementarlos por separado.
- ✓ Hay un API (Interfaz de Programación de Aplicaciones) muy bien definido; cualquiera que use el API, podrá reemplazar el modelo, la vista o el controlador, sin aparente dificultad.
- ✓ La conexión entre el modelo y sus vistas es dinámica; se produce en tiempo de ejecución, no en tiempo de compilación.

Al incorporar el modelo de arquitectura MVC a un diseño, las piezas de un programa se pueden construir por separado y luego unir las en tiempo de ejecución. Si uno de los Componentes, posteriormente, se observa que funciona mal, puede reemplazarse sin que las otras piezas se vean afectadas. Este escenario contrasta con la aproximación monolítica típica de muchos programas Java. Todos tienen un Marco (Frame) que contiene todos los elementos, un controlador de eventos, y una gran cantidad de cálculos y la presentación del resultado. Ante esta perspectiva, hacer un cambio aquí no es nada trivial.

### Definición de las partes

El **Modelo** es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El modelo no tiene conocimiento específico de los controladores o de las vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el modelo y sus vistas, y notificar a las vistas cuando cambia el modelo.

La **Vista** es el objeto que maneja la presentación visual de los datos representados por el modelo. Genera una representación visual del modelo y muestra los datos al usuario. Interactúa con el modelo a través de una referencia al propio modelo.

El **Controlador** es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del modelo o por alteraciones de la vista. Interactúa con el modelo a través de una referencia al propio modelo.

### 3.2 Modelo de Análisis.

El modelo de análisis es un modelo de objetos conceptual que analiza los requisitos mediante su refinamiento y estructuración y permite razonar sobre los aspectos internos del sistema, incluidos sus recursos compartidos. Además ofrece un mayor poder expresivo, una mayor formalización y proporciona una estructura centrada en el mantenimiento de aspectos tales como la flexibilidad ante los cambios y la reutilización.

Esta estructura no solo es útil para el mantenimiento de los requisitos, sino también que se utiliza como entrada en las actividades de diseño e implementación. El principal artefacto que se genera durante este flujo de trabajo es la realización de los casos de uso del análisis que describen como se refinan estos en términos de colaboraciones dentro del modelo de análisis y además de cómo se ejecuta un caso de uso determinado en términos de las clases del análisis y de sus objetos.

### Diagrama de Clases del Análisis

El diagrama de clases del análisis se realiza para cada caso de uso del sistema y muestra las clases participantes, y sus relaciones. Es un artefacto en el que se representan los conceptos en un dominio del problema y representa las cosas del mundo real. En los diagramas de clases se identifican tres tipos de clases: interfaz, controladoras y entidades.

Las clases interfaz se utilizan para modelar la interacción entre el sistema y sus actores, es decir, usuarios y sistemas externos. Esta interacción a menudo implica recibir información y peticiones de los usuarios y sistemas externos. Cada clase de interfaz debe asociarse con al menos un actor.

Las clases de control representan coordinación, secuencia, transacciones y control de otros objetos, se usan con frecuencia para encapsular el control de un caso de uso en concreto. También se utilizan para representar derivaciones y cálculos complejos, como la lógica del negocio. Los aspectos dinámicos del sistema se modelan con clases de control, debido a que ellas manejan y coordinan las acciones y los flujos de control principales, y delegan trabajo a otros objetos, es decir, objetos de interfaz y de entidad.

Las clases entidad se utilizan para modelar la información que posee una vida larga y que es a menudo persistente, además, modelan el comportamiento asociado a algún fenómeno o concepto, como una persona, un objeto del mundo real, o un suceso del mundo real. Las clases entidad suelen mostrar una estructura de datos lógica y contribuyen a comprender de qué información depende el sistema.

A continuación se muestran los diagramas de clases del análisis del Sistema Nacional de Rehabilitación.

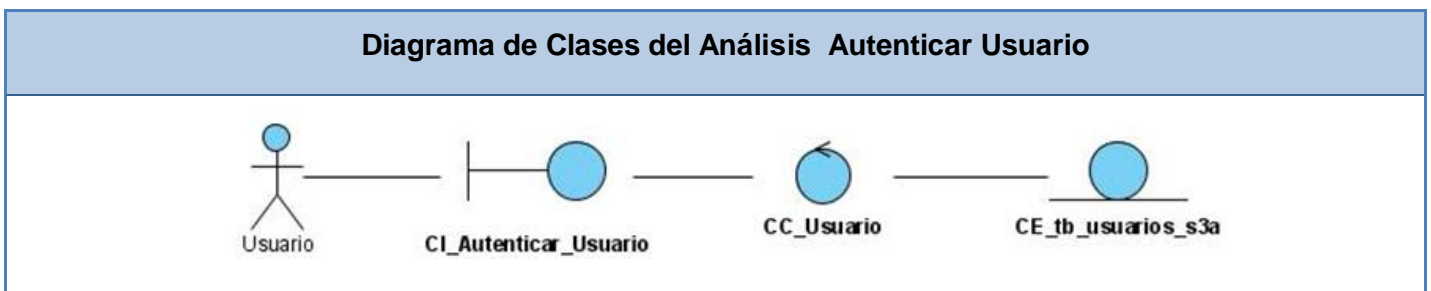




Diagrama de Clases del Análisis Buscar Paciente

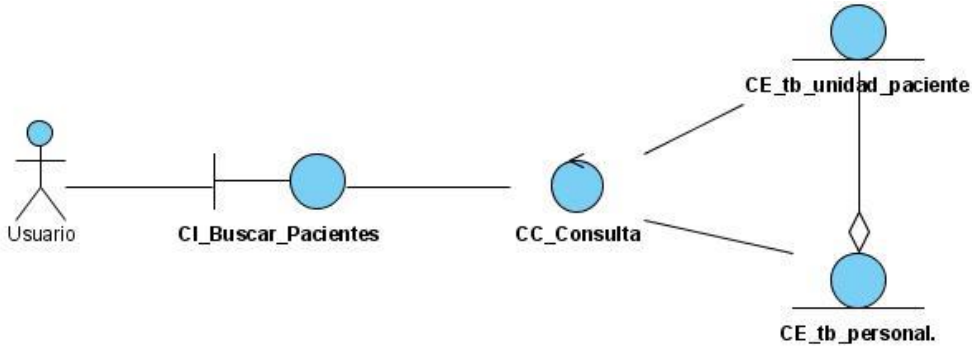
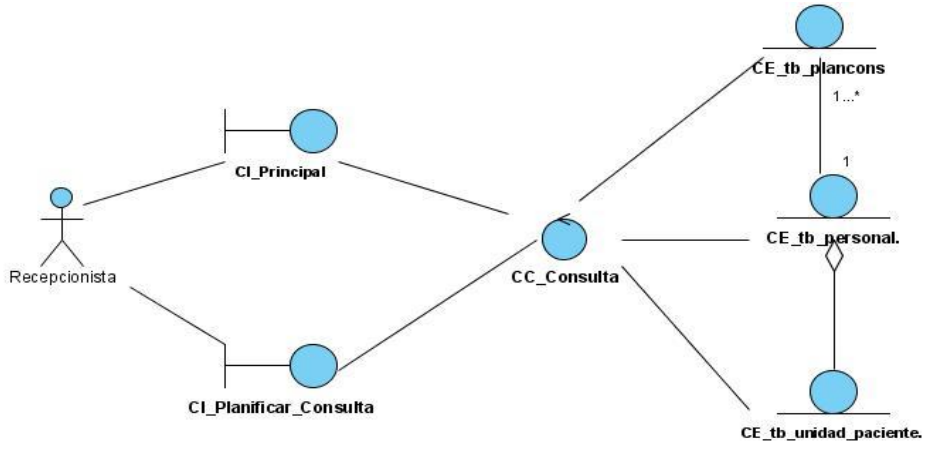


Diagrama de Clases del Análisis Planificar Consulta



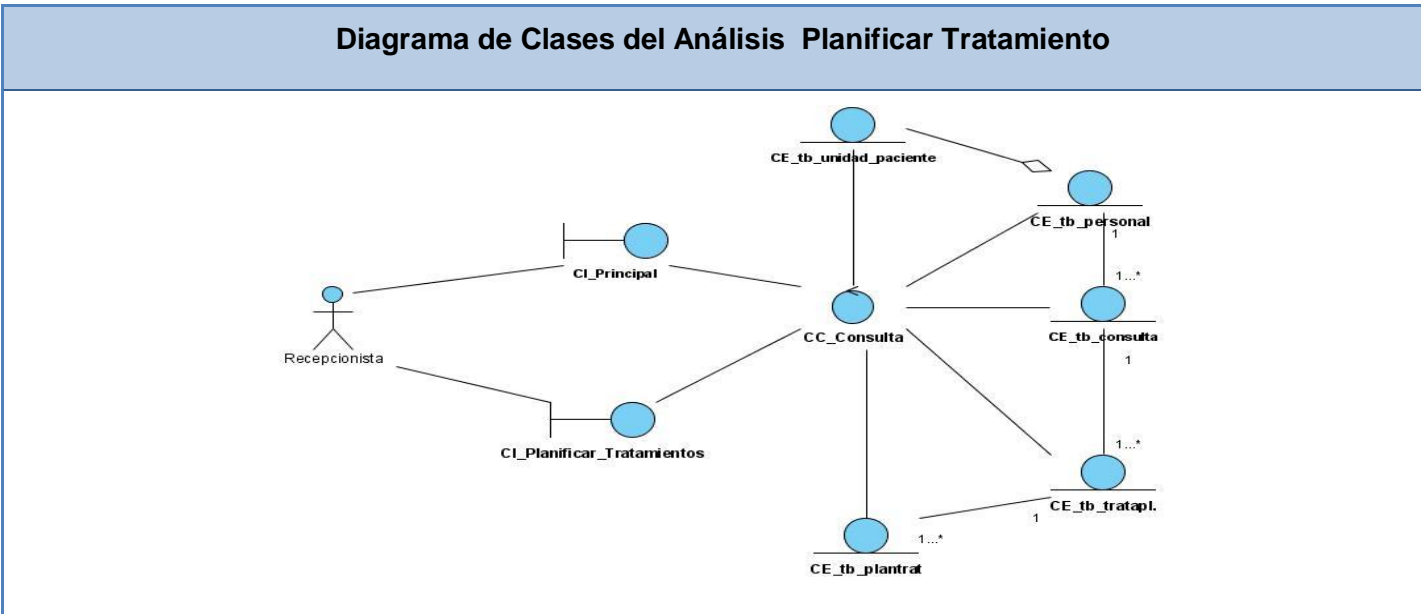
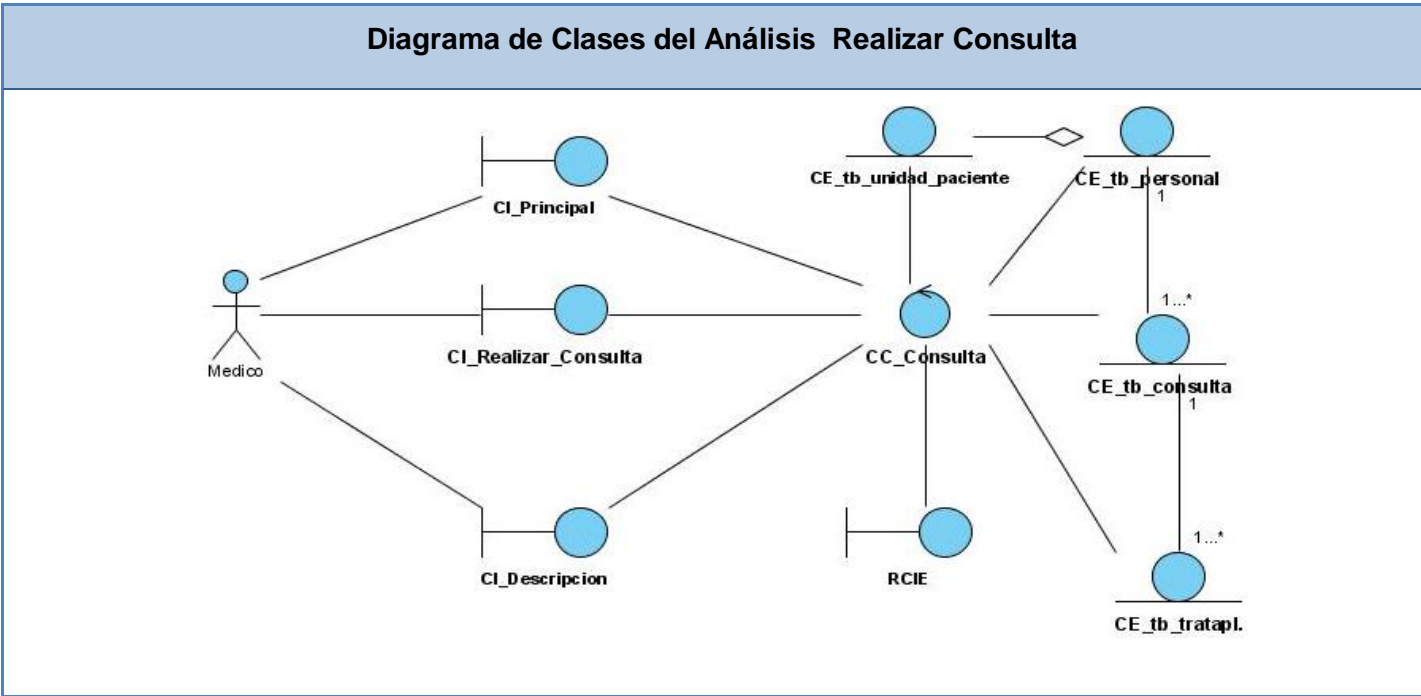


Diagrama de Clases del Análisis Interrumpir Tratamiento

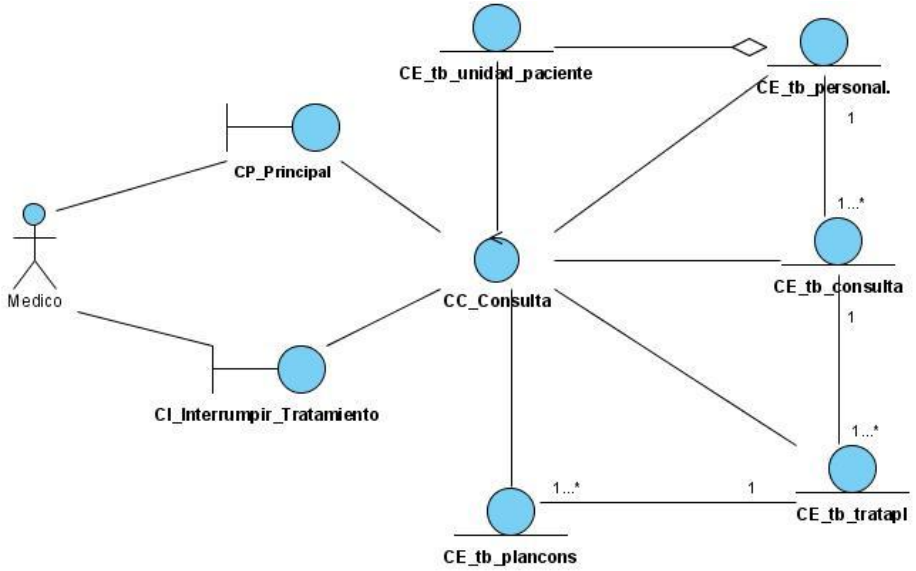


Diagrama de Clases del Análisis Asistencia Tratamiento

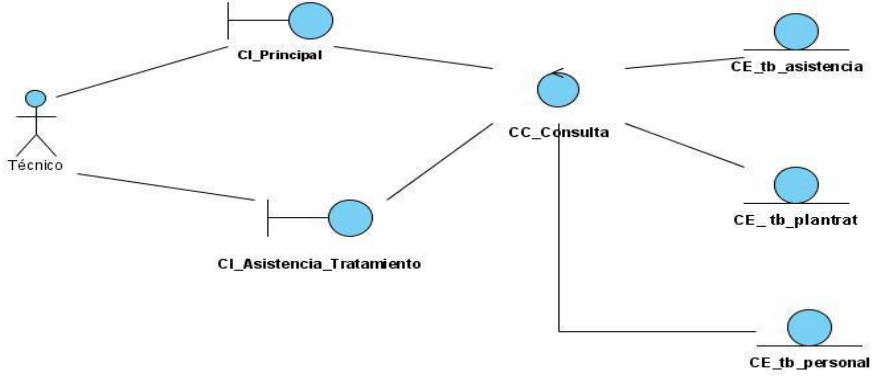


Diagrama de Clases del Análisis Gestionar Egreso

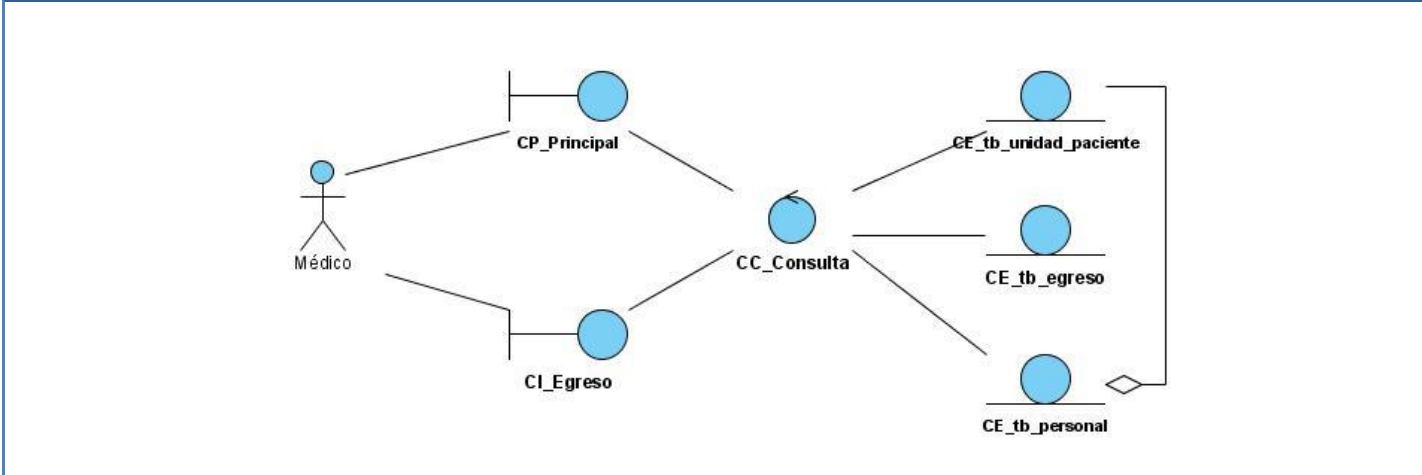
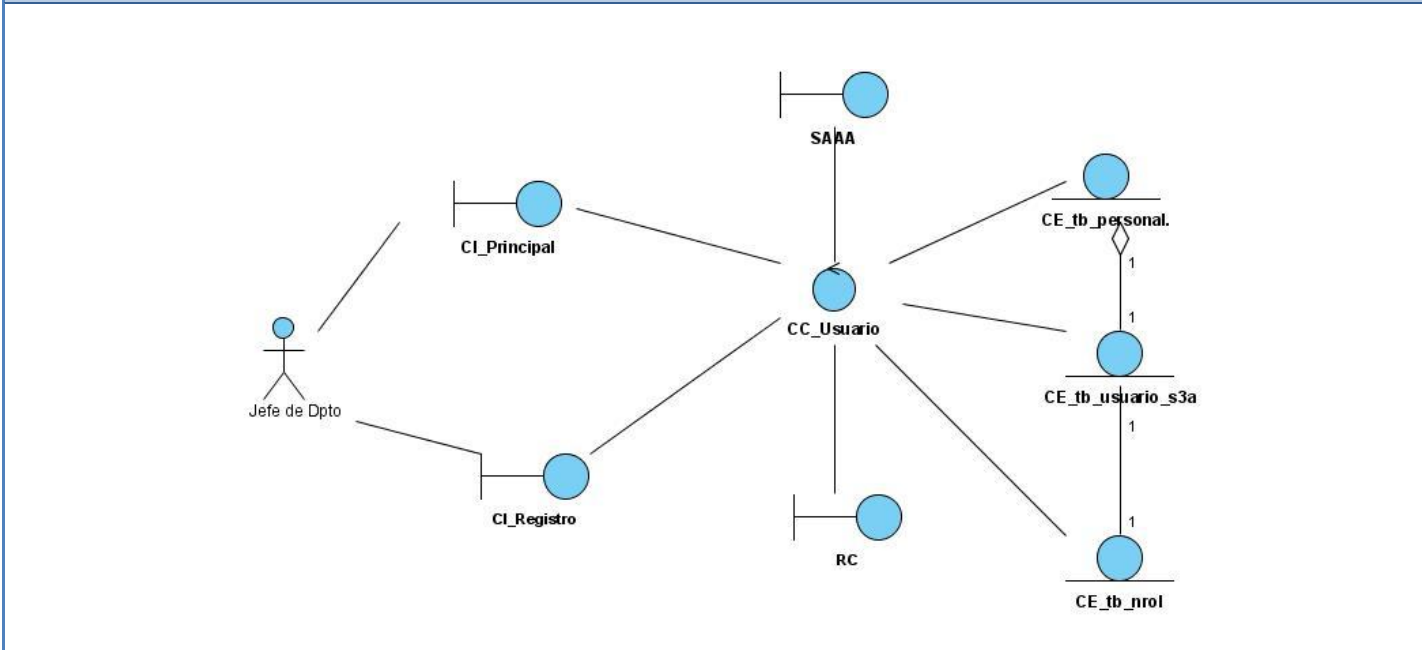
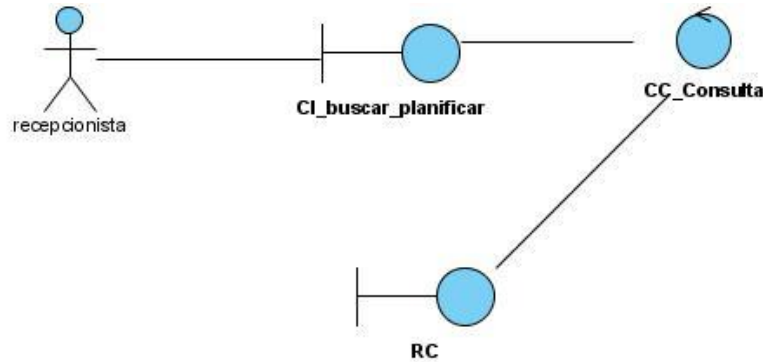


Diagrama de Clases del Análisis Registrar Personal



### Diagrama de Clases del Análisis Buscar Ciudadano



### 3.3 Modelo de Diseño

El modelo de diseño es un modelo de objetos que describe la realización de los casos de uso, y sirve como una abstracción del modelo de implementación y el código fuente. Es usado como una entrada inicial en las actividades de implementación y prueba.

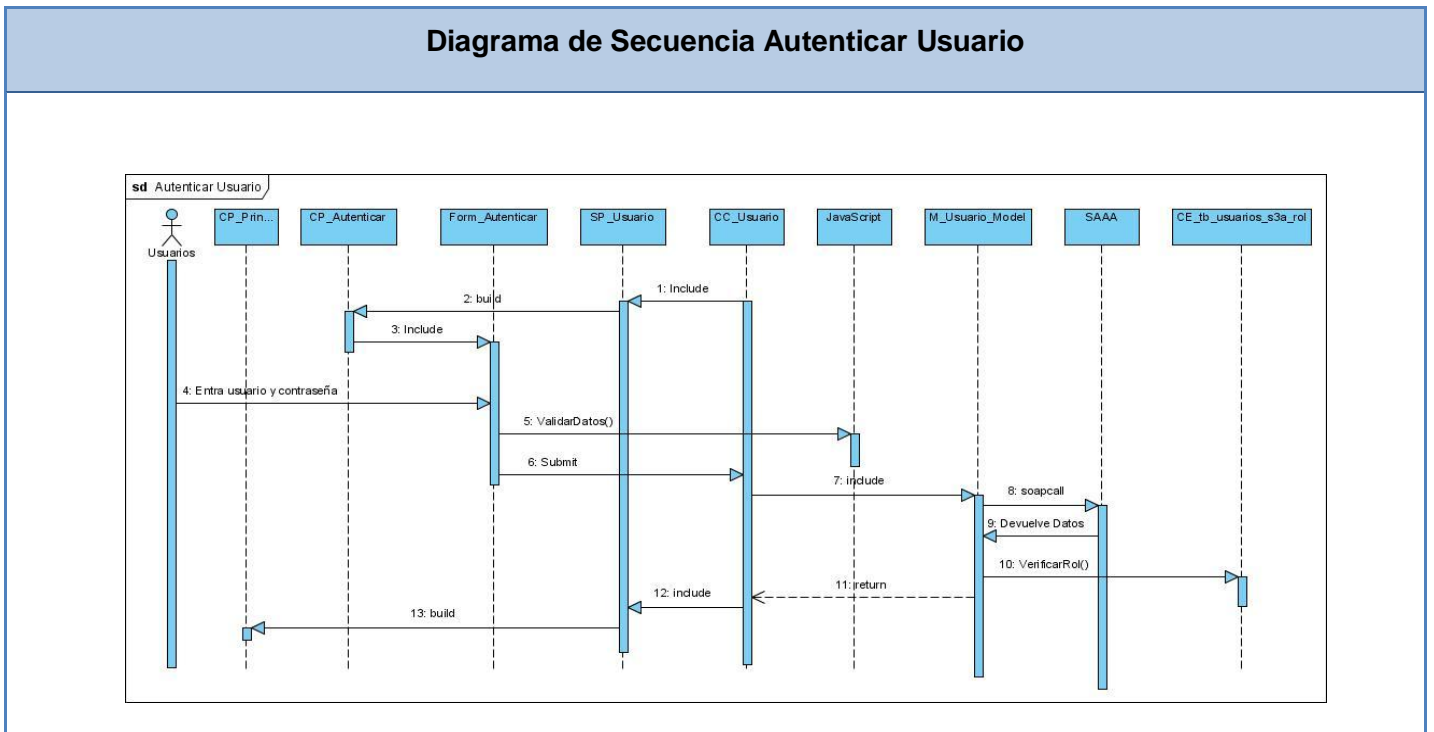
**Propósito:** el modelo de diseño es una abstracción de la implementación del sistema. Es usado para concebir un documento del diseño del sistema de software. Es abarcador, compuesto por artefactos que engloban todas las clases del diseño, subsistemas, paquetes, colaboraciones, y las relaciones entre ellos. [24]

### Diagrama de Interacción (Secuencia)

Los diagramas de secuencia y los diagramas de colaboración (ambos llamados diagramas de interacción) son dos de los cinco tipos de diagramas de UML que se utilizan para modelar los aspectos dinámicos de los sistemas. Un diagrama de interacción consiste en un conjunto de objetos y sus relaciones, incluyendo los mensajes que se pueden enviar entre ellos. [25]

Un diagrama de secuencia es un diagrama de interacción que destaca la ordenación temporal de los mensajes. Gráficamente, un diagrama de secuencia es una tabla que representa objetos, dispuestos a lo largo del eje X, y mensajes, ordenados según suceden en el tiempo, a lo largo del eje Y. [26]

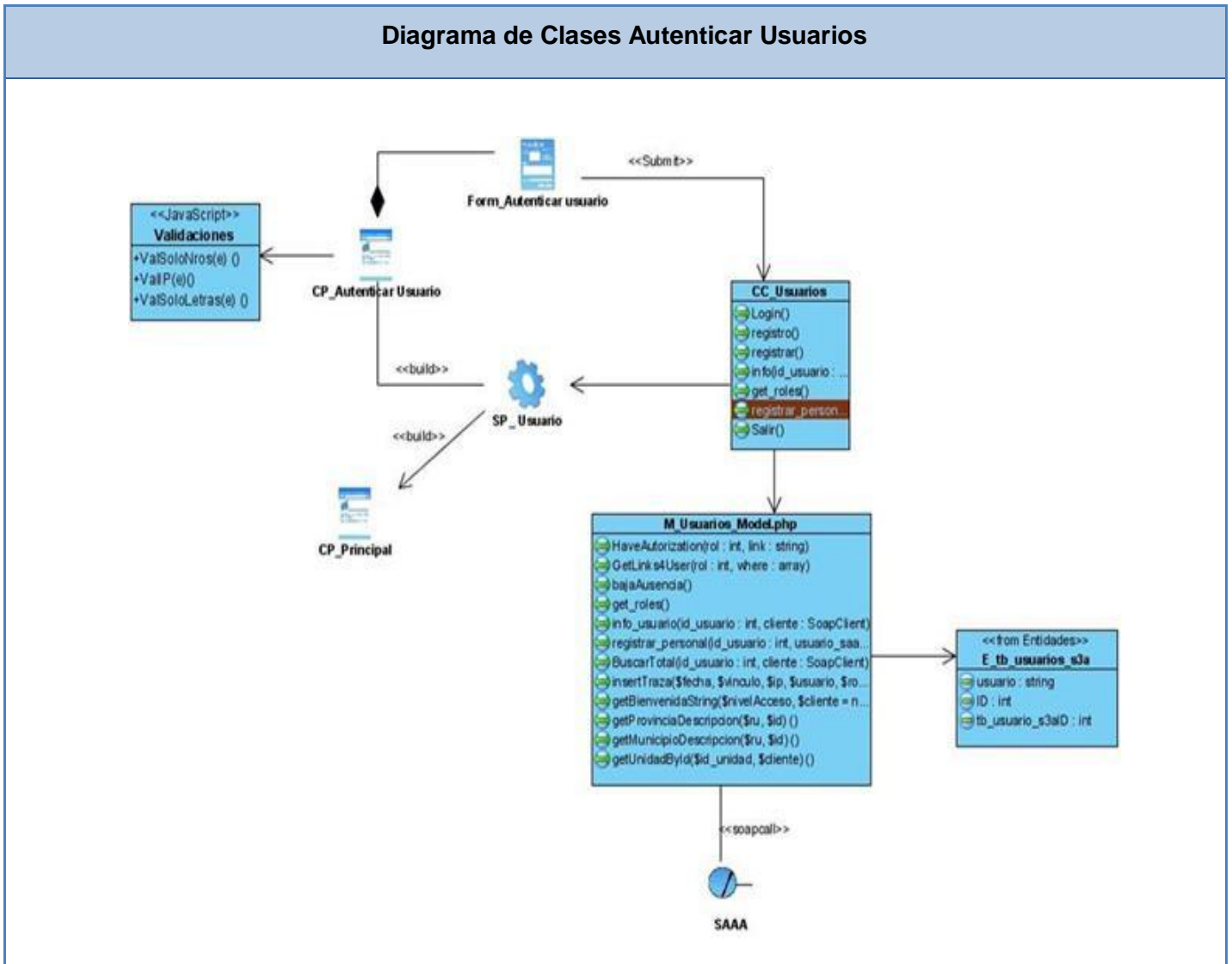
Los diagramas de secuencia son de gran importancia en el diseño de un sistema debido a que permiten observar las interacciones que ocurren entre los distintos objetos que participan en un escenario determinado. *(Ver Anexo1)*



### Diagrama de Clases del Diseño

Los diagramas de clases se utilizan para modelar la vista de diseño estática de un sistema. Esto incluye principalmente, modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas. Los diagramas de clases también son la base para un par de diagramas relacionados: los diagramas de componentes y los diagramas de despliegue. Los diagramas de clases son importantes no sólo para visualizar, especificar y documentar modelos estructurales, sino también para construir sistemas ejecutables, aplicando ingeniería directa e inversa. [27]

En el diagrama de clases del diseño se muestra la estructura estática del sistema, se crea el diseño conceptual de la información que se manejará en el sistema, así como los componentes que se encargarán del funcionamiento y la relación entre ellos. *(Ver Anexo2)*



**Descripción de las clases.**

<b>Nombre:</b> CC_Usuario
<b>Tipo de clase:</b> Controladora
<b>Para cada responsabilidad:</b>

Nombre:	Login()
Descripción:	Es utilizado para que el usuario se autentique haciendo uso de la funcionalidad autenticar del componente SAAA perteneciente al SiSalud.
Nombre:	registro()
Descripción:	Registrar profesionales de la Salud
Nombre:	registrar()
Descripción:	Realiza una autenticación con el SAAA para obtener datos del personal médico.
Nombre:	info(\$id_usuario)
Descripción:	Obtener información de un usuario del SAAA dado su id.
Nombre:	get_rol()
Descripción:	Devuelve el rol de un usuario.
Nombre:	registrar_personal()
Descripción:	Insertar datos del personal de la salud.
Nombre:	Salir()
Descripción:	Salir del Sistema

<b>Nombre: CC_Consulta</b>
<b>Tipo de clase: Controladora</b>
<b>Para cada responsabilidad:</b>



## Capítulo 3. Análisis y Diseño

Nombre:	buscar_planificar()
Descripción:	Buscar paciente para planificar consulta.
Nombre:	buscar_paciente()
Descripción:	Buscar pacientes por el RC.
Nombre:	buscar()
Descripción:	Buscar paciente dado el ID del RC.
Nombre:	disponibilidad()
Descripción:	Chequea la disponibilidad de un médico para una fecha.
Nombre:	planificar_consulta()
Descripción:	Insertar datos de la planificación de consulta.
Nombre:	buscar_consulta()
Descripción:	Muestra interfaz de la búsqueda de pacientes para consulta.
Nombre:	buscar_consulta_hoy()
Descripción:	Buscar pacientes para consulta.
Nombre:	grupos_RCIE()
Descripción:	Devuelve los grupos del RCIE.
Nombre:	categorias_RCIE()
Descripción:	Devuelve las categorías del RCIE.

Nombre:	subcategorias_RCIE()
Descripción:	Devuelve las subcategorías el RCIE.
Nombre:	enfermedad_RCIE()
Descripción:	Devuelve las enfermedades del RCIE.
Nombre:	datos_consulta()
Descripción:	Inserta los datos de la consulta
Nombre:	buscar_ptrat()
Descripción:	Muestra la interfaz de buscar pacientes con tratamientos sin planificar.
Nombre:	buscar_plantrat()
Descripción:	Buscar datos de los pacientes para planificar tratamientos.
Nombre:	ver_tratamientos()
Descripción:	Muestra todos los tratamientos de un paciente.
Nombre:	getPacTrat()
Descripción:	Devuelve los tratamientos de un paciente para planificar.
Nombre:	tratData()
Descripción:	Devuelve los datos de un tratamiento a aplicar.
Nombre:	getPlanificaciones()
Descripción:	Devuelve todas las planificaciones de un tratamiento para una fecha dada.

Nombre:	getStr(\$total, \$datos, \$cantidad, \$fecha)
Descripción:	Devuelve una cadena en formato HTML para mostrar los datos de las planificaciones.
Nombre:	insertPlanificacion()
Descripción:	Inserta las planificaciones de los tratamientos.
Nombre:	ver_asistencia()
Descripción:	Muestra la interfaz para la asistencia a los tratamientos.
Nombre:	getAsistencia()
Descripción:	Devuelve los tratamientos planificados para ese día.
Nombre:	insertAsistencia()
Descripción:	Dar asistencia a una planificación de tratamiento.
Nombre:	planData()
Descripción:	Devuelve los datos de una sesión.
Nombre:	interrumpirTratamiento()
Descripción:	Muestra la interfaz para la interrupción de tratamientos.
Nombre:	getTratamientos()
Descripción:	Devuelve los tratamientos a interrumpir.
Nombre:	getInterrupcionInfo()
Descripción:	Devuelve la información de un tratamiento a interrumpir.

Nombre:	interrumpir()
Descripción:	Insertar datos de la interrupción.
Nombre:	egreso()
Descripción:	Muestra interfaz para el egreso.
Nombre:	getEgresos()
Descripción:	Devuelve los pacientes listos para egreso.
Nombre:	getEgresoInfo()
Descripción:	Devuelve la información de un paciente listo para egreso.
Nombre:	egresarPaciente()
Descripción:	Insertar datos del egreso de un paciente.

### Descripción de las entidades.

<b>Nombre de la Clase:</b> E_tb_personal	
<b>Tipo de Clase:</b> Entidad	
<b>Atributo</b>	<b>Tipo</b>
id_personal	int
fecha_nac	string
nombre	string

## Capítulo 3. Análisis y Diseño

apellidos1	string
apellidos2	string
carnet_identidad	string
edad	int
sexo	int
calle	int
nro	string
discriminador	string
entre	string
apto	string
nombre_madre	string
nombre_padre	string
registrado_por	string
alergia	string

**Nombre de la Clase:** E\_tb\_unidad

**Tipo de Clase:** Entidad

Atributo	Tipo
----------	------

id_unidad	string
id_municipio	int
id_provincia	int

<b>Nombre de la Clase:</b> E_tb_egreso	
<b>Tipo de Clase:</b> Entidad	
<b>Atributo</b>	<b>Tipo</b>
id_egreso	int
fecha	string
rehabilitado	string
evo_final	string
otras_cosas	string

<b>Nombre de la Clase:</b> E_tb_consulta	
<b>Tipo de Clase:</b> Entidad	
<b>Atributo</b>	<b>Tipo</b>
id_consulta	int
motivo_consulta	string

microhistoria	string
grupo_rcie	string
capítulo_rcie	string
categoria_rcie	string
subcategoria_rcie	string
enfermedad_rcie	string
ecomplementario	string

**Nombre de la Clase:** E\_tb\_unidad\_paciente

**Tipo de Clase:** Entidad

Atributo	Tipo
id_unidad	int
id_personal	int

**Nombre de la Clase:** E\_tb\_usuario\_s3a

**Tipo de Clase:** Entidad

Atributo	Tipo
usuario	string

id_nrol	int
Id_personal	int

Nombre de la Clase: E_tb_tratapl	
Tipo de Clase: Entidad	
Atributo	Tipo
id_trapl	int
planificado	string
descripción	string
ausencias	int
num_ses	int
id_consulta	int
id_trat	int

Nombre de la Clase: E_tb_plantrat	
Tipo de Clase: Entidad	
Atributo	Tipo
id_plan	int



f_inicio	string
hora	string
horafin	string
asis	char

<b>Nombre de la Clase:</b> E_tb_asistencia	
<b>Tipo de Clase:</b> Entidad	
<b>Atributo</b>	<b>Tipo</b>
id_asistencia	int
fecha	string
observaciones	string

<b>Nombre de la Clase:</b> E_tb_nrol		
<b>Descripción:</b> en esta tabla se guardan los roles que tendrá la aplicación.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_rol	int	Responde al identificador de la tabla.
descripcion	char	Descripción del rol.

<b>Nombre de la Clase:</b> E_tb_plancons		
<b>Descripción:</b> en esta tabla se guarda todo lo relacionado con la planificación de las consultas.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_plancons	int	Responde al identificador de la tabla.
fecha_consulta	string	Fecha de la consulta
medico	int	Médico que realizará la consulta.
paciente	int	Id del paciente en el RC.
realizada	int	Responde si se realizó la consulta o no.

En este capítulo se ha completado el flujo de análisis y diseño con la realización de los diagramas de clases del análisis, los diagramas de clases de diseño y los diagramas de secuencia de los casos de uso del sistema, lo que proporciona una idea completa del software, materializando con precisión los requerimientos del cliente. Los diagramas y especificaciones de diseño que se proponen constituyen una guía que puede ser fácilmente leída y comprendida por los desarrolladores del sistema.

### Capítulo 4. Implementación

Uno de los flujos de trabajo que define RUP como metodología de desarrollo, es el de implementación. En este flujo se materializa la creación del analista, se refina la arquitectura y se obtiene una versión del producto. Sus propósitos se basan generalmente en definir la organización del código, implementar las clases y objetos en forma de componentes, probar los componentes desarrollados e integrarlos en un sistema ejecutable.

Este capítulo tiene por objetivo describir los diferentes elementos del modelo de datos así como la representación gráfica de los diagramas de componentes y el diagrama de despliegue.

#### 4.1 Modelo de Datos.

Al diseñar una base de datos y gestionar su información, se plasma una parte del mundo real en las tablas, registros y campos ubicados en el ordenador; creando un modelo semejante a la realidad. El principal paso para crear las tablas de la base de datos es diseñar un modelo de datos.

Este modelo es un artefacto de gran importancia dentro del proceso que incluye generar la base de datos. Proporciona una representación visual y física de los datos persistentes del sistema, que finalmente formarán las tablas de la base de datos. A partir del diagrama de clases persistentes que se diseña, se obtiene el modelo de datos. Cada clase que se diseña en el diagrama de clases persistentes constituye una entidad del modelo de datos.



### Descripción de las Tablas.

**Nombre de la Clase:** E\_tb\_personal

**Descripción:** en esta tabla se guarda todo lo relacionado con los pacientes que se atienden en la clínica y el personal que trabaja en la misma.

Atributo	Tipo	Descripción
id_personal	int	Identificador para los datos de esta tabla.
fecha_nac	string	Responde a la fecha de nacimiento.
nombre	string	Responde al nombre.
apellidos1	string	Responde al primer apellido.
apellidos2	string	Responde al segundo apellido.
carnet_identidad	string	Responde número del carnet de identidad.
edad	int	Responde edad de la persona.
sexo	int	Responde sexo de la persona.
calle	int	Nombre de las calles donde vive la persona.

## Capítulo 4. Implementación

nro	string	Número de la casa.
discriminador	string	Para identificar si la persona se insertó en la tabla como paciente o como médico.
entre	string	Entre calles.
apto	string	Número del apartamento.
nombre_madre	string	Responde al nombre de la madre.
nombre_padre	string	Responde al primer apellido del padre.
registrado_por	string	Responde al nombre de la persona que registró todos los datos.
alergia	string	Responde si el paciente presenta alergias o no.

**Nombre de la Clase:** E\_tb\_unidad\_paciente

**Tipo de Clase:** Entidad

Atributo	Tipo	Descripción
id_unidad	int	Responde al identificador de la tabla.
id_personal	int	Responde al identificador de la tabla personal.

Nombre de la Clase: E_tb_unidad		
Tipo de Clase: Entidad		
Atributo	Tipo	Descripción
id_unidad	string	Responde al identificador de la tabla.
id_municipio	int	Responde al identificador del municipio
id_provincia	int	Responde al identificador de la provincia

Nombre de la Clase: E_tb_egreso		
Descripción: en esta tabla se guarda todo lo relacionado con el egreso del paciente.		
Atributo	Tipo	Descripción
id_egreso	int	Responde al identificador de la tabla.
fecha	string	Fecha del egreso.
rehabilitado	string	Responde si el paciente se rehabilitó o no.
evo_final	string	Evolución final del paciente.
otras_cosas	string	Responde a cualquier aclaración que desee hacer el médico.

<b>Nombre de la Clase:</b> E_tb_consulta		
<b>Descripción:</b> en esta tabla se guarda todo lo relacionado con la consulta realizada por el médico.		
<b>Atributo</b>	<b>Tipo</b>	<b>Descripción</b>
id_consulta	int	Responde al identificador de la tabla.
motivo_consulta	string	Motivo de la consulta.
microhistoria	string	Comentario que realiza el médico en la consulta.
grupo_rcie	string	Grupo donde se encuentra la enfermedad del paciente el RCIE.
capitulo_rcie	string	Capítulo donde se encuentra la enfermedad del paciente el RCIE.
categoria_rcie	string	Categoría donde se encuentra la enfermedad del paciente el RCIE.
subcategoria_rcie	string	Subcategoría donde se encuentra la enfermedad del paciente el RCIE.
enfermedad_rcie	string	Enfermedad que presenta el paciente registrada en el RCIE.
ecomplementario	string	Comentario del examen que realiza el médico al paciente.



**Nombre de la Clase:** E\_tb\_usuario\_s3a

**Descripción:** esta tabla se encarga de controlar los usuarios del SAAA, el personal médico de la clínica y el rol que desempeña cada usuario en la aplicación.

Atributo	Tipo	Descripción
usuario	string	Nombre del usuario
id_personal	int	Id de la persona en el RC.
id_rol	int	Rol que ocupa.

**Nombre de la Clase:** E\_tb\_tratapl

**Descripción:** en esta tabla se guarda todo lo relacionado con los tratamientos que se le aplicarán al paciente.

Atributo	Tipo	Descripción
id_trapl	int	Responde al identificador de la tabla.
planificado	string	Planificación del tratamiento.
descripcion	string	Descripción del tratamiento que se aplicará al paciente.
ausencias	int	Ausencias del paciente durante el tratamiento.
num_ses	int	Número de sesiones que tendrá el paciente.

**Nombre de la Clase:** E\_tb\_plantrat

**Descripción:** en esta tabla se guarda todo lo relacionado con la planificación de los tratamientos.

Atributo	Tipo	Descripción
id_plan	int	Responde al identificador de la tabla.
f_inicio	string	Fecha de inicio del tratamiento.
hora	string	Hora que el paciente se realizará el tratamiento.
horafin	string	Hora de fin del tratamiento.
asis	char	Asistencia al tratamiento.
id_consulta	int	Responde al identificador de la consulta.
id_trat	int	Responde al identificador del tratamiento.

**Nombre de la Clase:** E\_tb\_asistencia

**Descripción:** en esta tabla se guarda todo lo relacionado con la asistencia del paciente a los tratamientos.

Atributo	Tipo	Descripción
id_asistencia	int	Responde al identificador de la tabla.
fecha	string	Fecha en que el paciente fue a realizarse el tratamiento.

## Capítulo 4. Implementación

observaciones	string	Valoración del técnico sobre la evolución del paciente ese día.
---------------	--------	---

Nombre de la Clase: E_tb_nrol		
<b>Descripción:</b> en esta tabla se guardan los roles que tendrá la aplicación.		
Atributo	Tipo	Descripción
id_rol	int	Responde al identificador de la tabla.
descripcion	char	Descripción del rol.

Nombre de la Clase: E_tb_plancons		
<b>Descripción:</b> en esta tabla se guarda todo lo relacionado con la planificación de las consultas.		
Atributo	Tipo	Descripción
id_plancons	int	Responde al identificador de la tabla.
fecha_consulta	string	Fecha de la consulta
medico	int	Médico que realizará la consulta.
paciente	int	Id del paciente en el RC.
realizada	int	Responde si se realizó la consulta o no.

### 4.2 Modelo Implementación.

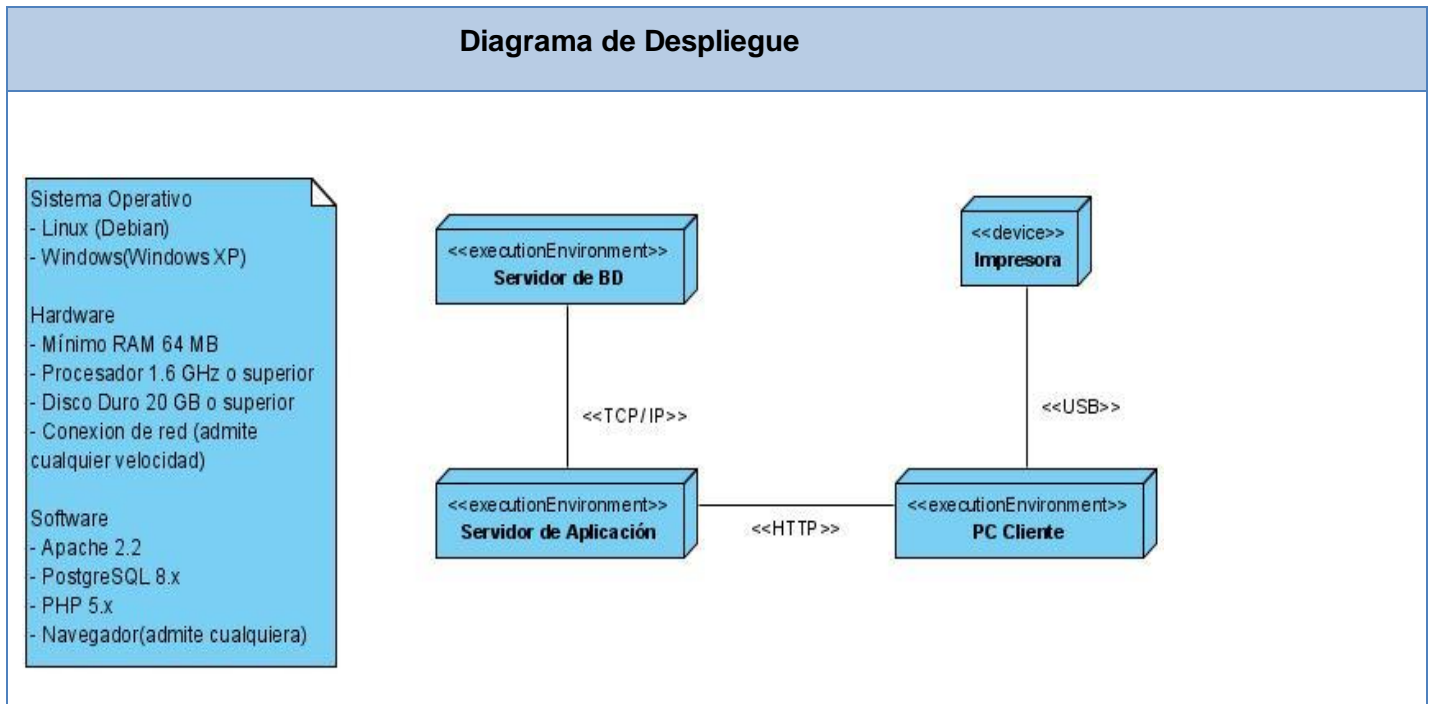
El modelo de implementación está compuesto por una colección de componentes y subsistemas. Estos componentes incluyen ficheros ejecutables, de código fuente o cualquier otro tipo necesario para obtener la versión del sistema y llevarlo al despliegue. Describe como se organizan los componentes de acuerdo con los mecanismos de estructuración disponibles en el entorno de la implementación.

### 4.3 Diagrama de Despliegue

Los diagramas de despliegue muestran las relaciones físicas de los distintos nodos que componen un sistema y el reparto de los componentes sobre dichos nodos. La vista de despliegue representa la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación. Un nodo es un recurso de ejecución tal como un computador, un dispositivo o memoria. Los estereotipos permiten precisar la naturaleza del equipo: [28]

- ✓ Dispositivos
- ✓ Procesadores
- ✓ Memoria

Los diagramas de despliegue muestran la configuración en funcionamiento del sistema, incluyendo su hardware y su software. Para cada componente de un diagrama de despliegue se deben documentar las características técnicas requeridas, el tráfico de red esperado, el tiempo de respuesta requerido. [29]



## 4.4 Diagrama de Componentes

Un diagrama de componentes muestra las organizaciones y dependencias lógicas entre componentes de software, sean éstos de código fuente, binarios o ejecutables. Desde el punto de vista del diagrama de componentes se tienen en consideración los requisitos relacionados con la facilidad de desarrollo, la gestión del software, la reutilización y las restricciones impuestas por los lenguajes de programación y las herramientas utilizadas en el desarrollo.[30]

El sistema cuenta con tres subsistemas de implementación: Vistas, Controladoras y Modelo, estos están estructurados según el patrón arquitectónico MVC. En el subsistema Vistas se encuentran encapsulados los componentes que permiten la interacción directa con los usuarios del sistema. El subsistema Controladoras contiene todas las clases que manipulan los eventos del usuario y realizan peticiones al modelo para mostrarlas en las vistas. El subsistema Modelo, agrupa las clases que interactúan con la base de datos y velan por el cumplimiento de las reglas del negocio.

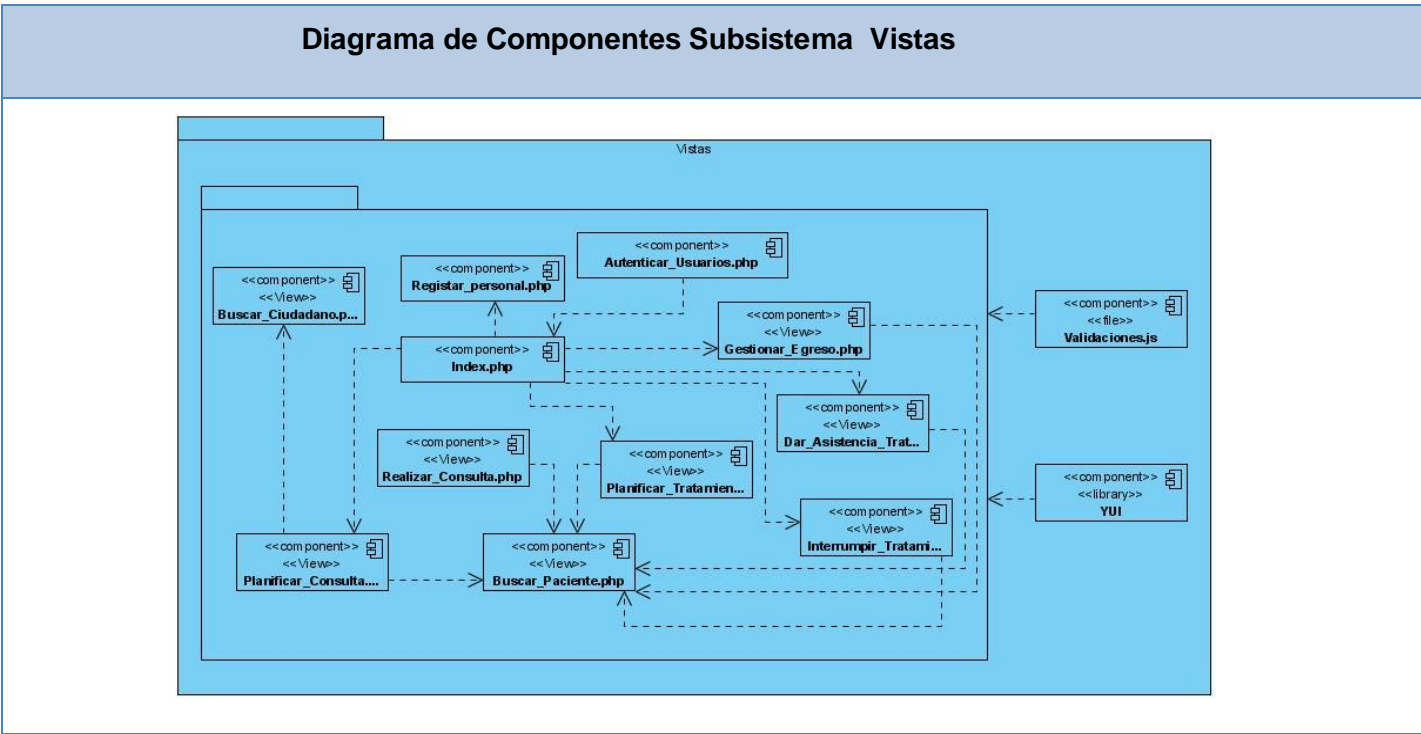
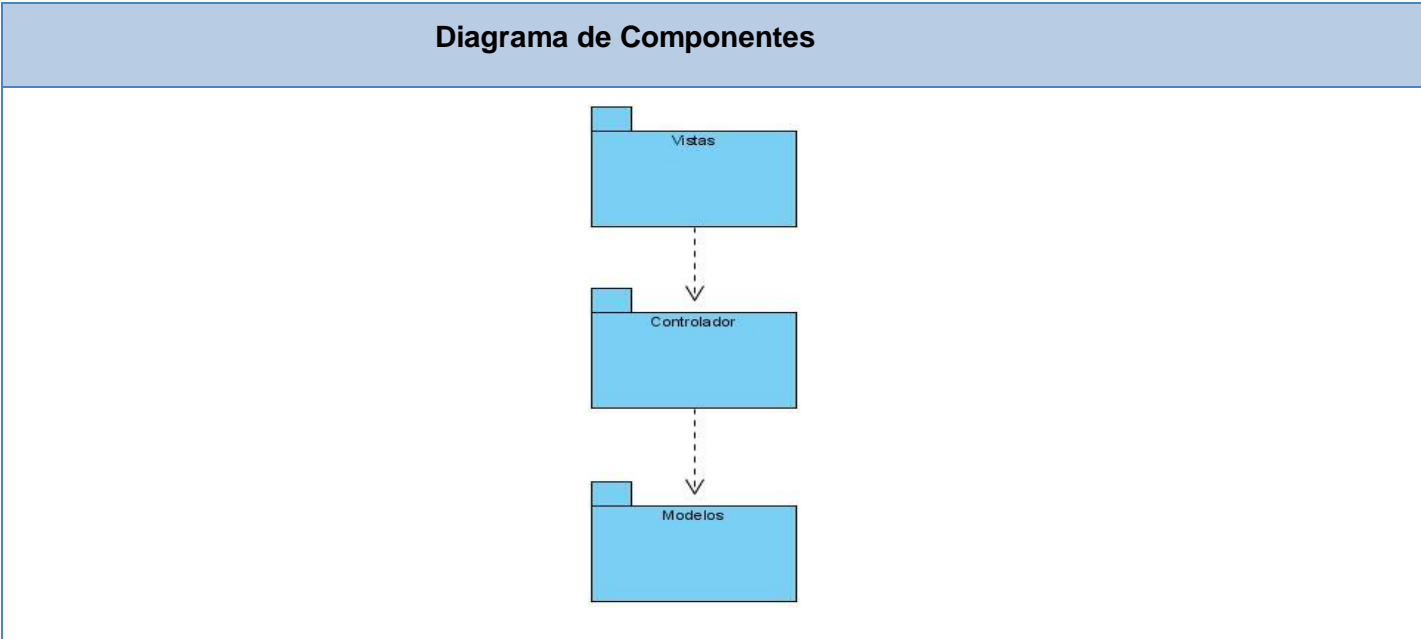


Diagrama de Componentes Subsistema Controladoras

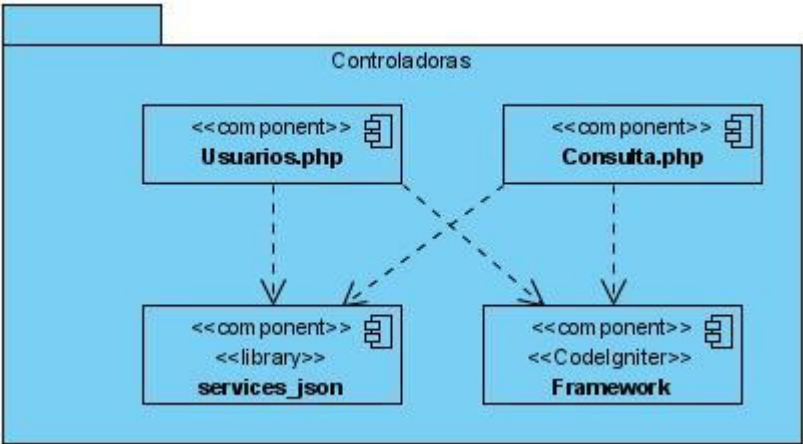
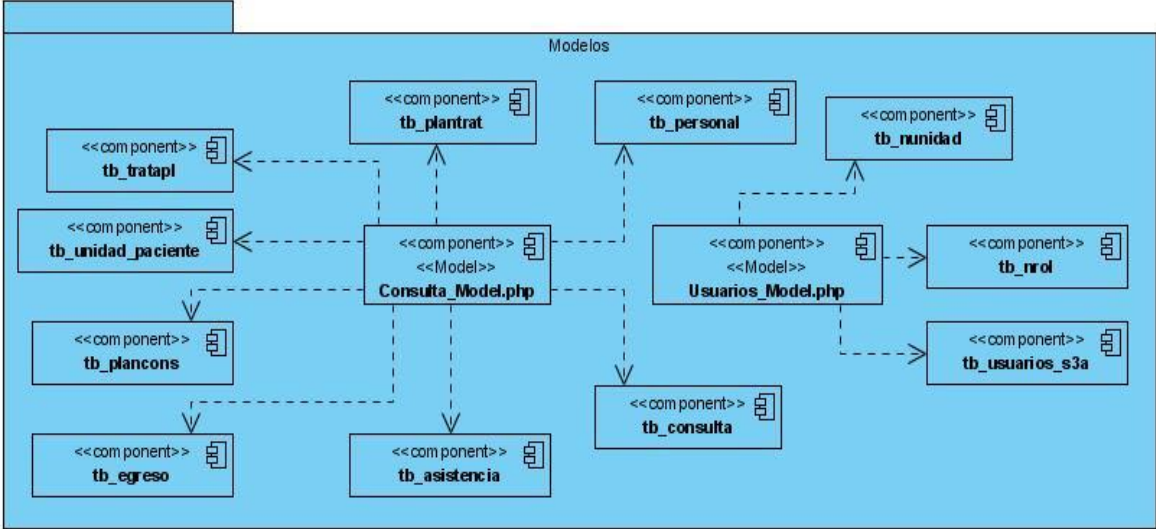
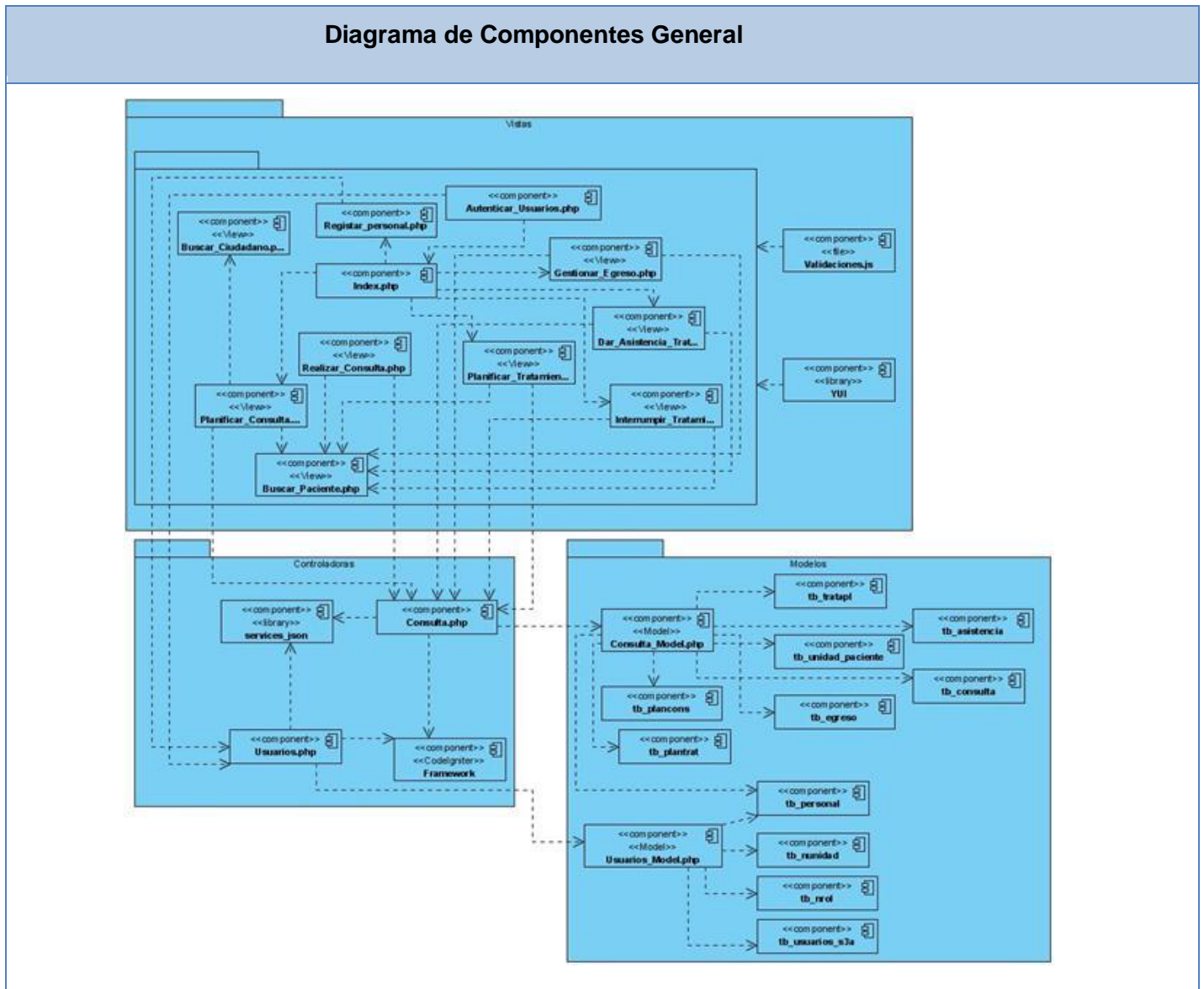


Diagrama de Componentes Subsistema Modelos



En el siguiente diagrama se ilustran las relaciones entre los componentes a modo general, pudiéndose apreciar de forma práctica como se implementa el patrón arquitectónico MVC.



## 4.5 Tratamiento de Errores

En el sistema se tiene en cuenta el tratamiento de todos los posibles errores que puedan aparecer, para garantizar así la integridad y confiabilidad de los datos. Estos se tratan en una página especial que incluye



el fichero de configuración general. Preparada para recoger los datos de los errores y presentar al usuario el tipo de error.

El sistema también muestra mensajes de error, que son de fácil comprensión para el usuario. Se pueden observar cuando se insertan datos y no se entran los campos que son obligatorios, cuando se registra un paciente que ya existe en la base de datos, ocurriendo lo mismo cuando se inserta información errónea en campos numéricos o un e-mail.

### 4.6 Seguridad

La seguridad de un sistema es de suma importancia para preservar la integridad, autenticidad y confiabilidad de la información. Para garantizar esto, se hizo uso del Registro de Seguridad, Autenticación, Autorización y Auditoría (SAAA) que brinda el Sistema de Información para la Salud (SiSalud). Este componente implementa un servicio de seguridad basado en el modelo de Autenticación, Autorización y Auditoría (AAA), aquí se describe la operación de autenticación, que responde al esquema organizativo de seguridad establecido por el Ministerio de Salud Pública (MINSAP) y se basa en tres capas: Niveles, Tipos de Usuario, Módulos.

En este modelo se implementan políticas de seguridad en las cuentas como: tamaño del usuario y la contraseña, expiración por tiempo de la contraseña, historicidad de las contraseñas, no permitir conexiones concurrentes en una misma cuenta, chequeo por parte de los administradores del tiempo de inactividad, registro de cuentas creadas, bloqueo de las cuentas por repetición fallida, obligación del uso de contraseñas fuerte, etc.

Para el acceso a las funcionalidades del sistema se crearon usuarios con diferentes roles que garantizan niveles de acceso jerárquicos dentro de la aplicación. Cada usuario podrá acceder a las funcionalidades definidas en dependencia del rol que desempeñe dentro del sistema.

La seguridad en el sistema también es garantizada evitando que un usuario acceda a una página escribiendo la dirección en el navegador o en el caso que un usuario deje por descuido abierta su sesión esta expirará en un período de 5 minutos. Para evitar el acceso directo a las tablas de la base de datos, la misma se ha asegurado con una contraseña.

### 4.7 Estrategias de Codificación. Estándares y estilos a utilizar.

Idioma: Se debe utilizar como idioma el español, las palabras no se acentuarán.

Palabras Reservadas: Las palabras reservadas van en minúsculas sin excepción alguna.

Identación		
<b>Objetivos:</b> Lograr una estructura uniforme para los bloques de código así como para los diferentes niveles de anidamiento		
<b>0 espacios del tabulador desde la izquierda en</b>	require include class	No se empleará ningún espacio del tabulador desde la izquierda para las instrucciones antes mencionadas. Se tomará como inicio de la página la etiqueta PHP <?
<b>1 espacio del tabulador desde la izquierda en</b>	function	Se dejará un espacio del tabulador desde la izquierda en las instrucciones antes mencionadas.
<b>1 espacio del tabulador desde la izquierda en</b>	Inicio y fin de bloque	Se recomienda dejar un espacio del tabulador desde la instrucción anterior para el inicio y fin de bloque {}. Lo mismo sucede para el caso de las instrucciones If, else, For, While, Do While, Switch, Foreach.
<b>Niveles de anidación</b>	Hasta 5 niveles	Se recomienda emplear hasta 5

	niveles de anidación en instrucciones If, For, While.
<b>Aspectos Generales</b>	Los inicios ({} y cierre (}) de ámbito deben estar alineados debajo de la declaración a la que pertenecen y deben evitarse si hay sólo una instrucción. Nunca colocar { en la línea de un código cualquiera, esto requiere una línea propia.

### Ejemplo de Identación

```
<?php
class MiClase
{
    function buscarPaciente($nombre, $ci)
    {
        if($ci)
        {
            ...
        }
        else
        {
            for ($i = 0;$i < $cantidadEquipos; $i++)
            {
                ...
            }
        }
    }
}
```

<pre>} ?&gt;</pre>		
<b>Comentarios, separadores, líneas, espacios en blanco y márgenes.</b>		
<b>Objetivo:</b> Crear una forma común de comentar el código que sea comprensible con tan sólo leerlo una vez.		
<b>Ubicación de comentarios</b>	Al inicio de cada clase o función.	Se recomienda comentar al inicio de cada clase o función expresando su objetivo.
<b>Separador de instrucciones</b>	Se utiliza el punto y coma (;)	Se recomienda el uso del separador al final de cada línea de instrucción. Ejemplo: \$cadenaTexto = "ejemplo";
<b>Líneas en blanco</b>	Se utilizan antes de una clase o función.	Se recomienda dejar una línea en blanco antes y después de la declaración de una clase o función.
<b>Espacios en blanco</b>	Entre operadores lógicos y aritméticos.	Se recomienda dejar un espacio en blanco entre los operadores para lograr mayor comprensión del código.
<b>Márgenes y líneas de continuidad</b>	Sobre márgenes y líneas de continuidad.	Los márgenes de cada línea de código no deben exceder los 80 caracteres, pero puede exceptuarse si es para terminar la escritura de una palabra. Las

## Capítulo 4. Implementación

		líneas de continuación deben estar alineadas entre sí e indentadas respecto al paréntesis abierto.
<b>Aspectos Generales</b>	Sobre los comentarios	Se debe evitar comentar cada línea de código. Cuando el comentario se aplica a un grupo de instrucciones debe estar seguido de una línea en blanco. En caso de que se necesite comentar una sola instrucción se suprime la línea en blanco o se escribe a continuación de la instrucción.
	Sobre espacios en blanco	No se debe usar espacio en blanco: Después del corchete abierto y antes del cerrado de un arreglo. Después del paréntesis abierto y antes del cerrado.
<b>Variables</b>		
<b>Apariencia de variables</b>	Tendrán un nombre descriptivo.	Los nombres de las variables deberán empezar con minúscula, en caso de que sea un nombre compuesto se pone junto y con mayúscula inicial el segundo nombre. El nombre empleado, debe permitir que con sólo leerlo

## Capítulo 4. Implementación

		<p>se conozca el propósito de la misma.</p> <p>Ejemplo: \$horaConsulta;</p>
<b>Clases</b>		
<b>Objetivo:</b> Nombrar las clases e instancias de forma estándar.		
<b>Para nombrar clases.</b>	Se utilizarán nombres descriptivos.	Los nombres se escribirán en minúsculas. Para diferenciar las clases modelos, sus nombres terminarán en “_model”.
<b>Apariencia de clases y objetos.</b>	Se nombrarán en minúsculas.	Los nombres de las clases se escribirán en minúsculas. En caso de nombres compuestos se escriben juntos y con mayúscula inicial el segundo nombre.
<b>Apariencia de atributos.</b>	Se escribirán en minúsculas.	El nombre que se le da a los atributos de las clases debe ser en minúsculas. En caso de nombres complejos se escribirán juntos y con mayúscula inicial el segundo nombre.
<b>Apariencia de las funciones.</b>	Se escribirán en minúsculas.	Para nombrar las funciones se debe tratar de utilizar verbos que denoten la acción que hace la función. Si son funciones que obtienen un dato se emplea el

## Capítulo 4. Implementación

		prefijo get.
<b>Declaración de parámetro en funciones.</b>	Se deben agrupar por tipos. Si alguno puede tomar valor nulo se pone al final.	Los parámetros que se le pasan a las funciones se recomienda sean declarados de forma tal que estén agrupados por el tipo de dato que contienen.
<b>Aspectos Generales</b>	Sobre las clases, los objetos, los atributos y las funciones.	Los nombres deben ser descriptivos con el propósito de conocer su objetivo con tan sólo leerlos.
<b>Bases de Datos (BD), Tablas y Campos.</b>		
<b>Apariencia de la BD</b>	Se modeló la BD haciendo uso de los estándares de codificación de la UCI.	La BD se nombra comenzando con el prefijo "UCI_". Luego continúa con un nombre descriptivo.
<b>Apariencia de las tablas y vistas de la BD.</b>	Se rige por los estándares de codificación de la UCI.	Los nombres de las vistas y las tablas se escriben en minúsculas.  Los nombres de las tablas comienzan con el prefijo "tb_", en caso que sean tablas de nomencladores comienzan con el prefijo "tb_n", luego continúan con un nombre descriptivo. Ejemplos: "tb_consulta" "tb_ntratamiento"  Los nombres de las vistas

## Capítulo 4. Implementación

---

		comienzan con el prefijo “v_” y continúan con un nombre descriptivo. Ejemplo: “v_planificaciones”
<b>Apariencia de los campos.</b>	Se rige por los estándares de codificación de la UCI.	Los nombres comienzan con minúsculas.  Los identificadores comienzan con el prefijo “id_”.  Si el tipo de datos es “bool” comienzan con el prefijo “es_”.

En este capítulo se ha completado el flujo de implementación y despliegue con la realización del modelo de datos describiendo las principales tablas del mismo, los diagramas de componentes y el diagrama de despliegue. Además se proporciona una idea completa de como se lleva a cabo la seguridad del sistema; así como el tratamiento de errores, los estándares y estilos a utilizar como estrategias de codificación.



### CONCLUSIONES

Una vez concluido el presente trabajo se han cumplido el objetivo y las tareas propuestas, se implementó la nueva versión del “Sistema Automatizado para la Gestión de la Información en Rehabilitación” que ha pasado una rigurosa fase de pruebas por el grupo de calidad de la Facultad 7 como resultado de la cual todas las funcionalidades han sido completadas satisfactoriamente, y se ha realizado en cada una de ellas, una buena gestión de errores.

Se demostró que las diferentes aplicaciones que han sido elaboradas y diseñadas para gestionar la información en las clínicas de rehabilitación, no se ajustan al Sistema Nacional de Salud, además no cumplen con los requerimientos que exige el cliente.

Se identificaron los principales procesos que se llevan a cabo en dichos centros como: planificar consulta médica, realizar consulta médica y aplicar tratamiento médico; determinándose la creación de una nueva versión del “Sistema Automatizado para la Gestión de la Información en Rehabilitación”. Se definieron como herramientas a utilizar PostgreSQL de Gestor de Base de Datos, PHP5 como Lenguaje de Programación, como Metodología de Desarrollo de Software el Proceso Unificado de Desarrollo más conocido por sus siglas RUP, UML como Lenguaje de Modelado y Visual Paradigm for UML 2.0 .

Se modelaron los flujos de trabajo propuestos por el Proceso Unificado de Desarrollo: Modelado del Negocio, Gestión de Requerimientos, Análisis y Diseño, e Implementación y Despliegue.

### RECOMENDACIONES

Teniendo como base los resultados de esta investigación y la experiencia adquirida durante el desarrollo de la misma, se proponen las siguientes recomendaciones:

- La utilización total de los componentes del YUI para mejorar la creación de las opciones del menú, plantillas, entre otros.
- La utilización de la Clasificación Internacional del Funcionamiento de la Discapacidad y de la Salud (CIF).
- Luego de realizar las pruebas al sistema proponer la utilización y generalización en las diferentes salas de rehabilitación del país.

### REFERENCIAS BIBLIOGRÁFICAS

[1] Instituto Nacional para Ciegos. [En línea] [Citado el: 5 de Noviembre 2008.]

<http://www.inci.gov.co/rehabilitacion.shtml>

[2] Orígenes de la Fisioterapia. Introducción a las Ciencias de la Salud. [En línea] [Citado el: 8 de Noviembre 2008.] <http://www.juntadeandalucia.es/averroes/~29701428/salud/fisio.htm>.

[3] Orígenes de la Fisioterapia. Introducción a las Ciencias de la Salud. [En línea] [Citado el: 8 de Noviembre 2008.] <http://www.juntadeandalucia.es/averroes/~29701428/salud/fisio.htm>.

[4] **Llanes, Nestor.** Sistema Automatizado para la Gestión de la Información en Rehabilitación. Ciudad Habana: s.n., 2007.

[5] Programa de derechos humanos y discapacidad. [En línea] [Citado el: 11 de Noviembre del 2008.] <http://www.discapacidad.gov.co/intranet/organizacion/doc-consulta/prorehabilitacion.doc>.

[6] Al servicio de los más necesitados. [En línea] [Citado el: 15 de Noviembre 2008.] <http://www.hoy.com.do/investigacion/2008/8/17/244007/Al-servicio-de-los-mas-necesitados>

[7] Atención Integral. [En línea] [Citado el: 17 de Noviembre 2008.]

[http://www.iin.oea.org/Cursos\\_a\\_distancia/cad\\_guia\\_disc\\_UT5.pdf](http://www.iin.oea.org/Cursos_a_distancia/cad_guia_disc_UT5.pdf)

[8] Fisiogest. [En línea] [Citado el: 17 de Noviembre 2008.]

<http://www.fisiogest.es/>

[9] ALC Físio. [En línea] [Citado el: 17 de Noviembre 2008.]

<http://www.alcprogramas.com/>

[10] Físio Office 2008. [En línea] [Citado el: 20 de Noviembre 2008.]

<http://www.fisiooffice.com.br/>

[11] IPFísio. [En línea] [Citado el: 20 de Noviembre 2008.]

<http://www.ipfisio.com/inicio>

- [12] **Enrique, Abarmarys y Miranda, Daniel.** Sistema Automatizado para la Gestión de la Información en Rehabilitación. Ciudad Habana: s.n., 2007.
- [13] Ídem a la referencia 11
- [14] Ídem a la referencia 11
- [15] Ídem a la referencia 11
- [16] Ídem a la referencia 11
- [17] Ídem a la referencia 11
- [18] Ídem a la referencia 11
- [19] Ídem a la referencia 11
- [20] Ídem a la referencia 11
- [21] **Ingeniería de Software 1.** Fase de Inicio. Modelo del Negocio. UCI. curso 2008\_2009. Conferencia #2.
- [22] Proceso Unificado de Desarrollo de Software. [En línea] [Citado el : 5 de Diciembre del 2008]. <http://www.chaco.gov.ar/UTN/disenodesistemas/apuntes/oo/ApunteRUP.pdf>.
- [23] Ídem a la referencia 11
- [24] **Ingeniería de Software 2.** “Continuación del FT Análisis y Diseño. Modelo de Diseño.” UCI. curso 2008\_2009. Conferencia # 1.
- [25] Ídem a la referencia 23
- [26] Ídem a la referencia 23
- [27] Ídem a la referencia 23
- [28] Diagrama de despliegue [En línea] [Citado el : 10 de Marzo del 2009] <http://virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc>
- [29] Ídem a la referencia 23
- [30] Diagrama de Componentes [En línea] [Citado el : 15 de Marzo del 2009] <http://tvdi.det.uvigo.es/~avilas/UML/node49.html>

### BIBLIOGRAFÍA

**Enrique, Abarmarys y Miranda, Daniel.** Sistema Automatizado para la Gestión de la Información en Rehabilitación. Ciudad Habana: s.n., 2007.

**Ingeniería de Software 1.** Fase de Inicio. Modelo del Negocio. UCI. curso 2008\_2009. Conferencia # 2.

**Ingeniería de Software 2.** “Continuación del FT Análisis y Diseño. Modelo de Diseño.” UCI. curso 2008\_2009. Conferencia # 1.

**Llanes, Nestor.** Sistema Automatizado para la Gestión de la Información en Rehabilitación. Ciudad Habana: s.n., 2007

Portal Instituto Nacional para Ciegos [Citado el: 5 de Noviembre 2008.]

<http://www.inci.gov.co/rehabilitacion.shtml>

Portal Conceptos básicos de Rehabilitación y Habilitación [Citado el: 5 de Noviembre 2008.]

<http://es.shvoong.com/medicine-and-health/epidemiology-public-health/1863076-conceptos-b%C3%A1sicos-rehabilitaci%C3%B3n-habilitaci%C3%B3n/>

Portal Orígenes de la Fisioterapia. Introducción a las Ciencias de la Salud. [En línea] [Citado el: 8 de Noviembre 2008.]

<http://www.juntadeandalucia.es/averroes/~29701428/salud/fisio.htm>.

Portal Orígenes de la Fisioterapia. Introducción a las Ciencias de la Salud. [En línea] [Citado el: 8 de Noviembre 2008.].

<http://www.juntadeandalucia.es/averroes/~29701428/salud/fisio.htm>.

Portal Programa de derechos humanos y diacapacidad. [En línea] [Citado el: 11 de Noviembre del 2008.]

<http://www.discapacidad.gov.co/intranet/organizacion/doc-consulta/prorehabilitacion.doc>.

Portal Al servicio de los más necesitados. [En línea] [Citado el: 15 de Noviembre 2008.]

<http://www.hoy.com.do/investigacion/2008/8/17/244007/Al-servicio-de-los-mas-necesitados>

Portal Atención Integral. [En línea] [Citado el: 17 de Noviembre 2008.]

[http://www.iin.oea.org/Cursos\\_a\\_distancia/cad\\_guia\\_disc\\_UT5.pdf](http://www.iin.oea.org/Cursos_a_distancia/cad_guia_disc_UT5.pdf)

Portal Fisiogest. [En línea] [Citado el: 17 de Noviembre 2008.]

<http://www.fisiogest.es/>

Portal ALC Físio. [En línea] [Citado el: 17 de Noviembre 2008.]

<http://www.alcprogramas.com/>

Portal Físio Office 2008. [En línea] [Citado el: 20 de Noviembre 2008.]

<http://www.fisiooffice.com.br/>

Portal IPFísio. [En línea] [Citado el: 20 de Noviembre 2008.]

<http://www.ipfisio.com/inicio>

Portal Proceso Unificado de Desarrollo de Software. [En línea] [Citado el : 5 de Diciembre del 2008].

<http://www.chaco.gov.ar/UTN/disenodesistemas/apuntes/oo/ApunteRUP.pdf>.

Portal "HelMakers". [En línea] [Citado el: 22 de enero de 2009.] <http://helpmaker.uptodown.com>

Portal "pixelco.us blog". [En línea] [Citado el: 7 de febrero de 2009.]

<http://pixelco.us/blog/codeigniter-framework-php>

Portal Diagrama de despliegue [En línea] [Citado el : 10 de Marzo del

2009] <http://virtual.usalesiana.edu.bo/web/practica/archiv/despliegue.doc>

Portal Diagrama de Componentes [En línea] [Citado el : 15 de Marzo del 2009]

<http://tvdi.det.uvigo.es/~avilas/UML/node49.html>

Portal "PHP Nuke". [En línea] [Citado el: 17 de Marzo de 2009.]

[http://www.adelat.org/media/docum/nuke\\_publico/lenguajes\\_del\\_lado\\_servidor\\_o\\_cliente.html](http://www.adelat.org/media/docum/nuke_publico/lenguajes_del_lado_servidor_o_cliente.html)

Portal de Desarrollo Web. [En línea] [Citado el: 22 de Marzo de 2009.]

<http://www.desarrolloweb.com/articulos/25.php>

Anexo 1 Diagramas de Secuencia

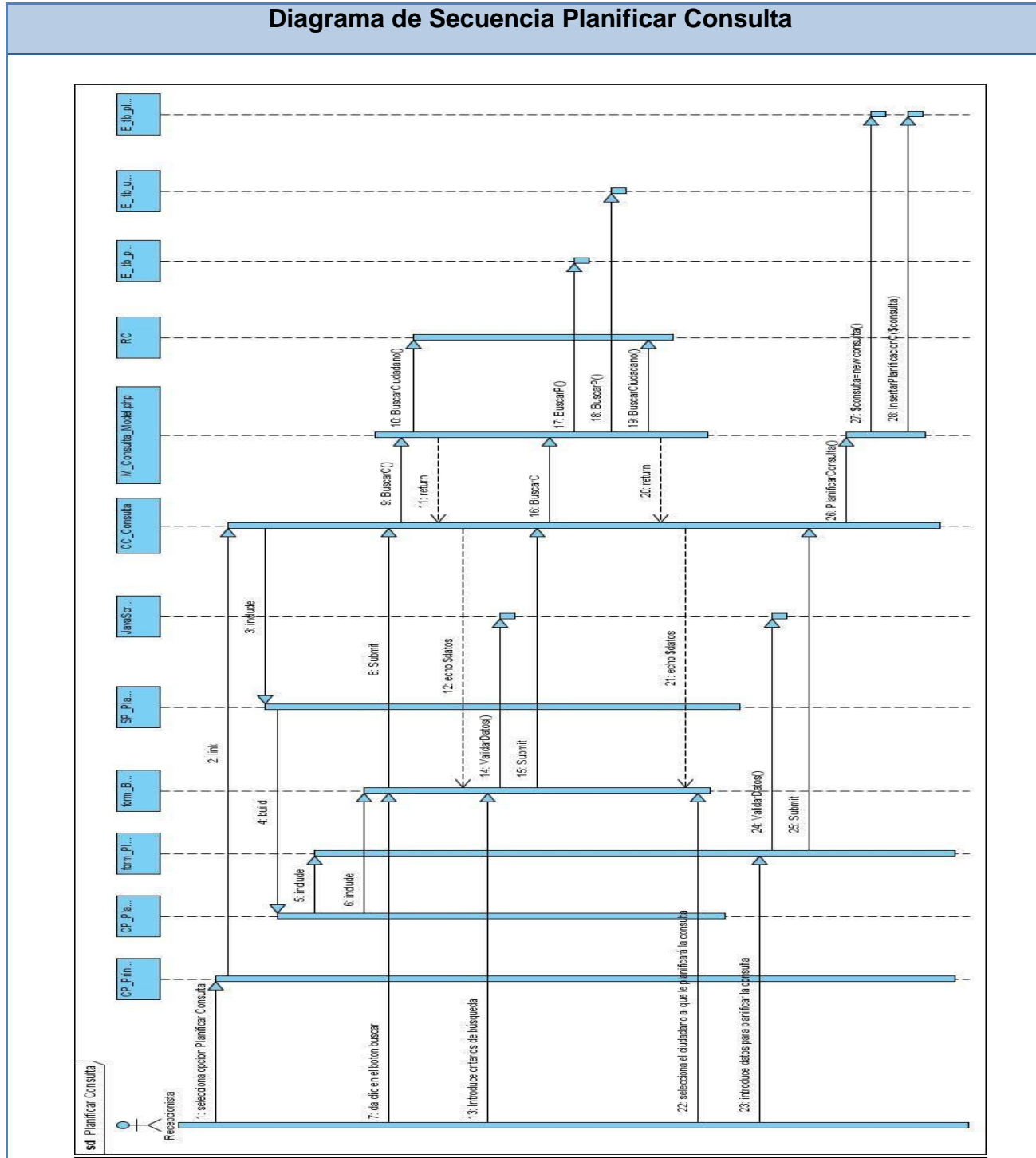
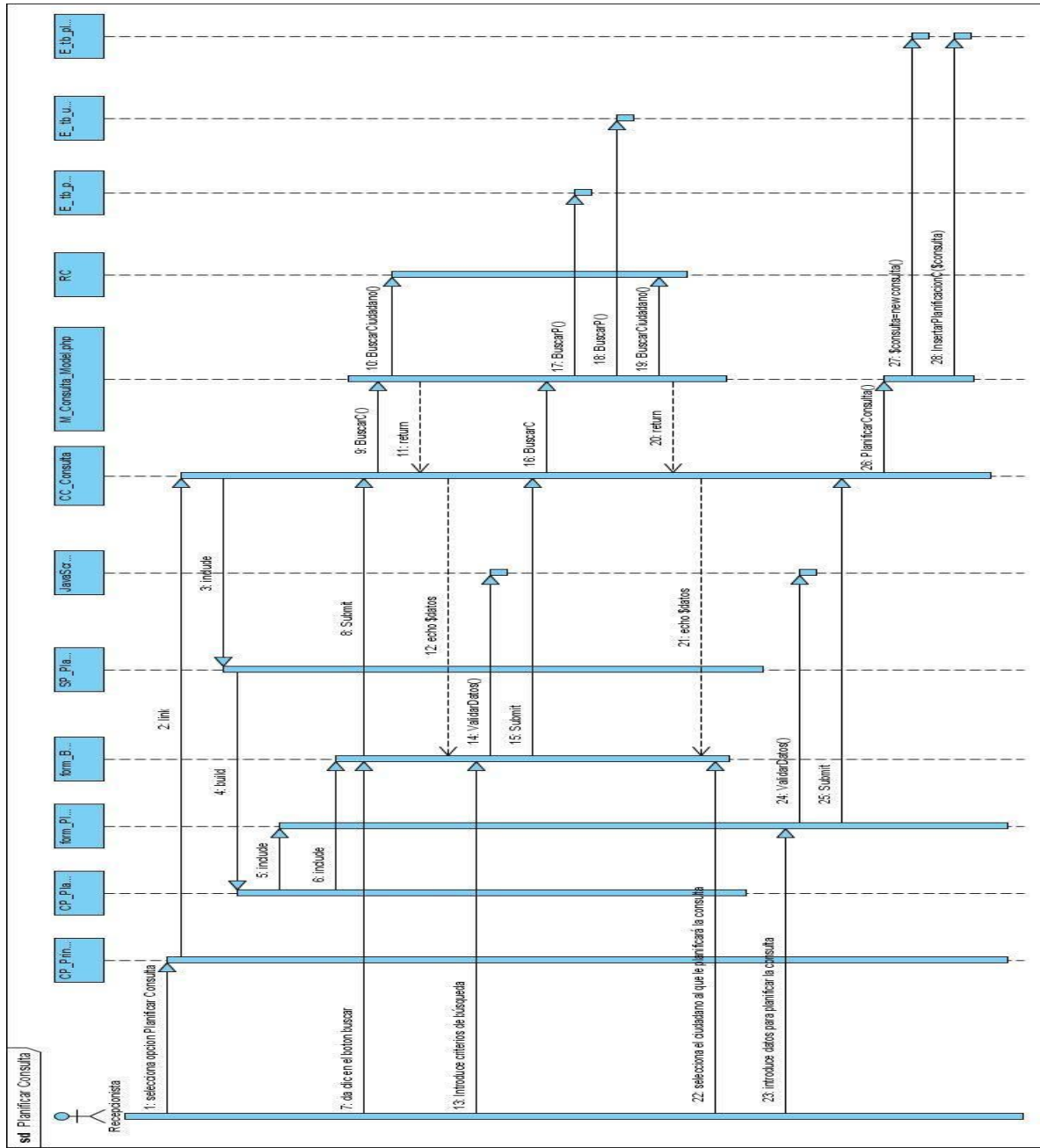








Diagrama de Secuencia Planificar Consulta



### Diagrama de Secuencia Dar Asistencia a Tratamiento

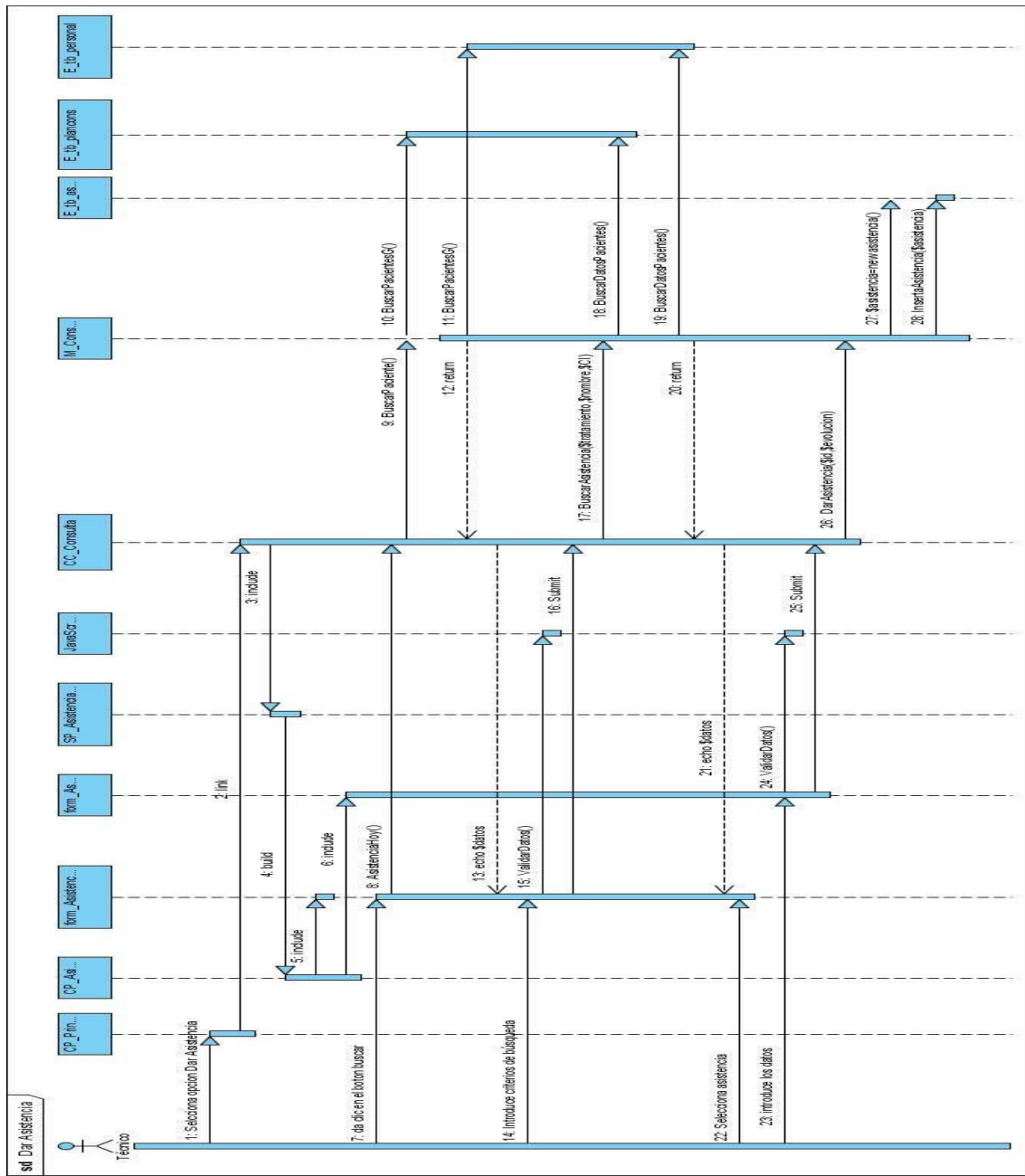
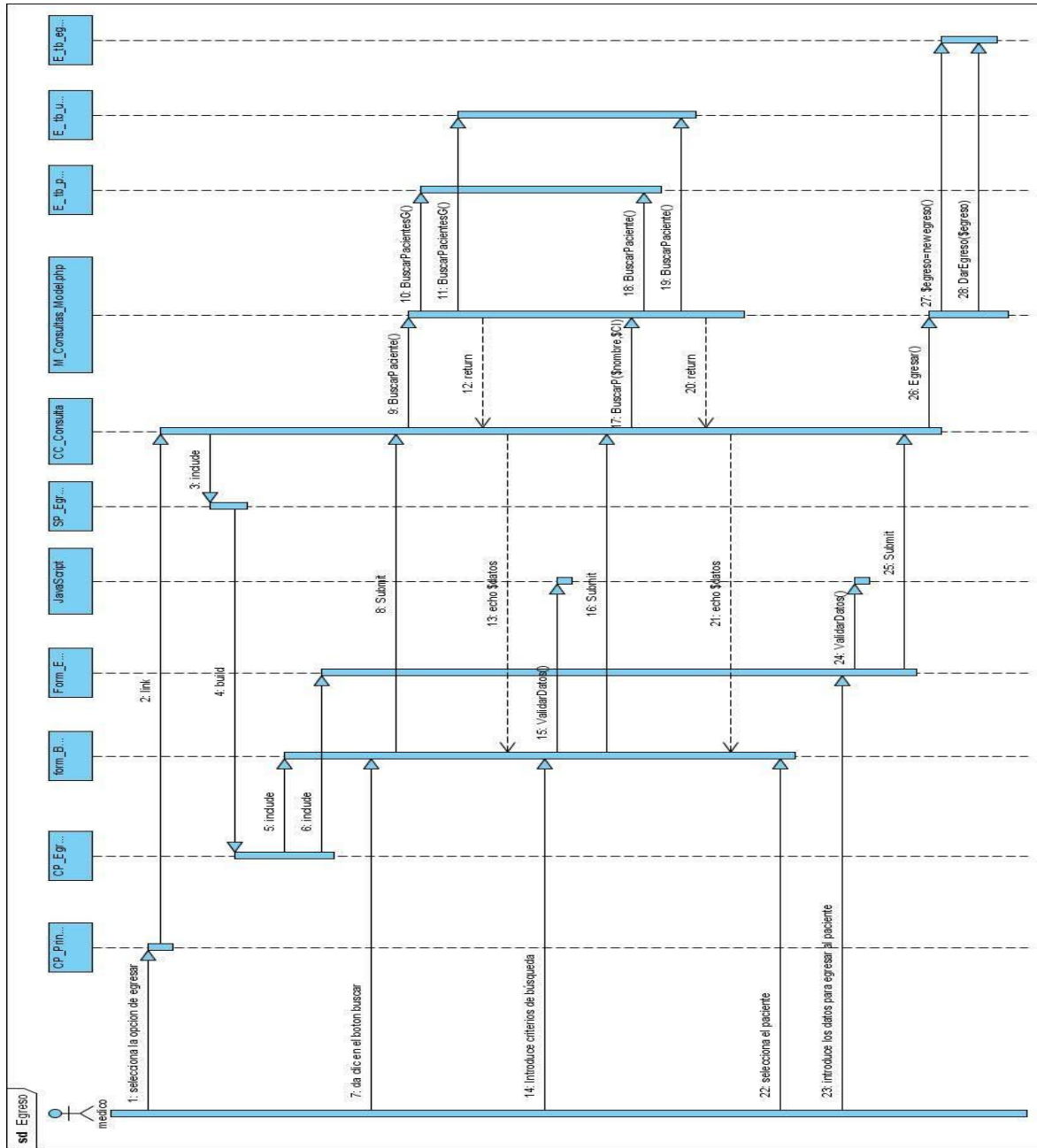
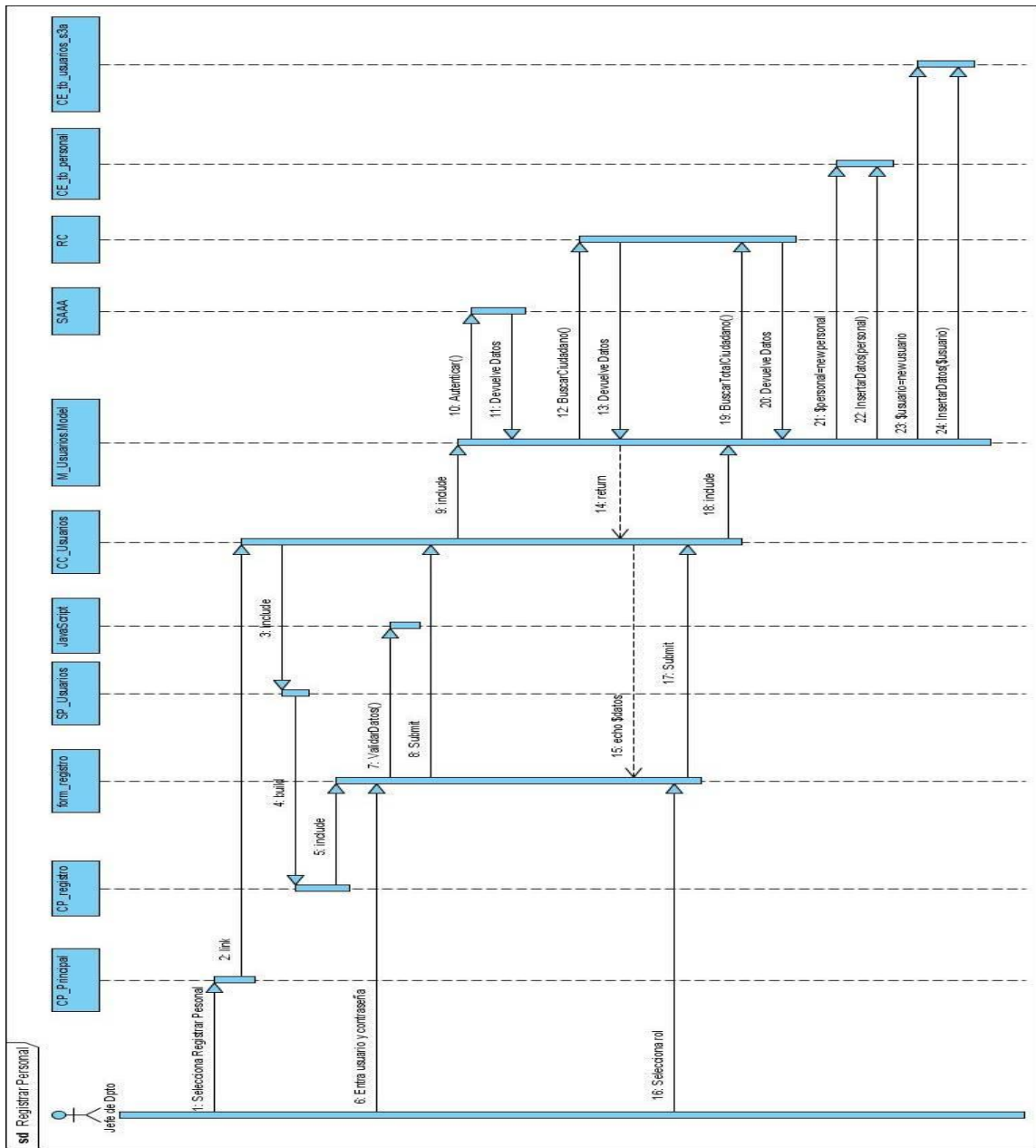




Diagrama de Secuencia Gestionar Egreso



### Diagrama de Secuencia Registrar Personal



Anexo 2 Diagrama de Clases del Diseño

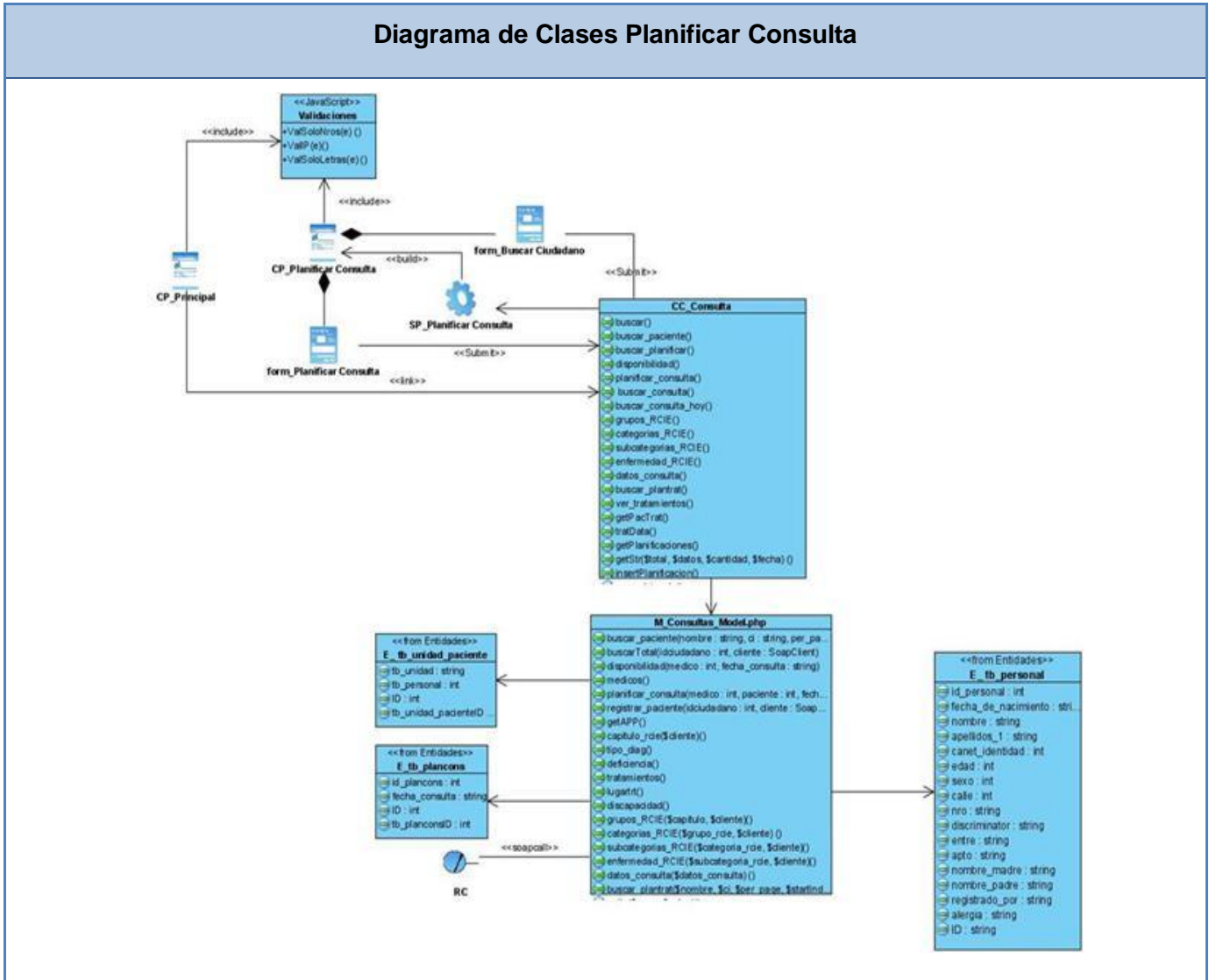


Diagrama de Clases Dar Asistencia a Tratamiento

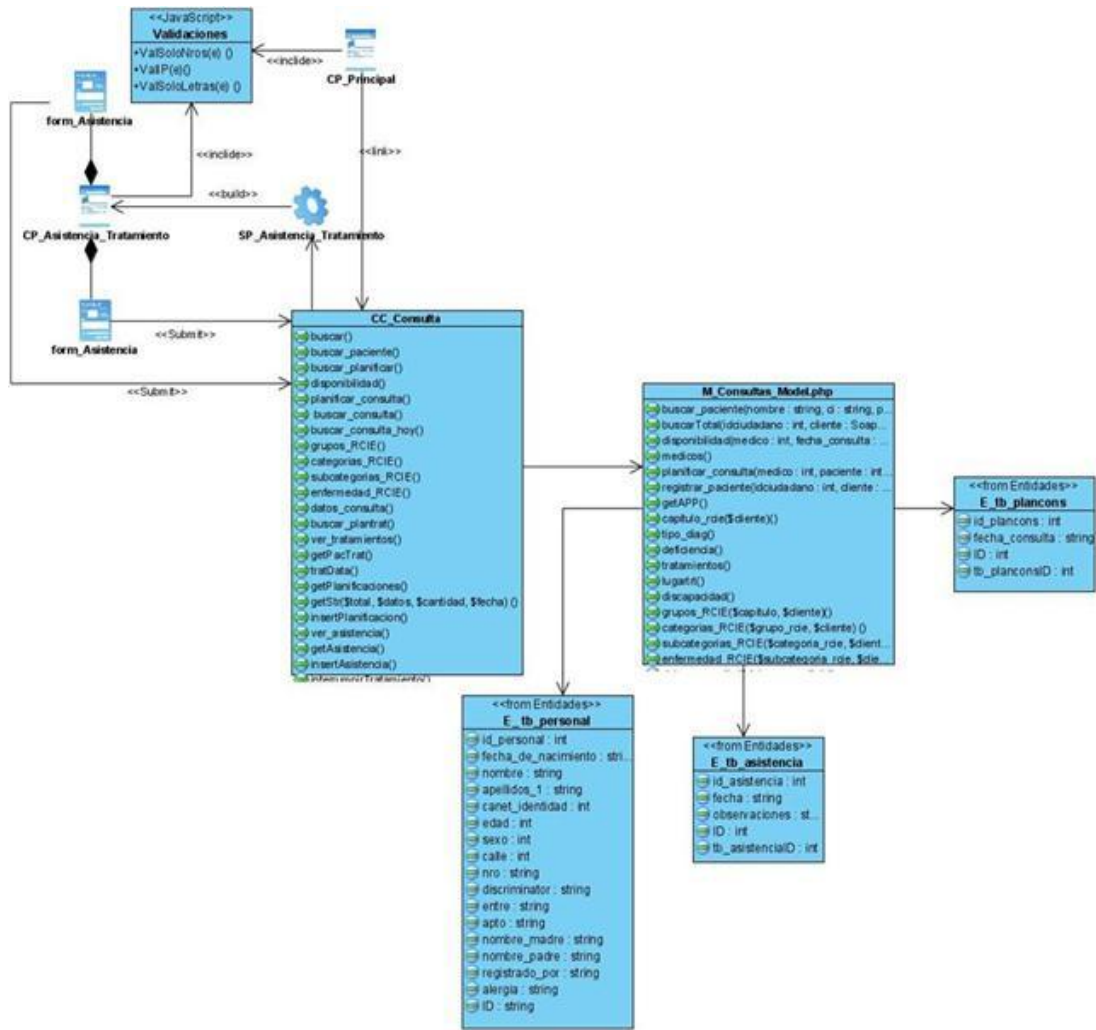




Diagrama de Clases Realizar Consulta

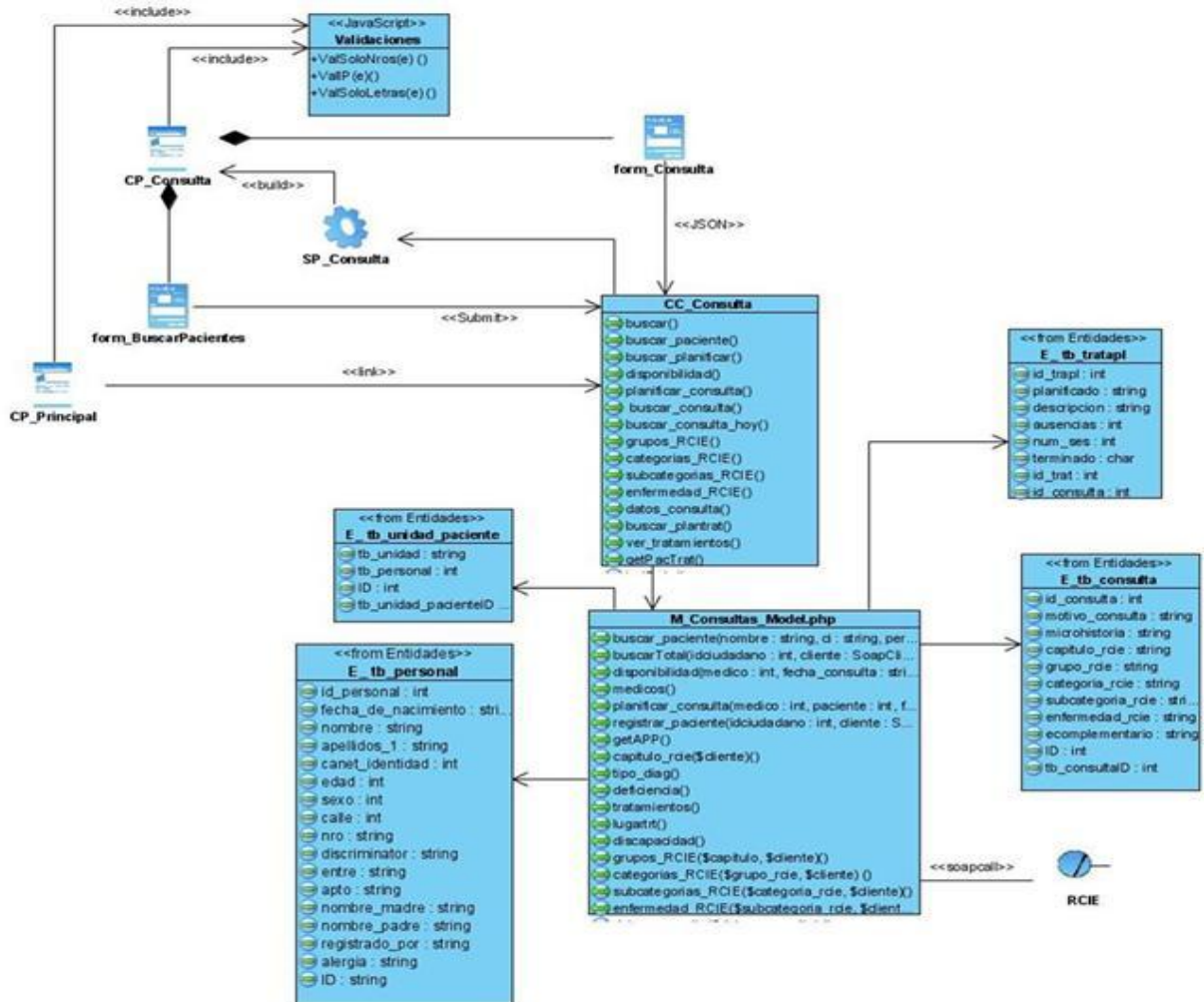
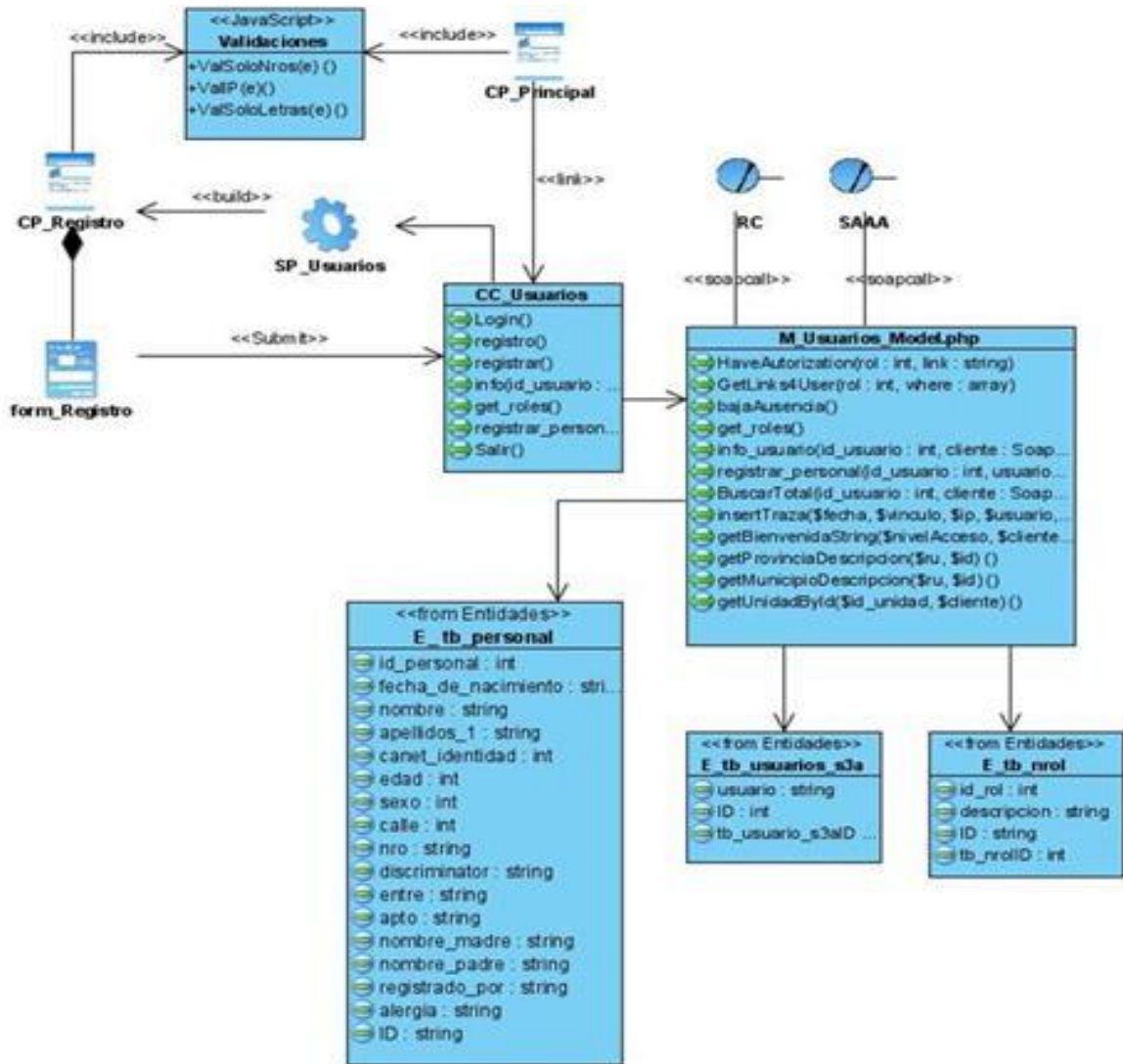






Diagrama de Clases Registrar Personal



### GLOSARIO DE TÉRMINOS

**Apache:** es un servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etcétera), Windows y otras, que implementa el protocolo HTTP/1.1.

**Aplicación Web:** especialización de las aplicaciones Cliente/Servidor, están comúnmente estructuradas como una aplicación en tres capas, son conocidas como aplicaciones servidor.

**Arquitectura Cliente/Servidor:** es un modelo para el desarrollo de sistemas de información, en el que las transacciones se dividen en elementos independientes que cooperan entre sí para intercambiar información, servicios o recursos.

**Framework:** en el desarrollo de software, es una estructura de soporte definida, mediante la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, puede incluir soporte de programas, bibliotecas y un lenguaje interpretado entre otros software para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

**CodeIgniter:** es un Framework para desarrollo de aplicaciones en PHP.

**Casos de Usos (CU):** es una secuencia de interacciones que se desarrollarán entre un sistema y sus actores en respuesta a un evento que inicia un actor principal sobre el propio sistema.

**DHTML:** Dynamic HTML/HTML Dinámico.

**Egreso:** dar de alta a un paciente.

**Fisioterapia:** se ocupa solo de la rehabilitación física del paciente.

**HTML:** HyperText Markup Language/Lenguaje Hipertexto de Marcado. Lenguaje usado para escribir documentos para servidores World Wide Web. Es una aplicación de la ISO Standard 8879:1986. Es un lenguaje de marcas. Los lenguajes de marcas no son equivalentes a los lenguajes de programación aunque se definan igualmente como "lenguajes". Son sistemas complejos de descripción de información, normalmente documentos, que se pueden controlar desde cualquier editor ASCII.

**HTTP:** HyperText Transfer Protocol/ Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas Web.

**Hardware:** Componentes electrónicos, tarjetas, periféricos y equipo que conforman un sistema de computación; se distinguen de los programas (software) porque son tangibles.

**Informática:** disciplina que estudia el tratamiento automático de la información utilizando dispositivos electrónicos y sistemas computacionales.

**Internet:** Sistema de redes de computación ligadas entre sí, con alcance mundial, que facilita servicios de comunicación de datos como registro remoto, transferencia de archivos, correo electrónico y grupos de noticias. Internet es una forma de conectar las redes de computación existentes que amplía en gran medida el alcance de cada sistema participante.

**Linux:** Es el nombre de un núcleo, pero se suele denominar con este nombre a un sistema operativo de libre distribución software libre (y de código abierto), donde el código fuente está disponible públicamente y cualquier persona, con los conocimientos informáticos adecuados, puede libremente estudiarlo, usarlo, modificarlo y redistribuirlo.

Ofrece una completa solución abierta para las tecnologías Web y estándares de hoy, incluyendo la accesibilidad y servicios Web.

**Medicina Física:** especialidad médica que desarrolla una serie de medidas de promoción de salud, prevención, diagnóstico, investigación y tratamiento, para obtener la máxima reincorporación posible a la comunidad de los pacientes con secuelas invalidantes, congénitas o adquiridas.

**Microsoft:** Compañía que manufactura los sistemas de operación DOS y Windows.

**MINSAP:** Ministerio de Salud Pública

**MySQL:** Es un sistema de gestión de bases de datos relacional que cuentan con todas las características de un motor de BD comercial: transacciones atómicas, triggers, replicación, llaves foráneas entre otras. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar.

**MVC:** Modelo Vista Controlador.

**Paquete:** mecanismo de propósito general para organizar elementos en grupos.

**PHP:** Hypertext Preprocessor/Preprocesador de Hipertexto. Es un lenguaje script del lado del servidor que permite crear y ejecutar aplicaciones Web dinámicas e interactivas. Con PHP se pueden combinar páginas HTML y scripts. Con el objetivo de crear aplicaciones potentes.

**PostgreSQL:** es un Sistema de Gestión de Bases de Datos Objeto-Relacionales (ORDBMS) libre.

**Personal Médico:** son los médicos, recepcionistas y técnicos de salud que laboran en los centros rehabilitación.

**Rehabilitación:** recuperación física, psíquica, social y laboral, es decir, la rehabilitación es la recuperación global del enfermo o lesionado.

**RUP:** Rational Unified Process (Proceso Unificado de Desarrollo). Metodología para el desarrollo de Software.

**Software:** es el conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

**Software Libre:** es el software que una vez copiado puede ser modificado, cambiado, mejorado y redistribuido libremente.

**Sitio Web:** es un conjunto de páginas web, típicamente comunes a un dominio de Internet o subdominio en la World Wide Web en Internet.

**SGBD:** Sistema de Gestión de Bases de Datos. Es el software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez.

**SOAP:** Simple Object Access Protocol /Protocolo Estándar de Acceso a Objetos

**UML:** Unified Modeling Language/Lenguaje Unificado de Modelado. Es una notación estándar para modelar objetos del mundo real como primer paso en el desarrollo de programas orientados a objetos. Es un lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de software.

**WEB (WWW):** red de documentos HTML intercomunicados y distribuidos entre servidores del mundo entero.

