

The logo for Universidad de las Ciencias Informáticas (UCi) features the letters 'UCi' in a bold, blue, sans-serif font. A thin blue arc is positioned above the 'i'.

**Universidad de las Ciencias Informáticas**

# Gedont

**Sistema** Integrado para la **Gestión Estomatológica**

## **Autores**

Ramón Torres Méndez  
Yunier Albrecht Delgado

## **Tutor**

Ing. Arian Cabezas Regal

Ciudad de La Habana, junio 20 | 2007 "Año 49 de la Revolución"

**TRABAJO PARA OPTAR POR EL TÍTULO DE INGENIERO EN CIENCIAS INFORMÁTICAS**

*“Nunca consideres el estudio como un deber, sino como una oportunidad para penetrar en el maravilloso mundo del saber.”*

*Albert Einstein*

## DEDICATORIA

**De Ramón:** a mi familia por ser la fuente de mi existencia, por estar siempre a mi lado y apoyarme en los momentos más difíciles, sin ustedes no se hubieran hecho posible mis sueños.

A mi queridísima madre por ese amor tan grande que me ha brindado siempre.

A mi padre por ser el mejor de todos los padres del mundo, que con su sabiduría ha sabido guiarme.

A mi hermano que ha sido mi padre, mi madre, mi hermano, mi amigo y el mejor consejero que he tenido.

**De Yunier:** A mi familia por todo el apoyo y la confianza brindados a lo largo de mi vida estudiantil, en especial a mis padres y mi hermana, que sin ellos nada de esto hubiera sido posible.

A mi padre por ser mi ejemplo a seguir en la vida, por sus consejos y enseñanzas; y por estar siempre seguro de mí.

A mi madre por todo el amor del mundo y por estar siempre ahí.

A mi hermana, amiga y confidente a la cual quiero con el corazón.

## AGRADECIMIENTOS

A nuestro tutor Arian por esa gran ayuda que nos ha brindado en todo momento, a todos los profesores que han contribuido a nuestra educación y formación como profesionales.

A nuestro invencible Comandante en Jefe por darnos la oportunidad de estudiar en esta maravillosa Universidad, por soñar y hacer realidad nuestros sueños.

### **De Ramón:**

Agradezco a toda mi familia y principalmente a mi madre, mi padre y mi hermano por ser mis tutores de toda la vida.

A todos aquellos que de una manera u otra han aportado un granito de arena para la realización de este trabajo y especialmente a mis queridos amigos del grupo y proyecto UCITeVe.

### **De Yunier:**

Agradezco a toda mi familia por apoyarme y confiar en mí, especialmente a mis padres y mi hermana por estar siempre a mi lado y no faltarme nunca.

A mis amigos por su amistad y su ayuda, en especial a mis grupos de clases a los cuales agradezco por haber estado ahí junto a mí y compartir todos juntos los buenos y malos ratos que han hecho de estos 5 años los mas especiales de mi vida. A todo el pueblo de UCITeVe por todas las cosas y los momentos que pasamos juntos, no habría sido lo mismo sin ustedes.

## DECLARACIÓN DE AUTORÍA

Declaramos que somos los únicos autores de este trabajo y autorizamos a la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Autor(es): Ramón Torres Méndez

\_\_\_\_\_

Yunier Albrecht Delgado

\_\_\_\_\_

Tutor(a): Ing. Arian Cabezas Regal

\_\_\_\_\_

## OPINIÓN DEL TUTOR

**Título:** Sistema Integrado para la Gestión Estomatológica

**Autores:** Ramón Torres Méndez y Yunier Albrecht Delgado

En el periodo de elaboración de la Tesis Sistema Integrado para la Gestión Estomatológica los estudiantes Ramón Torres Méndez y Yunier Albrecht Delgado han demostrado poseer el conocimiento y la responsabilidad necesaria para desarrollar esta aplicación con la calidad y prestaciones requeridas, sus habilidades y profesionalidad los dotan de las cualidades imprescindibles para ejercer la futura profesión de Ingenieros Informáticos. El resultado de su trabajo ya ha sido validado en el centro para el cual fue diseñado e implementado, lo que avala la excelencia del trabajo realizado, lo anterior les confiere por derecho propio la certificación de lo aprendido en los años de estudio, por lo que se le propone una calificación de 5 ptos...

Tutor: Ing. Arian Cabezas Regal

\_\_\_\_\_

Fecha: \_\_\_\_\_

## RESUMEN

Con el alto desarrollo de la informática y las comunicaciones, son cada vez más las organizaciones e instituciones que requieren del uso de aplicaciones que gestionen su información. La Universidad de las Ciencias Informáticas es un ejemplo claro de esto, debido a la cifra elevada de personas existente y que seguirá incrementando con el transcurso de los años se necesita de un buen sistema de gestión estomatológica que permita mejorar la salud bucal.

En este trabajo se presenta la descripción del Sistema de Gestión Estomatológica **GEDONT**, desde la óptica de sus funcionalidades y aplicaciones, se brinda una solución automatizada a todos los procesos de gestión estomatológicos que actualmente se realizan de forma manual en la UCI.

Para dar cumplimiento a los procesos que se automatizan se realizó un estudio de cómo estructurar el producto final y se determinó hacerlo de la siguiente manera: una solución WEB debido a que se necesita que el sistema sea accedido remotamente por los usuarios, o por los administradores según se quiera y un Gestor de Base de Datos para el almacenamiento de la información que se quiere procesar, así como para almacenar el resultado del procesamiento de esta y su fácil acceso.

La fundamentación del trabajo se basa en el conocimiento teórico-práctico de dos personas, que para lograr esta aplicación se han apoyado en su experiencia tanto en el desarrollo de sistemas similares como en las actividades diarias dentro de una Consulta Estomatológica.

# ÍNDICE

<b>INTRODUCCIÓN</b> .....	1
<b>CAPÍTULO 1 Fundamentación Teórica</b> .....	6
1.1 Introducción.....	6
1.2 Objeto de estudio .....	6
1.2.1 Descripción General.....	6
1.2.2 Descripción actual del dominio del problema.....	7
1.2.3 Situación problemática.....	8
1.2.4 Análisis de otras soluciones existentes.....	8
1.3 Conclusiones.....	10
<b>CAPITULO 2 Tendencias y Tecnologías Actuales</b> .....	11
2.1 Introducción.....	11
2.2 Las aplicaciones Web.....	11
2.3 Modelo Cliente Servidor.....	13
2.4 Lenguajes de Programación Web .....	15
2.4.1 PHP.....	15
2.5 Servidores Web.....	20
2.5.1 Apache .....	21
2.6 Sistema de Gestión de Bases de Datos (SGBD).....	23
2.6.1 MySQL .....	24
2.7 Proceso de Desarrollo de Software.....	27
2.8 Lenguaje Unificado de Modelado (UML) .....	30
2.9 Conclusiones.....	33
<b>CAPITULO 3 Presentación de la Solución Propuesta</b> .....	34
3.1 Introducción.....	34
3.2 Modelo de Negocio.....	34
3.2.1 Procesos de Negocio.....	34
3.2.2 Reglas del Negocio.....	35
3.2.3 Actores del negocio .....	36
3.2.4 Trabajadores del negocio.....	36
3.2.5 Diagrama de Casos de Uso del Negocio .....	37
3.2.6 Descripción textual de los Casos de Uso de Negocio .....	39
3.2.6.1 Caso de Uso “Asignar Turnos” .....	39
3.2.6.2 Caso de uso “Asignar Turnos (PU)”.....	40
3.2.6.3 Caso de uso “Atender Pacientes” .....	40
3.2.6.4 Caso de uso “Indicar Medicamentos” .....	41
3.2.6.5 Caso de uso “Remitir a Especialista” .....	42
3.2.6.6 Caso de Uso “Remitir a Ingreso” .....	43
3.2.7 Diagramas de Actividades para cada Caso de Uso del Negocio.....	44
3.2.7.1 Diagrama de Actividades para el CUN “Asignar Turnos” .....	45
3.2.7.2 Diagrama de Actividades para el CUN “Asignar Turnos PU” .....	46
3.2.7.3 Diagrama de Actividades para el CUN “Atender Paciente”.....	47
3.2.7.4 Diagrama de Actividades para el CUN “Indicar Medicamentos” .....	48



3.2.7.5	Diagrama de Actividades para el CUN “Remitir a Especialista” .....	49
3.2.7.6	Diagrama de Actividades para el CUN “Remitir a Ingreso” .....	50
3.3	Requerimientos Funcionales.....	51
3.4	Requerimientos No Funcionales .....	53
3.5	Descripción del Sistema Propuesto.....	55
3.5.1	Descripción de los actores del sistema.....	55
3.5.2	Casos de Uso del Sistema .....	56
3.5.2.1	Paquete Seguridad .....	57
3.5.2.2	Paquete Administrador .....	57
3.5.2.3	Paquete Jefe de Departamento.....	58
3.5.2.4	Paquete Recepcionista .....	58
3.5.2.5	Paquete Estomatólogo .....	59
3.5.3	Descripción textual de casos de usos del sistema .....	60
3.5.3.1	Caso de Uso “Autenticar Usuarios” .....	60
3.5.3.2	Caso de Uso “Gestionar Usuarios” .....	61
3.5.3.3	Caso de Uso “Gestionar Reservación” .....	63
3.5.3.4	Caso de Uso “Visualizar Turnos Reservados” .....	65
3.5.3.5	Caso de Uso “Gestionar Historias Clínicas” .....	66
3.5.3.6	Caso de Uso “Gestionar HC - Dentigrama” .....	70
3.5.3.7	Caso de Uso “Gestionar HC - Evolución” .....	72
3.5.3.8	Caso de Uso “Generar Parte Diario” .....	74
3.6	Conclusiones.....	76
	<b>CAPÍTULO 4 Construcción de la Solución Propuesta .....</b>	<b>77</b>
4.1	Introducción.....	77
4.2	Diagramas de Clases .....	77
4.2.1	Diagrama de Clases Web del CUS “Autenticar Usuario” .....	78
4.2.2	Diagrama de Clases Web del CUS “Gestionar Usuario” .....	79
4.2.3	Diagrama de Clases Web del CUS “Gestionar Reservación” .....	80
4.2.4	Diagrama de Clases Web del CUS “Gestionar HC” .....	81
4.2.4.1	Diagrama de Clases Web del CUS Gestionar HC. “Sección Crear HC” .....	81
4.2.4.2	Diagrama de Clases Web del CUS Gestionar HC. “Sección Modificar HC” .....	82
4.2.4.3	Diagrama de Clases Web del CUS Gestionar HC. “Sección Eliminar HC” .....	83
4.2.4.4	Diagrama de Clases Web del CUS Gestionar HC. “Sección Visualizar HC” .....	84
4.2.5	Diagrama de Clases Web del CUS “Gestionar Dentigrama” .....	85
4.2.6	Diagrama de Clases Web del CUS “Gestionar Evolución” .....	86
4.2.7	Diagrama de Clases Web del CUS “Visualizar Turnos Reservados” .....	87
4.2.8	Diagrama de Clases Web del CUS “Generar Parte Diario” .....	88
4.2.9	Diagrama de Clases Web del CUS “Generar Parte Gasto Material” .....	89
4.3	Principios de diseño.....	90
4.3.1	Estándares de la interfaz de la aplicación.....	91
4.3.2	Estándares de Codificación.....	92
4.3.3	Tratamiento de Errores .....	93
4.4	Diseño de la Base de Datos.....	93
4.4.1	Diagrama de Clases Persistentes .....	94

4.4.2	Modelo de Datos.....	95
4.5	Generalidades de la Implementación .....	96
4.6	Modelo de Despliegue .....	97
4.7	Modelo de Implementación.....	98
4.8	Conclusiones.....	100
<b>CAPÍTULO 5 Estudio de Factibilidad .....</b>		<b>101</b>
5.1	Introducción.....	101
5.2	Planificación basada en casos de usos.....	101
5.3	Costos.....	106
5.4	Beneficios Tangibles e Intangibles .....	106
5.5	Análisis de costos y beneficios.....	107
5.6	Conclusiones.....	107
<b>CONCLUSIONES.....</b>		<b>108</b>
<b>RECOMENDACIONES.....</b>		<b>110</b>
<b>BIBLIOGRAFÍA.....</b>		<b>111</b>
<b>GLOSARIO DE TÉRMINOS.....</b>		<b>114</b>
<b>ANEXOS.....</b>		<b>116</b>

## ÍNDICE DE TABLAS

Tabla 1. Descripción de los actores del negocio .....	36
Tabla 2. Descripción de los trabajadores del negocio .....	37
Tabla 3. Descripción textual del caso de uso “Asignar Turnos” .....	39
Tabla 4. Descripción textual del Caso de uso “Asignar Turnos (PU)” .....	40
Tabla 5. Descripción textual Caso de Uso “Atender Pacientes” .....	41
Tabla 6. Descripción textual del Caso de Uso “Indicar Medicamentos” .....	42
Tabla 7. Descripción textual del Caso de Uso “Remitir a Especialista” .....	43
Tabla 8. Descripción textual del Caso de Uso “Remitir a Ingreso” .....	43
Tabla 9. Descripción de los Actores del Sistema .....	56
Tabla 10. Descripción detallada del Caso de Uso del sistema “Autenticar Usuarios” .....	60
Tabla 11. Descripción detallada del caso de Uso del sistema “Gestionar Usuario” .....	63
Tabla 12. Descripción detallada del caso de Uso del sistema “Gestionar Reservación” .....	65
Tabla 13. Descripción detallada del caso de Uso del sistema “Visualizar Turnos Reservados” .....	66
Tabla 14. Descripción detallada del caso de Uso del sistema “Gestionar Historias Clínicas” .....	69
Tabla 15. Descripción detallada del caso de Uso del sistema “Gestionar HC - Dentigrama” .....	72
Tabla 16. Descripción detallada del caso de Uso del sistema “Gestionar HC – Evolución” .....	74
Tabla 17. Descripción detallada del caso de Uso del sistema “Generar Parte Diario” .....	75
Tabla 18. Factor de peso de los actores sin ajustar .....	101
Tabla 19. Factor de peso de los casos de uso sin ajustar .....	102
Tabla 20. Factor de complejidad técnica .....	103
Tabla 21. Factor de ambiente .....	104
Tabla 22. Esfuerzo del Proyecto .....	105
Tabla 23. Precios de PC por versiones .....	117
Tabla 24. Descripción detallada del CUS “Remitir a Especialista” .....	119
Tabla 25. Descripción detallada CUS “Indicar Medicamentos” .....	120
Tabla 26. Descripción detallada del CUS “Remitir a Ingreso” .....	121
Tabla 27. Descripción detallada del CUS “Parte Gasto Material” .....	122
Tabla 28. Descripción detallada del CUS “Gestionar Materiales” .....	124
Tabla 29. Descripción detallada del CUS “Visualizar Materiales” .....	125

## ÍNDICE DE FIGURAS

Figura 1. Diagrama de Casos de Usos del Negocio.....	38
Figura 2. Diagrama de actividades del Caso de Uso "Asignar Turno" .....	45
Figura 3. Diagrama de actividades del caso de uso "Asignar Turno PU" .....	46
Figura 4. Diagrama de actividades del Caso de Uso "Atender Pacientes".....	47
Figura 5. Diagrama de actividades del Caso de Uso "Indicar Medicamentos" .....	48
Figura 6. Diagrama de actividades del Caso de Uso "Remitir a Especialista" .....	49
Figura 7. Diagrama de actividades del Caso de Uso "Remitir a Ingreso".....	50
Figura 8. Paquetes del Modelo de Casos de Usos del Sistema .....	56
Figura 9. Caso de uso asociado al Paquete de Seguridad del Sistema.....	57
Figura 10. Caso de uso asociado Paquete de Administrador del Sistema.....	57
Figura 11. Casos de uso asociado al Paquete de Jefe de Departamento del Sistema .....	58
Figura 12. Casos de uso asociado al Paquete Recepcionista del Sistema.....	58
Figura 13. Casos de uso asociado al Paquete Estomatólogo del Sistema .....	59
Figura 14. Diagrama de Clases del Diseño del CUS Autenticar Usuario .....	78
Figura 15. Diagrama de Clases del Diseño del CUS Gestionar Usuarios.....	79
Figura 16. Diagrama de Clases del Diseño del CUS Gestionar Reservación .....	80
Figura 17. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Crear HC .....	81
Figura 18. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Modificar HC .....	82
Figura 19. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Eliminar HC.....	83
Figura 20. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Visualizar HC .....	84
Figura 21. Diagrama de Clases del Diseño del CUS Gestionar Dentigrama.....	85
Figura 22. Diagrama de Clases del Diseño del CUS Gestionar Evolución.....	86
Figura 23. Diagrama de Clases del Diseño del CUS Visualizar Turnos Reservados .....	87
Figura 24. Diagrama de Clases del Diseño del CUS Generar Parte Diario.....	88
Figura 25. Plantilla del Sistema en General .....	92
Figura 26. Diagrama de Clases Persistentes.....	94
Figura 27. Diagrama de Modelo de Datos .....	95
Figura 28. Diagrama de Despliegue .....	98
Figura 29. Diagrama de Implementación .....	99
Figura 30. Vista general de RUP.....	117
Figura 31. Flujos de trabajo de RUP .....	118
Figura 32. Diploma por haber obtenido la condición de Relevante en la 4ta Jornada Científica, Policlínico "Ernesto Che Guevara" .....	126

## INTRODUCCIÓN

El manejo de la información en el mundo actual se convierte en premisa fundamental para el logro de la excelencia en los más diversos sectores de la sociedad, esta información en muchos de los casos tiende a ser voluminosa, desorganizada, compleja en cuanto a su interpretación y sobre todo muy lento el poder de procesamiento manual sobre la misma.

Conocido es por todas las capacidades de cómputo que poseen los ordenadores modernos, capaces de realizar cálculos millones de veces más rápido que la más poderosa mente humana, aprovechando estas funcionalidades y revirtiéndolas en nuestros intereses se logran realizar sistemas automatizados que resuelven la problemática actual del manejo de la información.

La información que se procesa en una Consulta Estomatológica no queda exenta de este mal, numerosos son los procesos que se realizan sobre la misma relacionada a un paciente determinado, o a un conjunto de materiales (dentales y de oficina), o a la documentación relacionada con los diversos tipos de enfermedades y los tratamientos correspondientes que se aplican, o simplemente a la distribución de turnos por pacientes y su correspondencia con algún Doctor, o a la publicación de artículos educativos con propósitos preventivos y el posible estudio de los indicadores que influyen en las patologías bucales reincidentes de los pacientes del universo que analiza, por solo citar algunos.

Partiendo del conocimiento de la información que se procesa en una Consulta Estomatológica se realizó un estudio investigativo donde se determinó el área problemática, se identificaron elementos y se presentaron soluciones para el diseño del sistema de gestión de información de la clínica. Al realizar esta investigación se utilizaron varios métodos de análisis donde se presentaron soluciones adecuadas para el manejo de la información.

La Universidad de las Ciencias Informáticas crece en población cada año por lo que es necesario garantizar un conjunto de condiciones primarias de vida. Como parte de la ciudad digital a la que pertenecen estos servicios no pueden estar al margen del desarrollo cotidiano e incremental de la UCI, por lo que se hace inminente la automatización de los procesos de gestión y almacenamiento de datos referentes a los pacientes, logrando con esto facilidad de almacenamiento y ligereza en el trabajo.

Cada curso con el incremento notable de matrícula se hace más difícil el trabajo de organización, control y atención de las personas que cada año van formando parte de esta universidad. Es por lo que se nos

plantea la necesidad de implementación de un sistema en el área estomatológica del Hospital - Policlínico de la UCI, facilitando así el trabajo del personal médico y elevando la calidad de vida de los pacientes con una mejor y más rápida atención.

En estos momentos la labor del personal médico esta matizada por un número elevado de documento que repite información del paciente en uno y otro modelo, esto se facilitaría grandemente si a la hora de llenar los mismos la información que ya se conoce del paciente apareciera en el modelo, solo faltando la que de verdad no se sabía hasta el momento; así mismo todavía es necesario abandonar o demorar las consultas por gestiones de solicitar documentos como historias clínicas, que con este sistema se verían reducidas a simples acciones de búsquedas, solicitudes o preguntas en el ordenador.

En la UCI todos estos procesos se realizan de forma manual y dado el alto volumen de estudiantes, profesores, trabajadores existentes se hace muy complejo el manejo de la información generada por los pacientes en las consultas estomatológicas realizadas en el trabajo diario del personal médico, por este motivo se hace necesario implementar mecanismos que permitan una mejor organización de la información que se procesa en las consultas estomatológicas para así lograr mejores resultados.

Este proyecto pudiera servir como base para otros de este tipo en nuestro país. El resto de las universidades, con características similares a la UCI podrían también poner en práctica un sistema de este tipo que mejore la calidad de vida de todos sus estudiantes y trabajadores además de contribuir a la informatización del país, que tiene su punto de partida en esta universidad. Además de beneficiar en algunos casos a las comunidades que rodean a las universidades alejadas de la ciudad.

Lo esperado de este trabajo como aplicación práctica es brindar al cliente la posibilidad de interactuar con un sistema que le propicie un conjunto de servicios equivalentes a los procesos que se realizan en la actualidad además de encargarse de la generación de documentos, como constancia de la ocurrencia de estos procesos incluyendo en ellos los posibles documentos adjuntos y resultados almacenados, además de viabilizar y agilizar el trabajo de del los profesionales, conocer los distintos indicadores de salud de los pacientes atendidos.

En general el **problema** que surge de esta investigación a raíz de las situaciones expuestas es la inexistencia en la UCI de una aplicación que dé respuesta a las necesidades de gestión de información requeridas en el servicio de odontología lo que lleva de manera natural a la conclusión de la necesidad de

utilización de una herramienta automatizada que contribuya a aumentar la calidad en los servicios en este ámbito, lo cual es el propósito de este trabajo.

La aplicación informática que se propone desarrollar en esta Tesis, se organiza en torno a la clínica de estomatología de la Universidad de las Ciencias Informáticas, que tiene como eje principal al paciente (estudiantes, profesores, trabajadores), en el área de consultas.

Los aportes prácticos esperados del trabajo son:

- Centralización de toda la información de los servicios brindados.
- Rapidez en la búsqueda de información por parte del personal médico.
- Actualización rápida de la información.

El **objeto de estudio** lo constituyen los diferentes procesos que se desarrollan en la UCI durante la realización de las consultas estomatológicas diarias para garantizar una excelente salud bucal del personal de nuestra comunidad universitaria.

El **campo de acción** que abarca este trabajo es la automatización de la gestión de información que surge como resultado del flujo de trabajo durante una consulta estomatológica en la Universidad de las Ciencias Informáticas.

Lo planteado anteriormente lleva a la formulación de los siguientes objetivos:

**Objetivo General:**

- Desarrollo de un Sistema Integrado de Gestión Odontológica mediante el uso de herramientas de software libre que permita una mejor organización del flujo de información en una consulta estomatológica.

A partir de un análisis del objetivo general se derivan los siguientes objetivos específicos para la aplicación:

**Objetivos Específicos:**

- Realizar el análisis del Sistema Integrado para la gestión estomatológica.
- Desarrollar el diseño del Sistema Integrado para la gestión estomatológica.
- Diseñar una base de datos capaz de almacenar de forma organizada la información.

- Implementar el Sistema Integrado para la gestión estomatológica.

Para cumplir los objetivos se desarrollaron las siguientes **tareas**:

- Estudio del arte de desarrollo de las tecnologías utilizadas para el diseño de aplicaciones web y Base de Datos.
- Estudio y análisis del funcionamiento actual de las consultas estomatológicas realizadas en la UCI.
- Entrevistas a trabajadores del departamento de estomatología del Centro Médico de la Universidad para identificar las necesidades de cada proceso.
- Confección de la documentación completa UML del análisis y diseño del sistema.
- Desarrollo de la Base de datos que soporte todas las funcionalidades del sistema.
- Implementación de un prototipo funcional del Sistema Integrado para la gestión estomatológica.

El documento está estructurado en cinco capítulos, que incluye todo lo relacionado con el trabajo investigativo realizado, así como el análisis y el diseño del sistema que se propone.

Capítulo I. Fundamentación Teórica, incluye el análisis de la información existente acerca del tema a tratar a nivel internacional, nacional, de la Universidad y se describe el objeto de estudio en que se desarrolla al sistema.

Capítulo II. Tendencias y tecnologías actuales, incluye el análisis de la información existente acerca de las tendencias y tecnologías actuales que existen en el mundo.

Capítulo III. Presentación de la solución propuesta, se describe el objeto de estudio en términos de negocio, el entorno de trabajo en que se desarrolla el sistema, se especifican los detalles de la construcción del sistema, la propuesta del sistema, requerimientos funcionales y no funcionales y los casos de uso del sistema.

Capítulo IV. Construcción de la solución propuesta, aborda aspectos relacionados con la definición de los modelos de análisis de diseño, se modelan los diagramas de clases web, se plantea el modelo de datos y se describe la distribución física del sistema mediante el modelo de despliegue y el modelo de componentes.



Capitulo V. Estudio de Factibilidad, se realiza la planificación basada en casos de uso, se determina el costo y se hace un breve análisis de los costos y beneficios del proyecto.

# CAPÍTULO 1

## Fundamentación Teórica

### 1.1 Introducción

Este capítulo aborda aspectos generales de la estomatología y las consultas estomatológicas. Se brinda una explicación detallada de los aspectos relacionados sobre procesos que se llevan a cabo dentro de una consulta estomatológica y en específico las consultas del Departamento de Estomatología de la UCI. Se introduce a los problemas que se presentan en el departamento con el manejo de la información allí generada y se analiza las variantes posibles para la solución de dichos problemas. A nivel internacional y nacional se buscan y comparan alternativas.

### 1.2 Objeto de estudio

#### 1.2.1 Descripción General

A lo largo de la historia de la humanidad la salud ha sido tema de interés principal en las investigaciones de la ciencia para dar respuesta a las miles de incógnitas que el cuerpo humano guarda. Mucho ha sido el desarrollo alcanzado por la ciencia en la rama de la medicina y a medida que crece la cultura médica más interés manifiestan las personas por mantener una salud impecable. Es cuando las personas se preocupan por su aspecto personal que la salud bucal toma gran importancia, la estomatología es la rama de la medicina que se dedica al estudio y atención de la salud bucal. Las consultas o clínicas estomatológicas prestan los servicios para que las personas se atiendan cuando presenten problemas o incluso para informarse de cómo prevenir cualquier tipo de enfermedad bucal.

En Cuba estos tratamientos al igual que toda la atención médica en general se da de forma gratuita, y desde edades tempranas la salud bucal es objeto de atención por parte de las autoridades sanitarias del país, nuestra universidad cuenta con una clínica que presta servicios estomatológicos a toda su comunidad, tanto estudiantes como profesores, personal no docente e incluso los constructores que se encuentran en las obras constructivas de la universidad pueden atenderse en ella. Todos son pacientes que al llegar a la clínica primero pasarían por admisión y archivo donde pueden sacar un turno para atenderse con un estomatólogo, luego pasarían a la consulta, espacio donde se recibe toda la atención

necesaria para dar solución al problema que condujo la visita a la clínica. Incluso en la consulta se recibe educación para prevenir enfermedades que afectan la salud bucal. En el trabajo diario de los estomatólogos en la clínica de la UCI se genera gran cantidad de información, la cual es gestionada de forma manual y escrita. Todo esto es archivado para su uso posterior en caso que se necesite.

### **1.2.2 Descripción actual del dominio del problema**

En el Departamento de Estomatología de la UCI se encuentran archivadas todas las historias clínicas de cada paciente, el paciente aquejado de cualquier malestar acude a la clínica para ser atendido, al arribar primeramente son atendidos en la recepción donde se le asigna al paciente un turno médico, aquí se recogerán los datos más importantes del paciente a utilizar en todo el proceso, tales como: nombre, apellidos, facultad, numero de solapín, grupo, si tiene o no historia clínica, etc.

Una vez concluido este proceso los médicos son notificados con el listado de pacientes que serán atendidos en la consulta. Si el paciente no contiene Historia Clínica se solicitan los datos necesarios para crear una nueva. Luego el médico según los síntomas de paciente procede a realizar un examen bucal en todas sus regiones bucales para detectar cualquier tipo de enfermedad incluyendo además el completamiento del dentigrama para determinar los cariacos obturados perdidos (COP), luego informa un diagnostico definitivo y realiza el tratamiento a seguir.

Si el paciente contiene historia clínica el médico la localiza, luego verifica si tiene alguna cita asignada y cuáles fueron las observaciones de exámenes anteriores para darle continuidad a su tratamiento o de lo contrario procede de la misma manera que al terminar de crear una nueva historia clínica, por ultimo actualiza los datos en el historial médico del paciente y lo archiva.

Si el paciente necesitara de continuidad del tratamiento médico ya sea a través de medicamentos o desde la hospitalización del mismo o la remisión a algún especialista, los estomatólogos a través de recetas médicas o formularios de remisión le dan al paciente lo que necesita para la continuidad de su tratamiento.

En la Universidad de las Ciencias Informáticas las consultas estomatológicas se realizan semanalmente de lunes a sábado y están organizadas de tal manera que cada doctor tiene asignada varias facultades a atender excepto los días que están de urgencia. Solamente pueden ser atendidos por doctor 10 pacientes por la mañana y 10 pacientes por la tarde.

Conjuntamente con la atención prestada al paciente se lleva a cabo un proceso de educación donde se informa al paciente acerca de cómo llevar una salud bucal excelente, se le brindan materiales y documentos relacionados con la educación y prevención de enfermedades bucales, todo esto se realiza de forma personal.

### **1.2.3 Situación problemática**

Existen aspectos deficientes que han generado la situación problemática que se pretende resolver con este trabajo, en estos momentos el proceso de gestión de la información de un paciente en una consulta estomatológica en la UCI se realiza de forma manual, diariamente se originan un número elevado de datos que se almacenan en papel provocando un manejo poco eficiente de tan alto volumen de información.

Estos datos son guardados por meses sin tener ningún mecanismo que ayude a su procesamiento; esto hace necesario que todo el volumen de información a papel tenga que almacenarse y conservarse durante tiempo indefinido para evitar pérdidas irreparables. Esto trae como consecuencia que las búsquedas sean cada vez más difíciles de realizar a la vez que son más frecuentes en el trabajo.

Así también el control estadístico de información relacionada con la clínica, como pueden ser los partes diarios de cada estomatólogo o el gasto material general de las consultas o particular de cada estomatólogo; se lleva de forma manual. Este proceso suele ser muy engorroso y siempre está a expensas de errores humanos que traen como consecuencia que se obtengan resultados no esperados.

En general no se cuenta en la UCI con una aplicación que dé respuesta a las necesidades de gestión de información requeridas en el servicio de odontología por lo que se hace necesario la existencia de un sistema automatizado que contribuya a mejorar la calidad de los servicios.

### **1.2.4 Análisis de otras soluciones existentes**

Desde que en 1959 se hicieron los primeros intentos de introducción de técnicas de informática en la medicina, hasta la fecha, ha quedado bien claro el papel importante que desempeñan estas técnicas en los sistemas sanitarios modernos. Dentro de la informática, esta nueva ciencia y tecnología de la información, la informática médica, ha adquirido rango de verdadera especialidad.

A nivel mundial cada vez se desarrolla un gran número de aplicaciones para la informatización de los servicios médicos con similitudes y diferencias desde el punto de vista implementación y servicios brindados.

Numerosos son los intentos y más que intentos (pues se cuentan algunos logros) en la realización de Sistemas para la Gestión Odontológica a nivel Internacional y en la Industria Nacional, pues se han obtenido resultados en estos ámbitos aunque con algunas atenuantes en ambos.

Para los desarrollados a nivel internacional se tiene que, aunque sus capacidades de gestión en ocasiones sobrepasan las expectativas, tienden a ser por estas mismas causas en exceso caros y con un costo añadido de soporte que no es posible asumir, también poseen el inconveniente del gran esfuerzo asociado a su implantación para su adaptación a las necesidades de nuestro Sistema de Salud, añadido a esto encontramos las restricciones comerciales impuestas por los EEUU hacia nuestro país, las cuales incluyen la compra venta de Sistemas Informáticos. Los de elaboración nacional, que por demás no son demasiados, están atados en su elaboración a IDEs propietarios, SGBDs propietarios también, así como provistos de una pobre documentación que los hace sensibles a cambios futuros, estas atenuantes por solo citar algunas.

Existen varias aplicaciones automatizadas vinculadas al campo de acción, cada uno de ellos trata el problema del manejo de grandes volúmenes de información, pero que aun necesitan de una serie de servicios para poder brindar en algunos centros altamente informatizados como lo es el caso de la UCI.

Dentro de estas aplicaciones encontramos:

- 1. INFORDENT** Paso a ser una empresa de prestación de servicios informáticos al ser absorbida por otras empresas mayores (CA- T- IMATGE, S. L.).
- 2. @ODONTOCLINIC** Consiste en una sencilla aplicación de escritorio donde podrá conservar y administrar toda la información que genere su actividad profesional. Orientada totalmente a Windows en la mayoría de sus versiones de sistemas operativos.
- 3. ODONTOSOFT** De acuerdo a las necesidades de algunos odontólogos con un consultorio de manejo sencillo, y que no requieren de un software demasiado completo para su administración. Este software tiene su principal atenuante en el precio que se eleva a medida que crece el número de ordenadores online, además de que también es una aplicación de escritorio y orientada a plataforma propietarias, con todo lo que significa esto en materia de precios.

**4. ODONTO 4.0** Desarrollado en Cuba, sus funciones principales son el control de historias clínicas odontológicas, planificación de citas, dentigrama, facturación de consultas y más.

### **1.3 Conclusiones**

En este capítulo profundizamos en el conocimiento de cómo surgen los servicios de estomatología, como funciona de manera general una consulta estomatológica y específicamente el Departamento de Estomatología de la UCI. Se abordaron además conceptos básicos asociados al dominio del problema y un análisis de las soluciones existentes en la actualidad y si es posible su aplicación para dar solución la situación problemática planteada a partir del análisis de diferentes factores concurrentes en los procesos que se llevan a cabo en la clínica estomatológica de nuestra universidad.

## CAPITULO 2

# Tendencias y Tecnologías Actuales

### 2.1 Introducción

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

### 2.2 Las aplicaciones Web

Una aplicación web es aquella que los usuarios usan accediendo a un servidor web a través de Internet o de una intranet. Las aplicaciones web son populares debido a la practicidad del navegador web como cliente ligero. La habilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

En los primeros tiempos de la computación cliente/servidor, cada aplicación tenía su propio programa cliente y su interfaz de usuario, estos tenían que ser instalados separadamente en cada estación de trabajo de los usuarios. Una mejora al servidor, como parte de la aplicación, requería típicamente una mejora de los clientes instalados en cada una de las estaciones de trabajo, añadiendo un costo de soporte técnico y disminuyendo la eficiencia del personal.

En contraste, las aplicaciones web generan dinámicamente una serie de páginas en un formato estándar, soportado por navegadores web comunes como HTML o XHTML. Se utilizan lenguajes interpretados del lado del cliente, tales como Java Script, para añadir elementos dinámicos a la interfaz de usuario. Generalmente cada página web individual es enviada al cliente como un documento estático, pero la secuencia de páginas provee de una experiencia interactiva.

Las aplicaciones Web son una especialización y concreción de las aplicaciones cliente/servidor, o sea, su arquitectura general es la de un sistema cliente/servidor, donde tanto el cliente (el navegador) como el servidor (el servidor Web), y el protocolo mediante el que se comunican (el HTTP: HyperText Transfer Protocol) son estándar, y no han de ser creados por el desarrollador. La parte del cliente de las aplicaciones Web está formada por el código HTML (HyperText Markup Language) que forma la página Web, con opción a código ejecutable mediante los lenguajes script de los navegadores (JavaScript,

VBScript, PerlScript) o mediante pequeños programas (applets) en Java. La parte del servidor está formada por un programa o script que es ejecutado por el servidor Web, y cuya salida se envía al navegador del cliente. (1)

### Ventajas de las Aplicaciones Web: (2)

- **Multiplataforma:** Con un solo programa, un único ejecutable, nuestras aplicaciones pueden ser utilizada a través de múltiples plataformas, tanto de hardware como de software.
- **Actualización instantánea:** Debido que todos los usuarios de la aplicación hacen uso de un sólo programa que radica en el servidor, los usuarios siempre utilizarán la versión más actualizada del sistema.
- **Suave curva de aprendizaje:** Los usuarios, como utilizan la aplicación a través de un navegador, hacen uso del sistema tal como si estuvieran navegando por Internet, por lo cual su acceso es más intuitivo.
- **Fácil de integrar con otros sistemas:** Debido a que se basa en protocolos estándares, la información manejada por el sistema puede ser accedida con mayor facilidad por otros sistemas.
- **Acceso móvil:** El usuario puede acceder a la aplicación con la única restricción de que cuente con un acceso a la red privada de la organización o a Internet, dependiendo de las políticas de dicha organización; puede hacerlo desde una computadora de escritorio, una laptop o desde una agenda electrónica; desde su oficina, hogar u otra parte del mundo.

El desarrollo de aplicaciones Web está siendo utilizado en muchas organizaciones, ésta situación va ir creciendo indefinidamente. Es por ello que día a día se requieran más programadores capacitados para desarrollos basados en el Word Wide Web (WWW).

### Existen además algunas desventajas, estas son:

- **Acceso limitado,** la necesidad de conexión permanente y rápida a Internet hacen que el acceso a estas aplicaciones no esté al alcance de todos.
- **La interactividad no se produce en tiempo real,** en las aplicaciones web cada acción del usuario conlleva un tiempo de espera algunas veces excesivo hasta que se obtiene la reacción del sistema.



- Elementos de interacción muy limitados. En comparación con el software de escritorio, las posibilidades de interacción con el usuario que ofrecen las aplicaciones web (mediante formularios principalmente) son muy escasas.
- Diferencias de presentación entre plataformas y navegadores. La falta de estándares ampliamente soportados dificulta el desarrollo de las aplicaciones.

Aunque muchas variaciones son posibles, una aplicación web está comúnmente estructurada como una aplicación de tres-capas. En su forma más común, el navegador web es la primera capa, un motor usando alguna tecnología web dinámica (ejemplo: CGI, PHP, Java Servlets o ASP) es la capa de en medio, y una base de datos como última capa. El navegador web manda peticiones a la capa media, que la entrega valiéndose de consultas y actualizaciones a la base de datos generando una interfaz de usuario.

### 2.3 Modelo Cliente Servidor

La arquitectura cliente-servidor llamado modelo cliente-servidor o servidor-cliente es una forma de dividir y especializar programas y equipos de cómputo a fin de que la tarea que cada uno de ellos realizada se efectúe con la mayor eficiencia, y permita simplificarlas. (3)

Se dice que 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. (4)

Características de la arquitectura Cliente/Servidor: (5)

- Orientado a servicios. El servidor los ofrece y el cliente los consume.
- Compartición de recursos. Servicios ofrecidos a muchos clientes. Un servidor puede atender muchos clientes que solicitan esos servicios.
- Transparencia de ubicación. El servidor es un proceso que puede residir en el mismo ordenador que el cliente o en uno distinto a lo largo de una red. Un programa puede ser un servidor en un momento y convertirse en un cliente posteriormente.

- Mezcla e igualdad. Esta es una de las más importantes ventajas de este paradigma. Una aplicación cliente/servidor, idealmente es independiente del hardware y de sistemas operativos; mezclando e igualando estas plataformas.
- Interacción a través de mensajes, para envío y respuesta de servicios.
- Servicios encapsulados, exponiendo los servicios a través de interfaces, lo que facilita la sustitución de servidores sin afectar los clientes; permitiendo a la vez una fácil escalabilidad.

### Ventajas de la arquitectura cliente/servidor: (2)

- El servidor no necesita potencia de procesamiento, parte del proceso se reparte con los clientes.
- Se reduce el tráfico de red considerablemente. Idealmente, el cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre.

Las arquitecturas de dos capas contienen tres componentes distribuidos en dos capas: cliente (solicitante de servicios) y servidor (proveedor de servicios).

### Los tres componentes son: (2)

1. Interfaz de usuario al sistema. Tales como una sesión, entradas de texto, desplegado de menús, etc.
2. Administración de procesamiento. Tales como la ejecución de procesos, el monitoreo de los mismos y servicios de procesamiento de recursos.
3. Administración de bases de datos. Tales como los servicios de acceso a datos y archivos.

La arquitectura de software de tres capas emergió en la década de los noventa para solventar las limitaciones de la arquitectura de dos capas. La tercera capa (capa de servicios) se localiza entre la interfaz de usuarios (cliente) y el administrador de datos (servidor). Esta capa intermedia provee de servicios para la administración de procesos (tal como desarrollo, monitoreo y alimentación de procesos) que son compartidos por múltiples aplicaciones.

El servidor de la capa intermedia (también conocido como servidor de aplicaciones) centraliza la lógica de las aplicaciones, haciendo que la administración de cambios sea más sencilla. En arquitecturas más simples, cualquier cambio en la lógica, implica reescribir todas las aplicaciones que dependan de ésta. (6)

La calidad tan especial de esta arquitectura consiste en aislar la lógica de la aplicación y en convertirla en una capa intermedia bien definida y lógica del software. (7) Esta separación entre la lógica de aplicación de la interfaz de usuario añade una enorme flexibilidad al diseño de la aplicación. Pueden construirse y desplegarse múltiples interfaces de usuario sin cambiar en absoluto la lógica de aplicación siempre que esté presente una interfaz claramente definida a la capa de presentación.

## **2.4 Lenguajes de Programación Web**

La diferencia fundamental de Internet a otros medios de comunicación es la interacción y personalización de la información con el usuario. Esto se logra por medio de alguno de los diferentes lenguajes de programación para Web que existen hoy en día. Dichos lenguajes se clasifican en dos partes fundamentales: los lenguajes del lado del Servidor y los lenguajes del lado del Cliente.

Entre los lenguajes del lado del servidor podemos encontrar entre los más sobresalientes por el auge que estos han tenido, algunos como PERL, ASP, PHP, Java, etc. Estos se caracterizan por desarrollar la lógica de negocio dentro del Servidor, además de ser los encargados del acceso a Bases de Datos, tratamiento de la Información etc. Del lado del cliente se encuentra principalmente el Java Script, encargado de aportar dinamismo a la aplicación en los navegadores. Seguidamente analizamos el lenguaje utilizado para el desarrollo de este trabajo.

### **2.4.1 PHP**

El PHP, es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor. El PHP originalmente diseñado en Perl, seguidos por la escritura de un grupo de CGI binarios escritos en el lenguaje C por el programador Danés-Canadiense Rasmus Lerdorf en el año 1995 para mostrar su currículum vitae y guardar ciertos datos, como la cantidad de tráfico que su página web recibía. En los siguientes tres años, se fue convirtiendo en lo que se conoce como PHP/FI 2.0.

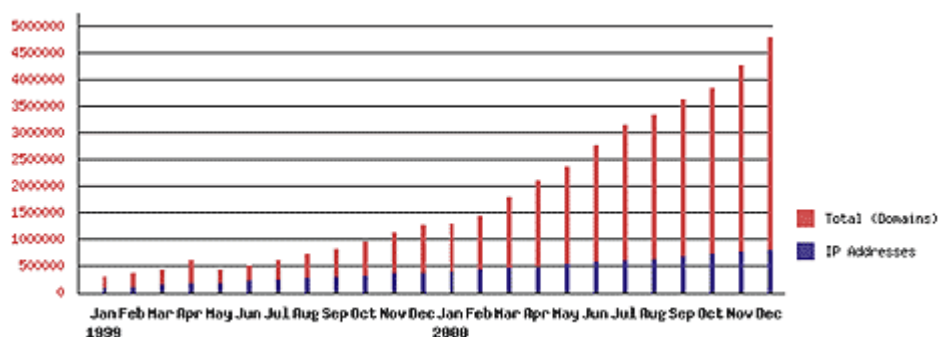
Esta forma de programar llegó a muchos usuarios, pero el lenguaje no tomó el peso actual hasta que dos programadores israelíes de Technion, Zeev Suraski y Andi Gutmans reescribieron el analizador gramatical en el año 1997, y crearon la base del PHP 3, cambiando el nombre del lenguaje a la forma actual. Una de las mejores características de PHP 3.0 era su gran extensibilidad. Además de proveer a los usuarios finales de una sólida infraestructura para muchísimas bases de datos, protocolos y APIs, las

características de extensibilidad de PHP 3.0 atrajeron a docenas de desarrolladores a unirse y enviar nuevos módulos de extensión. Sin duda, ésta fue la clave del enorme éxito de PHP 3.0. (8)

En el invierno de 1998, poco después del lanzamiento oficial de PHP 3.0, Andi Gutmans y Zeev Suraski comenzaron a trabajar en la reescritura del núcleo de PHP. Los objetivos de diseño fueron mejorar la ejecución de aplicaciones complejas, y mejorar la modularidad del código base de PHP. Estas aplicaciones se hicieron posibles por las nuevas características de PHP 3.0 y el apoyo de una gran variedad de bases de datos y APIs de terceros, pero PHP 3.0 no fue diseñado para el mantenimiento tan complejo de aplicaciones eficientemente. (8)

El nuevo motor, apodado 'Motor Zend' (comprimido de sus apellidos, Zeev y Andi), alcanzó estos objetivos de diseño satisfactoriamente, y se introdujo por primera vez a mediados de 1999. PHP 4.0, basado en este motor, y acoplado con un gran rango de nuevas características adicionales, fue oficialmente liberado en Mayo de 2000, casi dos años después que su predecesor, PHP 3.0. Además de la mejora de ejecución de esta versión, PHP 4.0 incluía otras características clave como el soporte para la mayoría de los servidores Web, sesiones HTTP, buffers de salida, formas más seguras de controlar las entradas de usuario y muchas nuevas construcciones de lenguaje. (8)

Hoy, se estima que PHP es usado por cientos de miles de programadores y muchos millones de sitios informan que lo tienen instalado, sumando más del 20% de los dominios en Internet.



Gráfica del número de dominios y direcciones IP que utilizan PHP.

En el último año, el número de servidores que utilizan PHP se ha disparado, logrando situarse cerca de los 5 millones de sitios y 800.000 direcciones IP, lo que le ha convertido a PHP en una tecnología popular. Esto es debido, entre otras razones, a que PHP es el complemento ideal para que el tándem Linux-Apache sea compatible con la programación del lado del servidor de sitios web. Gracias a la aceptación

que ha logrado, y los grandes esfuerzos realizados por una creciente comunidad de colaboradores para implementarlo de la manera más óptima, podemos asegurar que el lenguaje se convertirá en un estándar que compartirá los éxitos augurados al conjunto de sistemas desarrollados en código abierto. (9)

PHP es un lenguaje de programación usado generalmente para la creación de contenido para sitios web, se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios.

Una de sus características más potentes es su soporte para gran cantidad de bases de datos. Entre las que se pueden mencionar InterBase, mSQL, MySQL, Oracle, Informix, PostgreSQL, entre otras. PHP también ofrece la integración con varias bibliotecas externas, que dan al desarrollador la posibilidad de realizar cualquier tarea, desde generar documentos en pdf (Portable Document Format) hasta analizar código XML (eXtensible Markup Language) y últimamente también para la creación de otro tipo de programas incluyendo aplicaciones con interfaz gráfica usando la librería GTK+. (10)

Ofrece una solución simple y universal para las paginaciones dinámicas del Web de fácil programación. Su diseño elegante lo hace perceptiblemente más fácil de mantener y ponerse al día que el código comparables en otros lenguajes. Es software libre, lo que implica menos costes y servidores más baratos que otras alternativas. Es muy rápido y su integración con la base de datos MySQL y el servidor Apache, le permite constituirse como una de las alternativas más atractivas del mercado. (10)

Es multiplataforma, funciona tanto para Unix (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. Su sintaxis está inspirada en C, ligeramente modificada para adaptarlo al entorno en el que trabaja, de modo que si se está familiarizado con esta sintaxis, le resultará muy fácil aprender PHP. Su librería estándar es realmente amplia, lo que permite reducir los llamados "costes ocultos", uno de los principales defectos de ASP (Active Server Pages). (10)

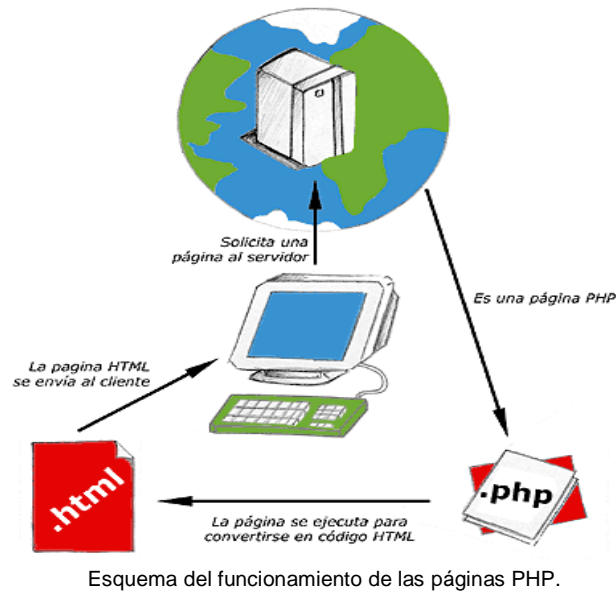
PHP tiene una de las comunidades más grandes en Internet, con lo que no es complicado encontrar ayuda, documentación, artículos, noticias, y más recursos. Ofrece una solución simple y universal para las paginaciones dinámicas del Web de fácil programación. Su diseño elegante lo hace perceptiblemente más fácil de mantener y ponerse al día, a diferencia con el código de otros lenguajes.

Debido a su amplia distribución PHP está perfectamente soportado por una gran comunidad de desarrolladores. Como producto de código abierto, PHP goza de la ayuda de un gran grupo de programadores, permitiendo que los fallos de funcionamiento se encuentren y se reparan rápidamente. El código se pone al día continuamente con mejoras y extensiones de lenguaje para ampliar las capacidades de PHP. (10)

Su interpretación y ejecución se da en el servidor, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para generar archivos PDF, FLASH, así como imágenes en diferentes formatos, entre otras cosas.

El funcionamiento del PHP se puede describir a través de los pasos siguientes:

- Escribir en las páginas HTML pero con el código PHP dentro.
- Guardar la página en el servidor Web.
- Un navegador solicita una página al servidor.
- El servidor interpreta el código PHP.
- El servidor envía el resultado del conjunto de código HTML y el resultado del código PHP que también es HTML.



En ningún caso se envía código PHP al navegador, por lo que todas las operaciones realizadas son transparentes al usuario, el código PHP es ejecutado en el servidor y el resultado es enviado al navegador. El resultado es normalmente una página HTML. Por lo que al usuario le parecerá que está visitando una página HTML que cualquier navegador puede interpretar. Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que el navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

(6)

Después de seis años, y después que la comunidad ha revisado el paquete de legados que ha dejado el PHP, se han realizado cambios estructurales en el lenguaje para ofrecer innovación en el nuevo PHP 5 y solucionar muchos de los problemas encontrados en PHP 4.

Afortunadamente, lo nuevo de PHP 5 mejora muchas áreas en el lenguaje y su ejecución, como por ejemplo:

- Programación orientada a objetos (OOP).
- MySQL.
- XML.

- Integración nativa con el Zend Engine.

Los diseñadores de PHP5 han realizado un cambio radical en el tratamiento de las variables objeto: en PHP5 todas las variables que nombran objetos son en realidad referencias. No hay que usar el operador '&' ni en las asignaciones, ni en el paso de parámetros que son objetos, ahorrándose con ello gran cantidad de potenciales errores. La principal novedad en las clases de PHP5 es la inclusión de modificadores de control de acceso para implementar la encapsulación piedra angular en la programación orientada a objetos de la que adolecía PHP4. (6)

Otros lenguajes como Perl (Practical Extraction and Report Language), ASP (Active Server Pages) y JSP (Java Server Pages) tienen características similares al PHP aunque poseen rasgos que los marcan y por ello los distingue.

Entre ellos podemos encontrar: (6)

- Características multiplataforma: Menos el ASP, que es solamente soportado por la plataforma Windows, los demás lenguajes están soportados en múltiples plataformas.
- Velocidad de ejecución: la velocidad es mayor en PHP, seguidos por PERL y JSP.
- Disponibilidad de recursos: actualmente los más utilizados en la Internet son el PHP y el JSP, siendo más utilizado en la publicación de artículos y códigos de ejemplos. PHP tiene una de las comunidades más grandes en Internet, al igual que la de Java.
- Familiaridad con el lenguaje: En la universidad los lenguajes más utilizados por los programadores es el ASP y el PHP.

De acuerdo a estas comparaciones, el PHP resulta mucho más favorecido, por tanto pensamos que es el adecuado para implementar la propuesta de sistema de este trabajo, particularmente PHP5.

## 2.5 Servidores Web

Un servidor web es un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos. (11)



Un servidor web se encarga de mantenerse a la espera de peticiones HTTP llevada a cabo por un cliente HTTP que solemos conocer como navegador. El navegador realiza una petición al servidor y éste le responde con el contenido que el cliente solicita. A modo de ejemplo, al teclear <http://tesis.uci.cu/> en nuestro navegador, éste realiza una petición HTTP al servidor de dicha dirección. El servidor responde al cliente enviando el código HTML de la página; el cliente, una vez recibido el código, lo interpreta y lo muestra en pantalla. Como vemos con este ejemplo, el cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página; el servidor tan sólo se limita a transferir el código de la página sin llevar a cabo ninguna interpretación de la misma. (11)

Algunos servidores web importantes son:

- Apache.
- Internet Information Server.

### 2.5.1 Apache

El servidor **Apache** es un software que está estructurado en módulos, es decir, está dividido en muchas porciones de código que hacen referencia a diferentes aspectos o funcionalidades del servidor web. Esta modularidad es intencionada ya que la configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo.

Hoy en día Apache es el servidor web más utilizado del mundo, encontrándose muy por encima de sus competidores, tanto gratuitos como comerciales. Es un software de código abierto que funciona sobre cualquier plataforma.

Tiene capacidad para servir páginas tanto de contenido estático, para lo que nos serviría sencillamente un viejo ordenador 486, como de contenido dinámico a través de otras herramientas soportadas que facilitan la actualización de los contenidos mediante bases de datos, ficheros u otras fuentes de información, es muy potente y altamente configurable.

El servidor web Apache es uno de los mayores triunfos del software libre. En Diciembre de 1997 tenía una cuota de mercado cercana al 45% y en la actualidad (Julio 2000) ya está por encima del 60%, según los estudios de Netcraft que ya se han establecido como la referencia dentro del mercado de servidores web.

Esta es la primera cifra que hace que cualquier responsable de la estrategia internet de una empresa tenga que tomar a Apache como el servidor de referencia. (12)

Apache era inicialmente unos parches al servidor que de WWW de NCSA conocido como httpd (principios de 1995). Al igual que GNU/Linux, fue un proyecto que atrajo a mucha gente por el gran interés de su objetivo: lograr el servidor web más rápido, más eficiente y con mayor funcionalidad desde el enfoque del software libre. Y ha sido un objetivo que como veremos, se ha logrado. Sólo hacían sobras en Apache ciertos aspectos de rendimiento, fundamentalmente por no utilizar hebras de ejecución. (12)

Con un enorme equipo de voluntarios a lo largo y ancho de toda la red, se han logrado estos objetivos, logrando batir a compañías comerciales de la talla de Microsoft y Netscape. Y no sólo se ha logrado batir a los servidores web de grandes compañías: se ha logrado atraer al todopoderoso IBM que está apoyando Apache tanto a nivel de soporte como de desarrollo. La incorporación de empresas al desarrollo de proyectos de software abierto es una característica que ha brotado en el año 2000. Otros ejemplos son Corel, Dell etc. (12)

Apache es un servidor web flexible, rápido y eficiente, continuamente actualizado y adaptado a los nuevos protocolos (HTTP). Entre sus características se destacan: (12)

- Multiplataforma
- Es un servidor de web conforme al protocolo HTTP/1.1
- Modular: Puede ser adaptado a diferentes entornos y necesidades, con los diferentes módulos de apoyo que proporciona, y con la API de programación de módulos, para el desarrollo de módulos específicos.
- Basado en hebras en la versión 2.0
- Incentiva la realimentación de los usuarios, obteniendo nuevas ideas, informes de fallos y parches para la solución de los mismos.
- Se desarrolla de forma abierta
- Extensible: gracias a ser modular se han desarrollado diversas extensiones entre las que destaca PHP, un lenguaje de programación del lado del servidor.

El servidor Apache es un software que está estructurado en módulos. La configuración de cada módulo se hace mediante la configuración de las directivas que están contenidas dentro del módulo. Los módulos del Apache se pueden clasificar en tres categorías: (13)

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

Las funcionalidades más elementales se encuentran en el módulo base, siendo necesario un módulo multiproceso para manejar las peticiones. Se han diseñado varios módulos multiprocesos para cada uno de los sistemas operativos sobre los que se ejecuta el Apache, optimizando el rendimiento y rapidez del código. El resto de funcionalidades del servidor se consigue por medio de módulos adicionales que se pueden cargar. Para añadir un conjunto de utilidades al servidor, simplemente hay que añadirle un módulo, de forma que no es necesario volver a instalar el software. (13)

## 2.6 Sistema de Gestión de Bases de Datos (SGBD)

Un Sistema Gestor de base de datos (SGBD) es un conjunto de programas que permiten crear y mantener una Base de datos, asegurando su integridad, confidencialidad y seguridad.

Para plasmar los tres niveles en el enfoque o modelo de datos seleccionado, es necesaria una aplicación que actúe de interfaz entre el usuario, los modelos y el sistema físico. Esta es la función que desempeñan los SGBD, ya reseñados, y que pueden definirse como un paquete generalizado de software, que se ejecuta en un sistema computacional anfitrión, centralizando los accesos a los datos y actuando de interfaz entre los datos físicos y el usuario. Las principales funciones que debe cumplir un SGBD se relacionan con la creación y mantenimiento de la base de datos, el control de accesos, la manipulación de datos de acuerdo con las necesidades del usuario, el cumplimiento de las normas de tratamiento de datos, evitar redundancias e inconsistencias y mantener la integridad. (14)

Algunas de las características deseables en un Sistema Gestor de base de datos SGBD son: (15)

- **Control de la redundancia:** La redundancia de datos tiene varios efectos negativos (duplicar el trabajo al actualizar, desperdicia espacio en disco, puede provocar inconsistencia de datos) aunque a veces es deseable por cuestiones de rendimiento.
- **Restricción de los accesos no autorizados:** cada usuario ha de tener unos permisos de acceso y autorización.
- **Cumplimiento de las restricciones de integridad:** el SGBD ha de ofrecer recursos para definir y garantizar el cumplimiento de las restricciones de integridad.

Ejemplo de los SGBD utilizados en el mundo son Oracle, MySQL, SQL Server de Microsoft, PostgreSQL entre otros. Estos presentan un enfoque relacional con un buen basamento matemático centrado en el Álgebra Relacional. Todos los sistemas mencionados anteriormente facilitan el trabajo con la base de datos y tienen características que los diferencian, por ejemplo: (15)

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

Como SGBD se seleccionó el MySQL por las ventajas que ofrece y por requerimientos del cliente.

### 2.6.1 MySQL

MySQL surgió como un intento de conectar el gestor mSQL a las tablas propias de MySQL AB, usando sus propias rutinas a bajo nivel. Tras unas primeras pruebas, vieron que mSQL no era lo bastante flexible para lo que necesitaban, por lo que tuvieron que desarrollar nuevas funciones. Esto resultó en una interfaz SQL a su base de datos, con una interfaz totalmente compatible a mSQL. Fue escrito en C y C++ y se destaca por su gran adaptación a diferentes entornos de desarrollo, permitiendo su interacción con los lenguajes de programación más utilizados como PHP, Perl y Java y su integración en distintos sistemas operativos. (17)

Es el gestor de bases de datos de código fuente abierto más usado del mundo. Su ingeniosa arquitectura lo hace extremadamente rápido y fácil de personalizar. La extensiva reutilización del código dentro del software y una aproximación mínima para producir características funcionalmente ricas, ha dado lugar a un sistema de administración de la base de datos incomparable en velocidad, compactación, estabilidad y facilidad de despliegue. La exclusiva separación del core server del manejador de tablas, permite funcionar a MySQL bajo control estricto de transacciones o con acceso a disco no transaccional ultrarrápido. (16)

Es un sistema de administración de bases de datos. Una base de datos es una colección estructurada de datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto monto de información en una red corporativa. Para agregar, acceder y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server. Dado que los computadores son muy buenos manejando grandes cantidades de información, los administradores de bases de datos juegan un papel central en computación, como aplicaciones independientes o como parte de otras aplicaciones. (16)

Es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido. (16)

Es un software de código abierto esto quiere decir que es accesible para cualquiera, para usarlo o modificarlo. Podemos descargar MySQL desde Internet y usarlo sin pagar nada, de esta manera cualquiera puede inclinarse a estudiar el código fuente y cambiarlo para adecuarlo a sus necesidades. MySQL usa el GPL (GNU Licencia Publica General) para definir que podemos y no podemos hacer con el software en diferentes situaciones. Entre otras cuestiones esta licencia aclara que no cuesta dinero a menos que lo incluyamos en un software comercial y tenemos el código fuente. (16)

Es muy rápido, confiable, robusto y fácil de usar tanto para volúmenes de datos grandes como pequeños (siempre, claro está, comparada con las de su categoría, como veremos más adelante en este informe). Además tiene un conjunto muy práctico de características desarrolladas en cooperación muy cercana con los usuarios. Sin embargo bajo constante desarrollo, MySQL hoy en día ofrece un rico y muy útil conjunto de funciones. La conectividad, velocidad y seguridad hace de MySQL altamente conveniente para acceder a bases de datos en Internet. (16)

Las principales características de este gestor de bases de datos son las siguientes: (17)

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de APIs en gran cantidad de lenguajes (C, C++, Java, PHP, etc.).
- Gran portabilidad entre sistemas.
- Soporta hasta 32 índices por tabla.
- Gestión de usuarios y passwords, manteniendo un muy buen nivel de seguridad en los datos.
- El principal objetivo de MySQL es velocidad y robustez.
- Escrito en C y C++, testado con GCC 2.7.2.1. Usa GNU autoconf para portabilidad.
- Puede trabajar en distintas plataformas y S.O. distintos.
- Sistema de contraseñas y privilegios muy flexibles y seguros.
- Todas las palabras de paso viajan encriptadas en la red.
- Registros de longitud fija y variable.
- Todas las columnas pueden tener valores por defecto.
- Utilidad (Isamchk) para chequear, optimizar y reparar tablas.
- Todos los datos están grabados en formato ISO8859\_1.
- Los clientes usan TCP o UNIX Socket para conectarse al servidor.
- El servidor soporta mensajes de error en distintas lenguas.
- Todos los comandos tienen -help o -? Para las ayudas.
- Diversos tipos de columnas como enteros de 1, 2, 3, 4, y 8 bytes, coma flotante, doble precisión, carácter, fechas, enumerados, etc.
- ODBC para Windows 95 (con fuentes), se puede utilizar ACCESS para conectar con el servidor.

Este gestor de bases de datos es, probablemente, el gestor más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen

infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración.

## 2.7 Proceso de Desarrollo de Software

Para controlar y planificar la propuesta que presenta este trabajo, se decidió utilizar como metodología el Proceso Unificado de Modelado (RUP), por sus características y las facilidades que aporta a todo el proceso.

El Proceso Racional Unificado o RUP (Rational Unified Process), es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. (6)

Es un proceso de software genérico que puede ser utilizado para una gran cantidad de tipos de sistemas de software, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de competencia y diferentes tamaños de proyectos. Provee un enfoque disciplinado en la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su meta es asegurar la producción de software de muy alta calidad que satisfaga las necesidades de los usuarios finales, dentro de un calendario y presupuesto predecible. (6)

RUP es uno de los procesos más generales de los existentes actualmente, ya que en realidad está pensado para adaptarse a cualquier proyecto, y no tan solo de software.

Las características del proceso unificado de modelado son: (18)

- **Centrado en los Modelos:** Los diagramas son un vehículo de comunicación más expresivo que las descripciones en lenguaje natural. Se trata de minimizar el uso de descripciones y especificaciones textuales del sistema.
- **Guiado/Manejado por casos de uso:** La razón de ser de un sistema software es servir a usuarios ya sean humanos u otros sistemas; un caso de uso es una facilidad que el software debe proveer a sus usuarios. Los casos de uso reemplazan la antigua especificación funcional tradicional y constituyen la guía fundamental establecida para las actividades a realizar durante todo el proceso de desarrollo incluyendo el diseño, la implementación y las pruebas del sistema.

- **Centrado en arquitectura:** La arquitectura involucra los elementos más significativos del sistema y está influenciada entre otros por plataformas software, sistemas operativos, manejadores de bases de datos, protocolos, consideraciones de desarrollo como sistemas heredados y requerimientos no funcionales. Los casos de uso guían el desarrollo de la arquitectura y la arquitectura se realimenta en los casos de uso, los dos juntos permiten conceptualizar, gestionar y desarrollar adecuadamente el software.
- **Iterativo e Incremental:** Para hacer más manejable un proyecto se recomienda dividirlo en ciclos. Para cada ciclo se establecen fases de referencia, cada una de las cuales debe ser considerada como un mini proyecto cuyo núcleo fundamental está constituido por una o más iteraciones de las actividades principales básicas de cualquier proceso de desarrollo.
- **Desarrollo basado en componentes:** La creación de sistemas intensivos en software requiere dividir el sistema en componentes con interfaces bien definidas, que posteriormente serán ensamblados para generar el sistema. Esta característica en un proceso de desarrollo permite que el sistema se vaya creando a medida que se obtienen o que se desarrollan y maduran sus componentes.
- **Utilización de un único lenguaje de modelación:** UML es adoptado como único lenguaje de modelación para el desarrollo de todos los modelos.
- **Proceso Integrado:** Se establece una estructura que abarque los ciclos, fases, flujos de trabajo, mitigación de riesgos, control de calidad, gestión del proyecto y control de configuración; el proceso unificado establece una estructura que integra todas estas facetas. Además esta estructura cubre a los vendedores y desarrolladores de herramientas para soportar la automatización del proceso, soportar flujos individuales de trabajo, para construir los diferentes modelos e integrar el trabajo a través del ciclo de vida y a través de todos los modelos.

Un proyecto realizado siguiendo RUP se divide en cuatro fases: (19)

1. **inicio:** se hace un plan de fases, se identifican los principales casos de uso y se identifican los riesgos
2. **elaboración:** se hace un plan de proyecto, se completan los casos de uso y se eliminan los riesgos



3. **construcción:** se concentra en la elaboración de un producto totalmente operativo y eficiente y el manual de usuario
4. **transición:** se implementa el producto en el cliente y se entrena a los usuarios. Como consecuencia de esto suelen surgir nuevos requisitos a ser analizados.

En cada fase se ejecutarán una o varias iteraciones (de tamaño variable según el proyecto), y dentro de cada una de ellas seguirá un modelo de cascada para los flujos de trabajo que requieren las nuevas actividades anteriormente citadas. (19)

RUP define nueve actividades a realizar en cada fase del proyecto (19)

1. Modelado del negocio
2. Análisis de requisitos
3. Análisis y diseño
4. Implementación
5. Test
6. Distribución
7. Gestión de configuración y cambios
8. Gestión del proyecto
9. Gestión del entorno

Y el flujo de trabajo (workflow) entre ellas en base a los llamados diagramas de actividad. El proceso define una serie de roles que se distribuyen entre los miembros del proyecto y que definen las tareas de cada uno y el resultado (artefactos en la jerga de RUP) que se espera de ellos.

RUP se basa en casos de uso para describir lo que se espera del software y está muy orientado a la arquitectura del sistema, documentándose lo mejor posible, basándose en UML (Unified Modeling Language) como herramienta principal. (19)

RUP es un proceso muy general y muy grande, por lo que antes de usarlo habrá que adaptarlo a las características de la empresa. Está acompañado de una herramienta muy buena que soporta cada uno de los procesos que necesitamos: Rational Rose Enterprise Edition 2003. Además 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.

## 2.8 Lenguaje Unificado de Modelado (UML)

En todas las disciplinas de la Ingeniería se hace evidente la importancia de los modelos ya que describen el aspecto y la conducta de "algo". Ese "algo" puede existir, estar en un estado de desarrollo o estar, todavía, en un estado de planeación. Es en este momento cuando los diseñadores del modelo deben investigar los requerimientos del producto terminado y dichos requerimientos pueden incluir áreas tales como funcionalidad, performance y confiabilidad. Además, a menudo, el modelo es dividido en un número de vistas, cada una de las cuales describe un aspecto específico del producto o sistema en construcción. (20)

El modelado sirve no solamente para los grandes sistemas, aun en aplicaciones de pequeño tamaño se obtienen beneficios de modelado, sin embargo es un hecho que entre más grande y más complejo es el sistema, más importante es el papel de que juega el modelado por una simple razón: "El hombre hace modelos de sistemas complejos porque no puede entenderlos en su totalidad". (20)

UML es una técnica para la especificación sistemas en todas sus fases. Nació en 1994 cubriendo los aspectos principales de todos los métodos de diseño antecesores y, precisamente, los padres de UML son Grady Booch, autor del método Booch; James Rumbaugh, autor del método OMT e Ivar Jacobson, autor de los métodos OOSE y Objectory. La versión 1.0 de UML fue liberada en Enero de 1997 y ha sido utilizado con éxito en sistemas construidos para toda clase de industrias alrededor del mundo: hospitales, bancos, comunicaciones, aeronáutica, finanzas, etc. (20)

El Lenguaje de Modelado Unificado UML (Unified Modeling Language) es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad, es un lenguaje estándar para escribir planos de software. UML puede utilizarse para visualizar, especificar, construir y documentar los artefactos de un sistema que involucra gran cantidad de software. (21)

UML ofrece un estándar para describir un plano del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocios y funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos y componentes de software reutilizables.

UML se usa para definir un sistema de software; para detallar los artefactos en el sistema; para documentar y construir el lenguaje en el que está descrito el modelo. UML se puede usar en una gran variedad de formas para soportar una metodología de desarrollo de software (tal como el Proceso Unificado de Rational), pero no especifica en sí mismo qué metodología o proceso usar.

El UML permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma convencional y fácil de comprender para comunicarlas a otras personas que estén involucradas en el proceso de desarrollo de los sistemas, esto se lleva a cabo mediante un conjunto de símbolos y diagramas.

El UML está compuesto por diversos elementos gráficos que se combinan para conformar diagramas y proporciona un estándar que permite al analista de sistemas generar un anteproyecto de varias facetas que sean comprensibles para los clientes, desarrolladores y todos aquellos que estén involucrados en el proceso de desarrollo. Un modelo UML indica que es lo que supuestamente hará el sistema pero no como lo hará.

De forma general las principales características son: (20)

- Lenguaje unificado para la modelación de sistemas.
- Tecnología orientada a objetos.
- El cliente participa en todas las etapas del proyecto.
- Corrección de errores viables en todas las etapas.
- Aplicable para tratar asuntos de escala inherentes a sistemas complejos de misión crítica, tiempo real y cliente/servidor.

Los principales beneficios de UML son:

- Mejores tiempos totales de desarrollo (de 50 % o más).
- Modelar sistemas (y no sólo de software) utilizando conceptos orientados a objetos.

- Establecer conceptos y artefactos ejecutables.
- Encaminar el desarrollo del escalamiento en sistemas complejos de misión crítica.
- Crear un lenguaje de modelado utilizado tanto por humanos como por máquinas.
- Mejor soporte a la planeación y al control de proyectos.
- Alta reutilización y minimización de costos.

El Lenguaje Unificado de Modelado prescribe un conjunto de notaciones y diagramas estándar para modelar sistemas orientados a objetos, y describe la semántica esencial de lo que estos diagramas y símbolos significan. Mientras que ha habido muchas notaciones y métodos usados para el diseño orientado a objetos, ahora los modeladores sólo tienen que aprender una única notación. (22)

UML se puede usar para modelar distintos tipos de sistemas: sistemas de software, sistemas de hardware, y organizaciones del mundo real. UML ofrece nueve diagramas en los cuales modelar sistemas. (22)

- Diagramas de Casos de Uso para modelar los procesos 'business'.
- Diagramas de Secuencia para modelar el paso de mensajes entre objetos.
- Diagramas de Colaboración para modelar interacciones entre objetos.
- Diagramas de Estado para modelar el comportamiento de los objetos en el sistema.
- Diagramas de Actividad para modelar el comportamiento de los Casos de Uso, objetos u operaciones.
- Diagramas de Clases para modelar la estructura estática de las clases en el sistema.
- Diagramas de Objetos para modelar la estructura estática de los objetos en el sistema.
- Diagramas de Componentes para modelar componentes.
- Diagramas de Implementación para modelar la distribución del sistema.

El desarrollo de sistemas con UML siguiendo el proceso unificado incluye actividades específicas, cada una de ellas a su vez contienen otras subactividades las cuales sirven como una guía de cómo deben ser las actividades desarrolladas y secuenciadas con el fin de obtener sistemas exitosos; consecuentemente

el desarrollo de los sistemas puede variar de desarrollador en desarrollador, de proyecto en proyecto, de empresa en empresa adoptando siempre un Proceso de Desarrollo.

## **2.9 Conclusiones**

Para desarrollar el sistema se hace uso de la tecnología para la programación de páginas dinámicas el lenguaje PHP5 y con soporte de base de datos en MySQL. El proceso de desarrollo es RUP, el cual está basado en la orientación a objetos y la modelación visual usando UML, lo cual permite incorporar al proceso de desarrollo de software un mejor control de los requerimientos y cambios.

## CAPITULO 3

# Presentación de la Solución Propuesta

### 3.1 Introducción

En el presente capítulo se describe el negocio, aclarando sus principales procesos y datos de información manipulable resultantes, para a partir de ese análisis proponer una solución presentando un sistema que cumpla con los requisitos funcionales que respondan a las necesidades de informatización que se planteen en el modelado del negocio.

### 3.2 Modelo de Negocio

Cuando se comienza a desarrollar un sistema para una organización, empresa e institución es necesario comprender todos los procesos que en estas tienen lugar. Cada uno de estos procesos se caracteriza por una colección de datos que son producidos y manipulados mediante un conjunto de tareas, en las que ciertos agentes (trabajadores o departamentos) participan de acuerdo a un flujo de trabajo determinado. Además, estos procesos se hallan sujetos a un conjunto de reglas de negocio, que determinan la estructura de la información y las políticas de la organización. Por tanto, la finalidad del modelado del negocio es describir cada proceso del negocio, especificando sus datos, actividades, roles y reglas de negocio. (23)

#### 3.2.1 Procesos de Negocio

El primer paso del modelado del negocio consiste en capturar los procesos de negocio de la organización bajo estudio. La definición del conjunto de procesos del negocio es una tarea crucial, ya que define los límites del proceso de modelado posterior. (23)

A continuación se muestra una descripción detallada de los procesos del negocio propuesto:

1. **Asignar Turnos Médicos:** el turno lo asigna la recepcionista y se da diariamente a los pacientes por orden de llegada.
2. **Atender Pacientes:** una vez que el paciente llega a atenderse con el estomatólogo si tiene historia clínica se busca y esta se actualiza a medida que se atiende al paciente, sino se crea una historia clínica nueva con todos los datos necesarios, los cuales se van llenando en la misma medida en

que se atiende al paciente. Como conclusión de este proceso se elabora un informe de gasto material y parte diario de cada estomatólogo.

3. **Indicar Medicamentos:** en caso de que el paciente necesite medicamentos el estomatólogo lo envía a farmacia para que obtenga sus medicamentos a través de una receta médica.
4. **Remitir a un Especialista:** el estomatólogo al detectar que el paciente tiene una enfermedad que debe ser tratada con un especialista lo remite a esa especialidad a través de una orden de remisión a especialista.
5. **Remitir a Ingreso:** el estomatólogo reconoce que el paciente necesita ser ingresado y lo remite a la sala de ingreso a través de una orden de ingreso.

### 3.2.2 Reglas del Negocio

- Todas las personas que lleguen como paciente al departamento de estomatología tienen que obligatoriamente reservar un turno médico para poder ser atendidos.
- El turno médico es solamente asignado por la recepcionista.
- El estomatólogo es la única persona que puede realizar la consulta estomatológica a un paciente.
- Al terminar un tratamiento de un paciente el estomatólogo debe anotar el gasto de material utilizado.
- Los pacientes atendidos por urgencia no se les registra en la historia clínica pero si en parte diario y gasto de material del estomatólogo que lo atiende.

### 3.2.3 Actores del negocio

Una vez que se hayan identificado los procesos de negocio, es preciso encontrar los actores involucrados en su realización. Cada uno de estos actores del negocio juega un rol cuando colabora con otros para llevar a cabo las actividades que conforman dicho proceso del negocio. (23)

Un actor del negocio es cualquier individuo, grupo, organización, máquina o sistema de información externo que interactúa con el negocio. El término actor significa el rol que algo o alguien juega cuando interactúa con el negocio para beneficiarse de sus resultados. De acuerdo con esta idea un actor del negocio representa un tipo particular de usuario del negocio más que un usuario físico, ya que varios usuarios físicos pueden realizar el mismo papel en relación al negocio, o sea, ser instancias de un mismo actor.

Actores del negocio y su justificación:

Actores del Negocio	Justificación
Paciente	El paciente que puede ser estudiante, profesor o trabajador de la UCI, es el que inicia las acciones que dan lugar a una consulta estomatológica, es el que solicita y recibe los servicios brindados y al mismo tiempo se beneficia con los resultados de dichos procesos.

Tabla 1. Descripción de los actores del negocio

### 3.2.4 Trabajadores del negocio

Un trabajador define el comportamiento y las responsabilidades de un individuo que actúa en el negocio realizando una o varias actividades, interactuando con otros trabajadores del negocio y manipulando entidades del negocio.



Trabajadores del negocio y su justificación:

Trabajadores	Justificación
Jefe de Departamento	Es el encargado de la dirección del Departamento de Estomatología.
Estomatólogo	Es el encargado de atender al paciente en una consulta estomatológica.
Asistente	Es el encargado de asistir a estomatólogo y preparar los materiales a utilizar.
Recepcionista	Es el encargado de admisión y archivo.

**Tabla 2. Descripción de los trabajadores del negocio**

### 3.2.5 Diagrama de Casos de Uso del Negocio

Para tener una visión general de los diferentes procesos de negocio, se construye un diagrama de casos de uso del negocio, en el cual aparece cada proceso del negocio como un caso de uso. El diagrama de casos de uso del negocio representa gráficamente los procesos del negocio y su interacción con los actores del negocio y además permite mostrar los límites y el entorno de una organización bajo estudio.

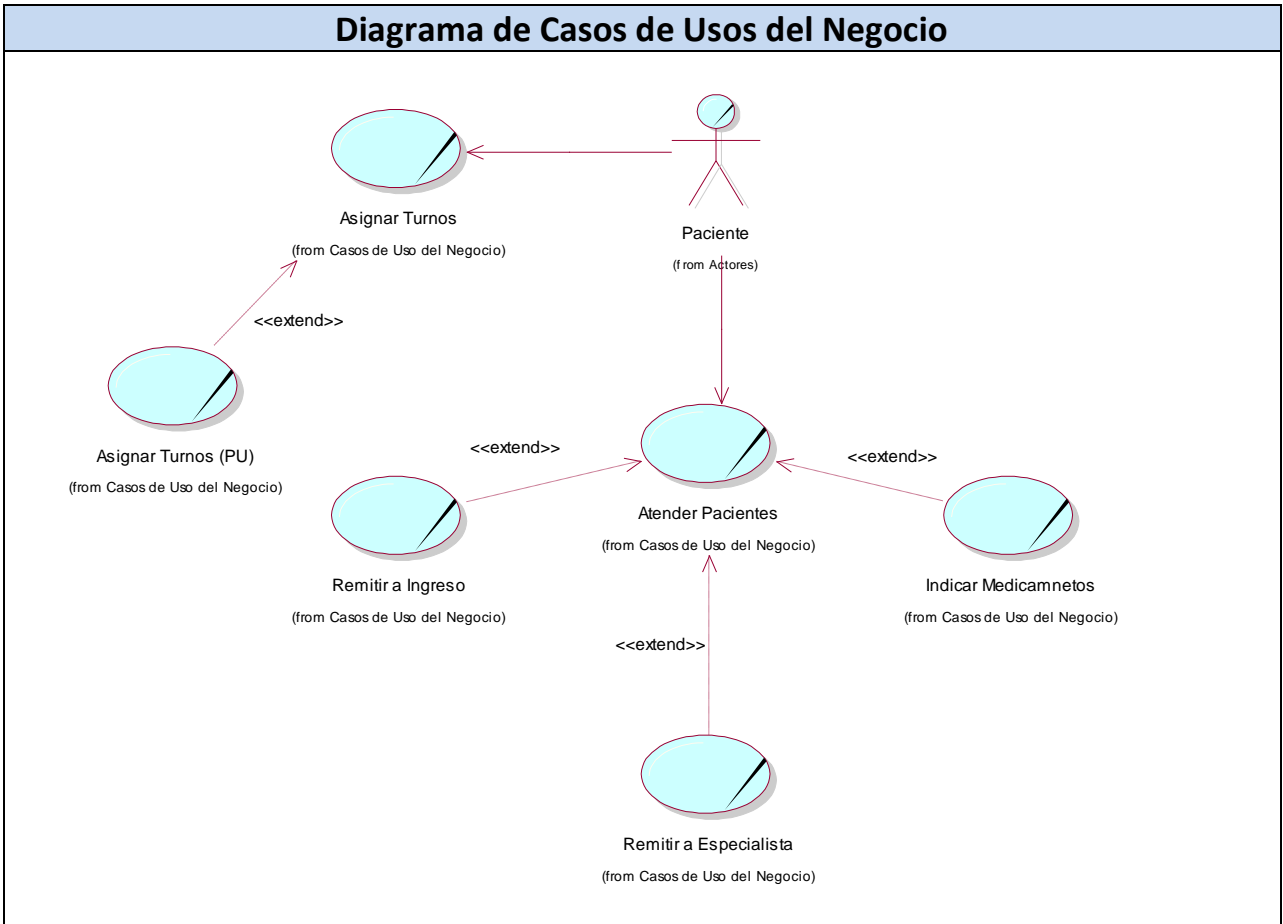


Figura 1. Diagrama de Casos de Usos del Negocio

### 3.2.6 Descripción textual de los Casos de Uso de Negocio

#### 3.2.6.1 Caso de Uso “Asignar Turnos”

<b>Nombre del Caso de Uso</b>	<b>Asignar Turnos</b>	
<b>Actores</b>	Paciente (Inicia).	
<b>Propósito</b>	Permitir que el paciente pueda recibir un turno médico para atenderse con el estomatólogo.	
<b>Resumen</b>	El caso de uso inicia cuando el paciente llega al departamento de estomatología y solicita un turno médico.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>	<b>Respuestas del proceso del negocio</b>	
1. El paciente llega al dpto. de estomatología para solicitar un turno.	2. La recepcionista solicita nombre de la facultad a la que pertenece	
3. El paciente le da el nombre de su facultad.	4. La recepcionista verifica que el doctor que atiende a su facultad no esté de urgencia.	
	5. La recepcionista verifica la disponibilidad de turnos del doctor.	
	6. La recepcionista solicita los datos esenciales del paciente para hacer la reservación del turno medico.	
7. El paciente brinda los datos solicitados.	8. La recepcionista completa el formulario de reservación, asignándole de esta manera un turno al paciente.	
	9. La recepcionista informa al paciente que debe esperar a que le corresponda su turno.	
10. El paciente se incorpora a la sala de espera.		
<b>Curso alternativo de los eventos</b>		
<b>Acciones del actor</b>		
Acción 4	La recepcionista informa al paciente que el doctor que atiende a su facultad esta de urgencia y por lo tanto debe volver otro día. Pasa a la Acción 7.	
Acción 5	La recepcionista informa al paciente que no hay turnos disponibles y por lo tanto debe volver otro día. Pasa a la Acción 7.	
Acción 7	El paciente se retira de la clínica de estomatología.	
<b>Prioridad</b>	Critico	
<b>Mejoras</b>		
<b>Otros</b>		

Tabla 3. Descripción textual del caso de uso “Asignar Turnos”

## 3.2.6.2 Caso de uso “Asignar Turnos (PU)”

<b>Nombre del Caso de Uso</b>	<b>Asignar Turnos (PU).</b>	
<b>Actores</b>	Paciente (Inicia).	
<b>Propósito</b>	Permitir que el paciente pueda recibir un turno médico por urgencia para atenderse con el estomatólogo.	
<b>Resumen</b>	El caso de uso inicia cuando el paciente llega al departamento de estomatología y solicita un turno medico por urgencia.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>		<b>Respuestas del proceso del negocio</b>
1. El paciente llega al dpto. de estomatología para solicitar un turno por urgencia.		2. La recepcionista solicita los datos necesarios para hacer la reservación del turno medico por urgencia.
3. El paciente brinda los datos solicitados.		4. La recepcionista completa el formulario de reservación, asignándole de esta manera un turno al paciente.
		5. La recepcionista informa al paciente que debe esperar a que le corresponda su turno.
6. El paciente se incorpora a la sala de espera.		
<b>Curso alternativo de los eventos</b>		
<b>Acciones del actor</b>		
<b>Prioridad</b>	Critico	
<b>Mejoras</b>		
<b>Otros</b>		

Tabla 4. Descripción textual del Caso de uso “Asignar Turnos (PU)”

## 3.2.6.3 Caso de uso “Atender Pacientes”

<b>Nombre del Caso de Uso</b>	<b>Atender Pacientes</b>	
<b>Actores</b>	Paciente (Inicia).	
<b>Propósito</b>	Permitir que el estomatólogo atienda a un paciente.	
<b>Resumen</b>	El caso de uso inicia cuando el paciente llega a atenderse con el estomatólogo, si el paciente ya tiene historia clínica se busca y se actualizan sus datos o de lo contrario se crea una historia clínica nueva con los datos necesarios del paciente.	
<b>Curso normal de los eventos</b>		
<b>Acciones del actor</b>		<b>Respuestas del proceso del negocio</b>
1- El paciente solicita atención.		2- El estomatólogo identifica al paciente



	3- Si el paciente es interno de la UCI el estomatólogo busca la historia clínica.
	4- Si el paciente tiene historia clínica el estomatólogo revisa la HC existente.
	5- Modifica la historia clínica según la atención que se le dio al paciente.
	6- Guarda la historia clínica del paciente.
7- El paciente recibe atención.	
8- El paciente se retira.	9- El estomatólogo elabora el parte diario
	10- El estomatólogo elabora el informe de gasto material.
<b>Curso alternativo de los eventos</b>	
<b>Acciones del actor</b>	
	3- Si el paciente no es de la UCI, es decir externo el estomatólogo lo atiende.
4- El paciente externo recibe atención.	
5- El paciente se retira	6- El estomatólogo elabora el parte diario.
	7- El estomatólogo elabora el informe de gasto material
<b>Prioridad</b>	Critico
<b>Mejoras</b>	
<b>Otros</b>	

Tabla 5. Descripción textual Caso de Uso “Atender Pacientes”

### 3.2.6.4 Caso de uso “Indicar Medicamentos”

<b>Nombre del Caso de Uso</b>	<b>Indicar Medicamentos.</b>
<b>Actores</b>	Paciente (Inicia).
<b>Propósito</b>	Permitir que el estomatólogo indique medicamento al paciente si lo necesita.
<b>Resumen</b>	El caso de uso inicia cuando el paciente se está atendiendo con el estomatólogo y este reconoce que el paciente necesita medicamentos para mejorarse.
<b>Curso normal de los eventos</b>	
<b>Acciones del actor</b>	<b>Respuestas del proceso del negocio</b>
1- El paciente solicita atención.	2- El estomatólogo atiende al paciente.
	3- El estomatólogo reconoce si el paciente necesita de medicamentos.
	4- El estomatólogo confecciona una receta médica.
	5- El estomatólogo entrega la receta médica al

	paciente y lo manda a farmacia.
6. El paciente recibe la receta médica.	
<b>Curso alternativo de los eventos</b>	
<b>Acciones del actor</b>	
	3- Si el paciente no necesita de medicamentos el estomatólogo informa al paciente que ha concluido la consulta. Pasa a la Acción 4.
4-El paciente se retira.	
<b>Prioridad</b>	Secundario.
<b>Mejoras</b>	
<b>Otros</b>	

Tabla 6. Descripción textual del Caso de Uso “Indicar Medicamentos”

### 3.2.6.5 Caso de uso “Remitir a Especialista”

<b>Nombre del Caso de Uso</b>	<b>Remitir a Especialista.</b>
<b>Actores</b>	Paciente (Inicia).
<b>Propósito</b>	Permitir que el estomatólogo remita un paciente al especialista.
<b>Resumen</b>	El caso de uso inicia cuando el paciente se está atendiendo con el estomatólogo y este detecta alguna enfermedad que debe ser tratada con especialista.
<b>Curso normal de los eventos</b>	
<b>Acciones del actor</b>	<b>Respuestas del proceso del negocio</b>
1- El paciente solicita atención.	2- El estomatólogo atiende al paciente.
	3- El estomatólogo detecta si el paciente tiene alguna enfermedad que sea tratada con un especialista.
	4- El estomatólogo confecciona el formulario de remisión a un especialista.
	5- El estomatólogo entrega al paciente la orden de remisión al especialista.
6. El paciente recibe la orden de remisión.	
<b>Curso alternativo de los eventos</b>	
<b>Acciones del actor</b>	
	3- Si el paciente no tiene enfermedad tratada con especialista el estomatólogo concluye la consulta. Pasa a la Acción 4.
4-El paciente se retira.	



<b>Prioridad</b>	Secundario.
<b>Mejoras</b>	
<b>Otros</b>	

Tabla 7. Descripción textual del Caso de Uso “Remitir a Especialista”

### 3.2.6.6 Caso de Uso “Remitir a Ingreso”

<b>Nombre del Caso de Uso</b>	<b>Remitir a Especialista.</b>
<b>Actores</b>	Paciente (Inicia).
<b>Propósito</b>	Permitir que el estomatólogo remita un paciente al especialista.
<b>Resumen</b>	El caso de uso inicia cuando el paciente se está atendiendo con el estomatólogo y este determina que el paciente debe ser tratado con seguimiento y lo remite a ingreso hospitalario.
<b>Curso normal de los eventos</b>	
<b>Acciones del actor</b>	<b>Respuestas del proceso del negocio</b>
1- El paciente solicita atención.	2- El estomatólogo atiende al paciente.
	3- El estomatólogo determina que el paciente debe ser ingresado para seguimiento.
	4- El estomatólogo confecciona el formulario de remisión a ingreso.
	5- El estomatólogo entrega al paciente la orden de remisión a ingreso.
6. El paciente recibe la orden de remisión.	
<b>Curso alternativo de los eventos</b>	
<b>Acciones del actor</b>	
	3- Si el paciente no tiene necesidad de ser ingresado el estomatólogo concluye con la consulta. Pasa a la Acción 4.
4-El paciente se retira.	
<b>Prioridad</b>	Secundario.
<b>Mejoras</b>	
<b>Otros</b>	

Tabla 8. Descripción textual del Caso de Uso “Remitir a Ingreso”

### **3.2.7 Diagramas de Actividades para cada Caso de Uso del Negocio**

Un diagrama de actividad demuestra la serie de actividades que deben ser realizadas en un proceso del negocio, así como las distintas rutas que pueden irse desencadenando. Este es dividido en canales, donde cada canal representa el actor que está llevando a cabo la actividad y muestra cómo se utilizan las entidades del negocio.

Un diagrama de actividades es un caso especial de un diagrama de estados en el cual casi todos los estados son estados de acción y casi todas las transiciones son enviadas al terminar la acción ejecutada en el estado anterior. Puede dar detalle a un caso de uso, un objeto o un mensaje en un objeto. Sirven para representar transiciones internas, sin hacer mucho énfasis en transiciones o eventos externos.



### 3.2.7.1 Diagrama de Actividades para el CUN “Asignar Turnos”

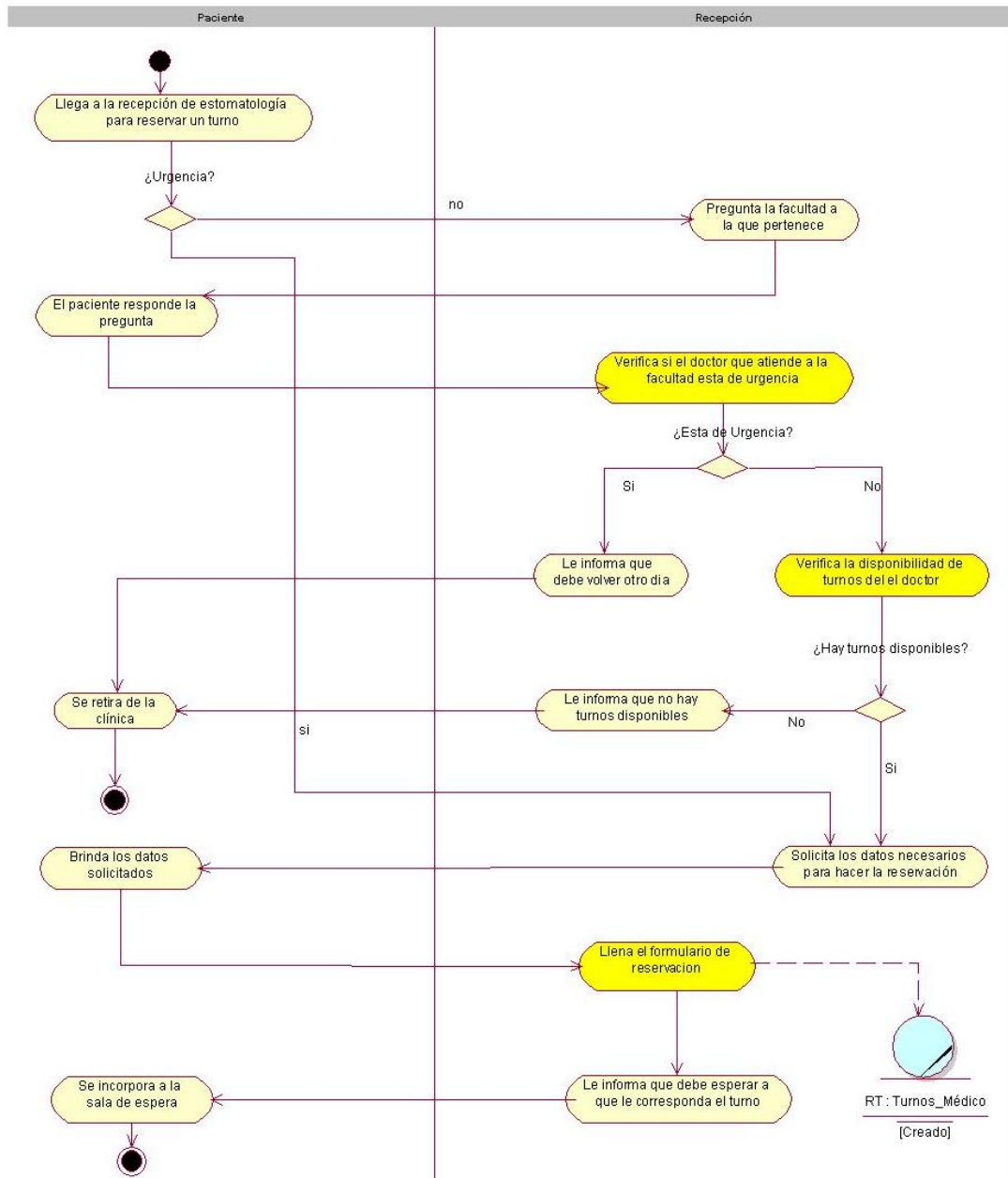


Figura 2. Diagrama de actividades del Caso de Uso “Asignar Turno”

### 3.2.7.2 Diagrama de Actividades para el CUN “Asignar Turnos PU”

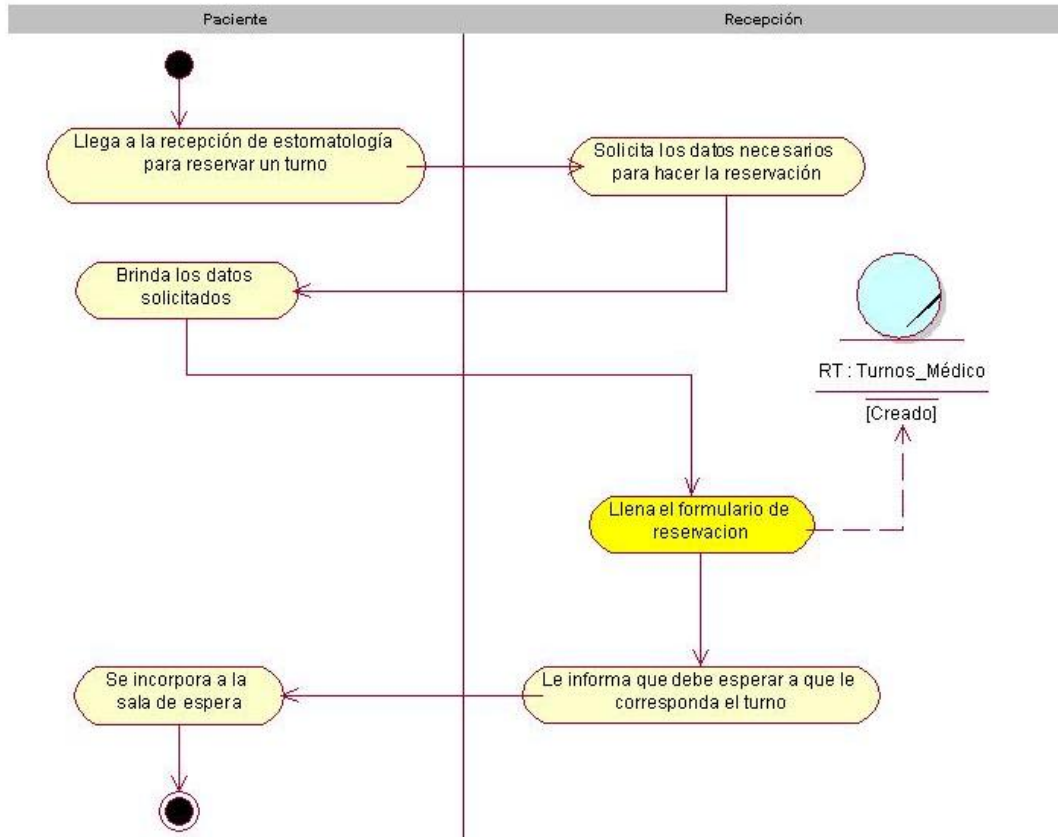


Figura 3. Diagrama de actividades del caso de uso “Asignar Turno PU”

### 3.2.7.3 Diagrama de Actividades para el CUN “Atender Paciente”

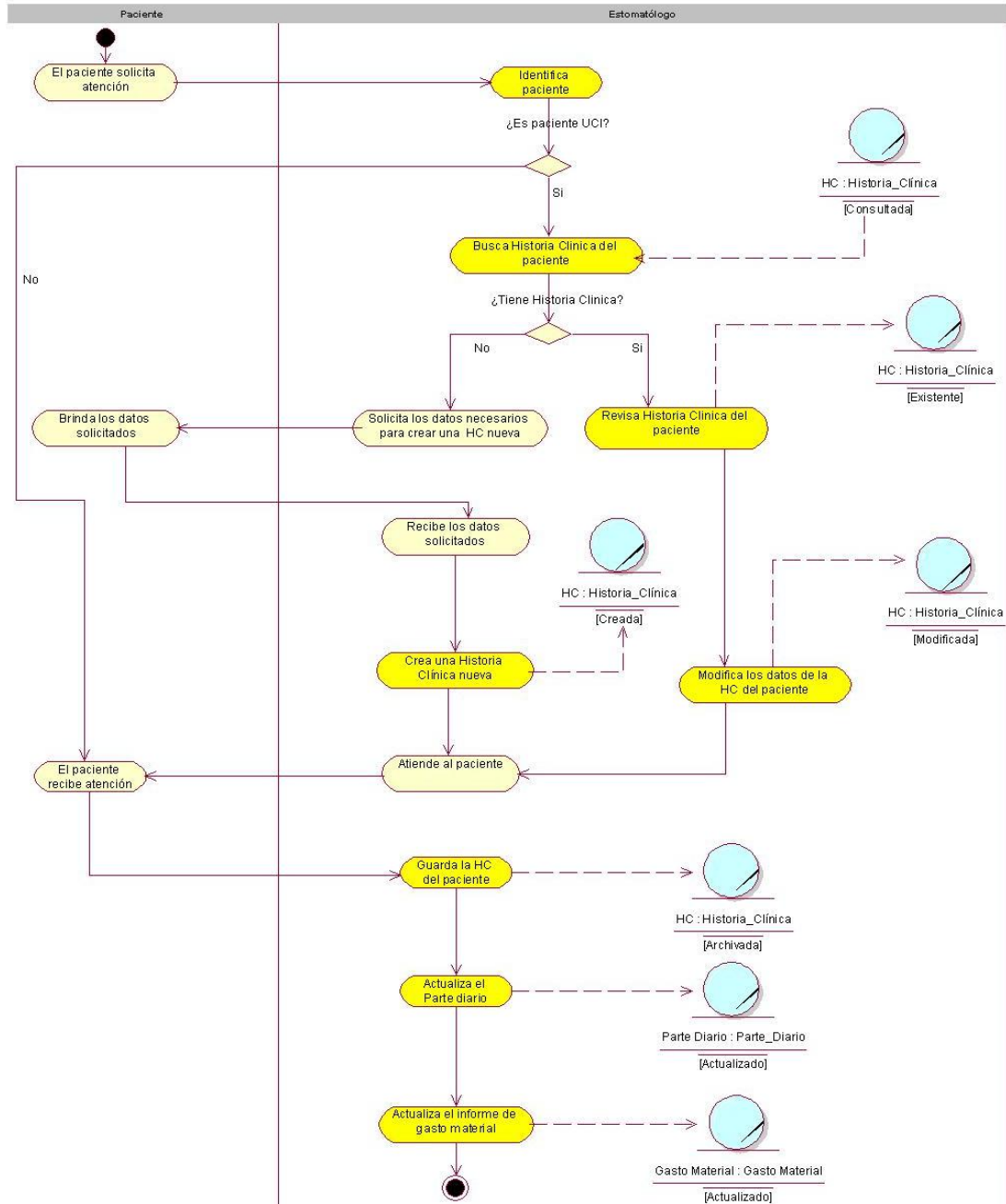


Figura 4. Diagrama de actividades del Caso de Uso “Atender Pacientes”

### 3.2.7.4 Diagrama de Actividades para el CUN “Indicar Medicamentos”

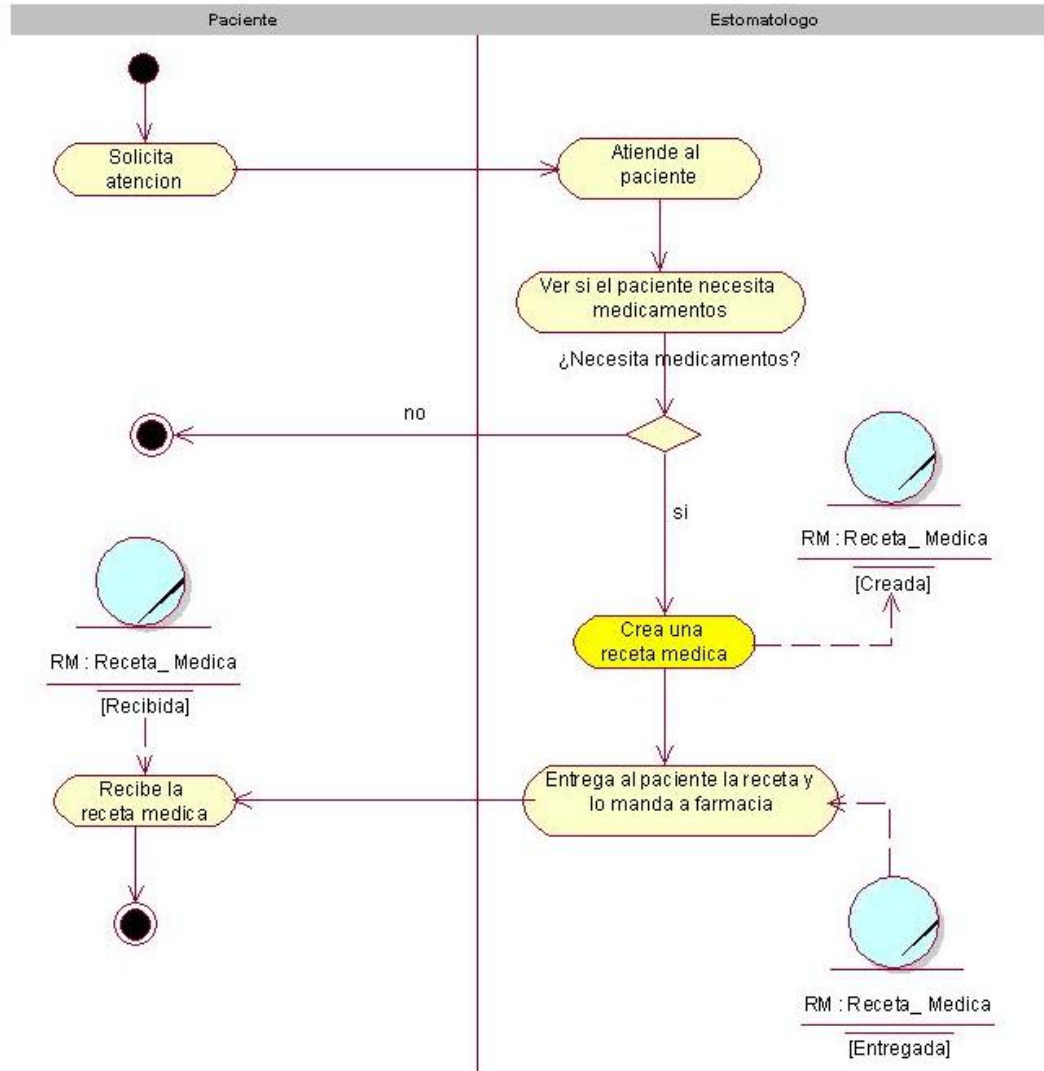


Figura 5. Diagrama de actividades del Caso de Uso “Indicar Medicamentos”

### 3.2.7.5 Diagrama de Actividades para el CUN “Remitir a Especialista”

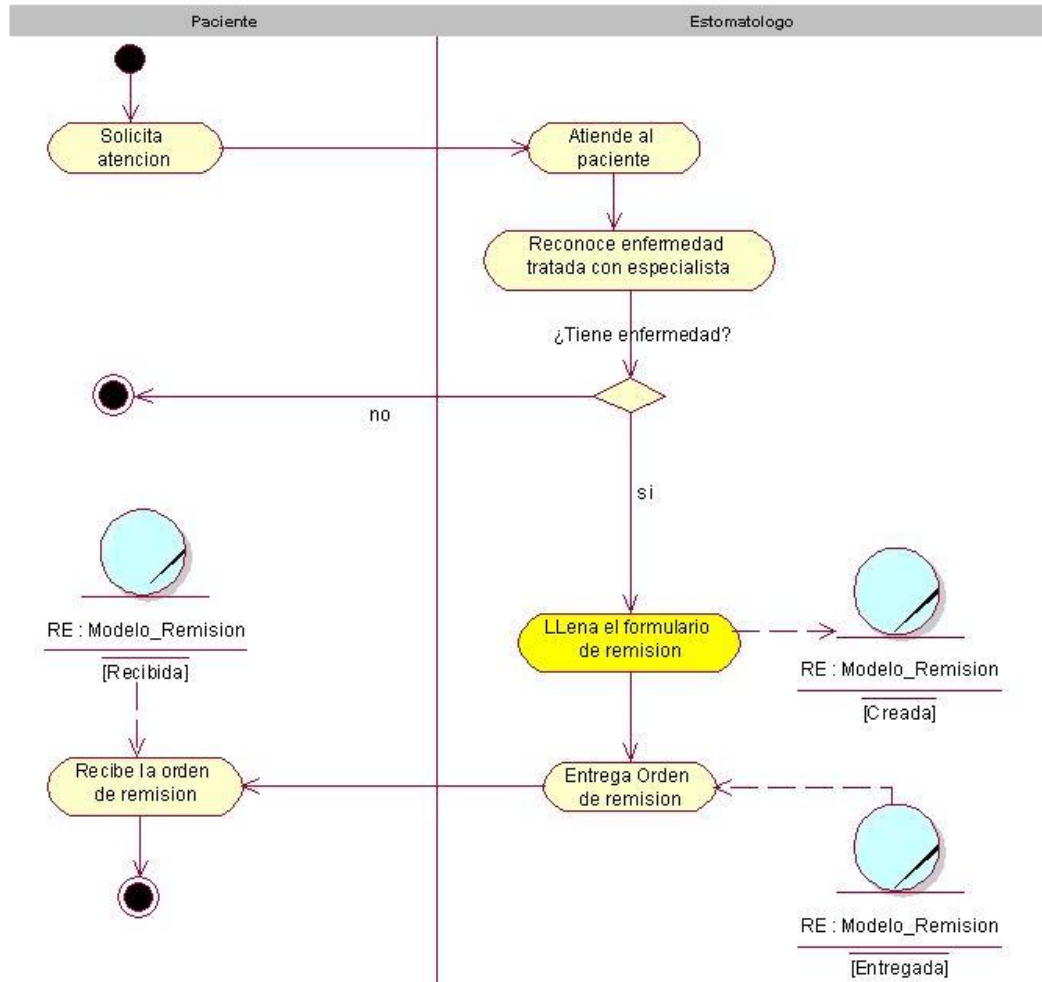


Figura 6. Diagrama de actividades del Caso de Uso “Remitir a Especialista”

### 3.2.7.6 Diagrama de Actividades para el CUN “Remitir a Ingreso”

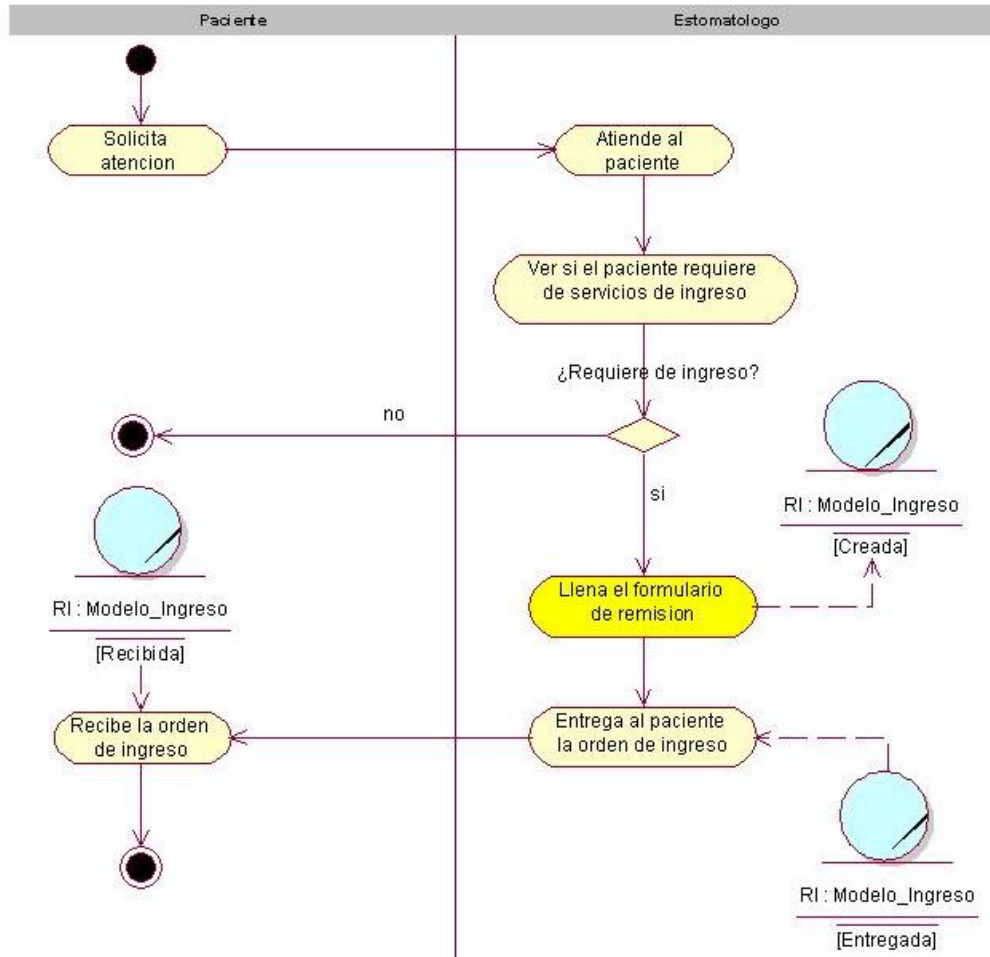


Figura 7. Diagrama de actividades del Caso de Uso “Remitir a Ingreso”

### **3.3 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.

A continuación las funcionalidades que debe cumplir el sistema propuesto:

#### **R1- Autenticar Usuarios.**

#### **R2- Gestionar Usuarios del Sistema.**

- 2.1- Crear usuarios.
- 2.2- Modificar usuarios.
- 2.3- Eliminar usuarios.

#### **R3- Reservar Turnos Médicos.**

- 3.1- Reservar Turnos médicos pacientes uci internos
- 3.2- Reservar Turnos médicos pacientes uci externos
- 3.3- Reservar Turnos médicos pacientes externos

#### **R4- Mostrar Turnos Asignados por Doctor.**

#### **R5- Gestionar Historias Clínicas.**

- 5.1- Crear historias clínicas datos generales.
- 5.2- Modificar historias clínicas datos generales.
- 5.3- Visualizar historias clínicas datos generales.
- 5.4- Eliminar historias clínicas.

#### **R6- Gestionar Historias Clínicas Dentigrama.**

- 6.1- Crear historias clínicas dentigrama.
- 6.2- Modificar historias clínicas dentigrama.
- 6.3- Visualizar historias clínicas dentigrama.

#### **R7- Gestionar Historias Clínicas Evolución.**

7.1- Crear historias clínicas evolución.

7.2- Modificar historias clínicas evolución.

7.3- Visualizar historias clínicas evolución.

**R8- Indicar Medicamentos.**

8.1- Crear solicitud de medicamentos.

9.2- Archivar solicitud de medicamentos.

9.3- Imprimir solicitud de medicamentos.

**R9- Remitir a un Especialista.**

9.1- Crear orden de remisión a especialista.

9.2- Archivar orden de remisión a especialista.

9.3- Imprimir orden de remisión a especialista.

**R10- Remitir a Ingresos.**

10.1- Crear orden de remisión a ingreso.

10.2- Archivar orden de remisión a ingreso.

10.3 Imprimir orden de remisión a ingreso.

**R11- Confeccionar Partes Diarios.**

11.1- Generar partes diarios.

11.2- Archivar partes diarios.

**R12-Confeccionar Partes de Gasto Material.**

12.1- Generar Informe de Gasto Material.

12.2- Archivar Informe de Gasto Material.

**R13- Gestionar Materiales Educativos.**

13.1- Publicar materiales educativos.

13.2- Eliminar materiales educativos.



## **R14- Visualizar Materiales Educativos.**

### 14.1- Visualizar Materiales.

## **3.4 Requerimientos No Funcionales**

Los requerimientos no funcionales especifican propiedades del sistema como restricciones de ambiente y desarrollo, performance, dependencias de plataformas, y confiabilidad.

A continuación se muestran los requerimientos no funcionales:

### **• Apariencia o interfaz externa**

La interfaz no contiene 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.

### **• Usabilidad**

El sistema ha sido diseñado de tal forma que permite a los usuarios tener control en todo momento del sitio web, evitando para eso la sobrecarga de información y para cada proceso una muestra a través de mensaje de las acciones hechas por los usuarios. Los colores usados permiten el acceso a usuarios con problemas de distinción de color, se reduce el tiempo de latencia, además, requiriendo un mínimo proceso de aprendizaje.

### **• Rendimiento**

La disponibilidad de trabajo en red contra el servidor es constante. Se garantiza que la respuesta a solicitudes de los usuarios del sistema sea en un período de tiempo breve (de segundos) para evitar la acumulación de trabajo por parte de los responsables y público en los puntos de admisión. El sistema deberá de ser lo más estable y confiable posible.

### **• Soporte**

Se requiere que el producto reciba mantenimiento ante cualquier fallo que ocurra. El sistema es de fácil instalación.

- **Software**

Para el funcionamiento del sistema en el servidor es necesario el S.O Windows 98 o superior, Linux o Unix, en sus versiones de S.O servidores. Para el funcionamiento del sistema en las terminales cliente es necesario el S.O Windows 98 o superior.

- **Hardware**

Para la utilización de la aplicación por los usuarios, estos requieren de una microcomputadora conectada a la red con las siguientes características:

Microprocesador Pentium II o superior

Memoria RAM: 128 Mb o superior

Disco Duro: 20 Gb o superior

Periféricos: Teclado, impresora, tarjeta de Red

Para la Base de datos y la aplicación se requiere un servidor con las siguientes características:

Microprocesador Pentium II o superior

Memoria RAM: 64 Mb o superior

Disco Duro: 20 Gb o superior

Periféricos: Teclado, tarjeta de Red

- **Portabilidad**

El producto es usado bajo los S.O Windows. El producto corre sobre una plataforma Web, codificada en “PHP5” y sus sistemas de bases de datos en MySQL.

- **Seguridad**

El sistema se encarga de controlar los diferentes niveles de acceso y funcionalidad de usuarios al sitio, de identificar al usuario antes de que pueda realizar cualquier acción sobre el sistema. Garantiza que la información sea vista únicamente por quien tiene derecho a verla. Se hacen 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.

- **Confidencialidad**

Toda la información está protegida del acceso no autorizado, los usuarios solo pueden ver o transformar la información a la que tienen acceso.

- **Disponibilidad**

Se garantiza a los usuarios del sistema el acceso a la información solicitada en todo momento (si tiene permiso para ello).

- **Restricciones en el diseño y la implementación**

Es una aplicación Web desarrollada con la tecnología para creación de páginas Web dinámicas PHP5 y base de datos en MySQL.

- **Confiabilidad**

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

### 3.5 Descripción del Sistema Propuesto

#### 3.5.1 Descripción de los actores del sistema

Un actor del sistema es un rol que un usuario juega con respecto al sistema. Es importante destacar que un actor no necesariamente representa a una persona en particular, sino más bien la labor que realiza frente al sistema, es decir que un equipo o un sistema automatizado, son parte del sistema, y pueden intercambiar información con él o ser recipientes pasivos de información.

Actores del sistema y su justificación.

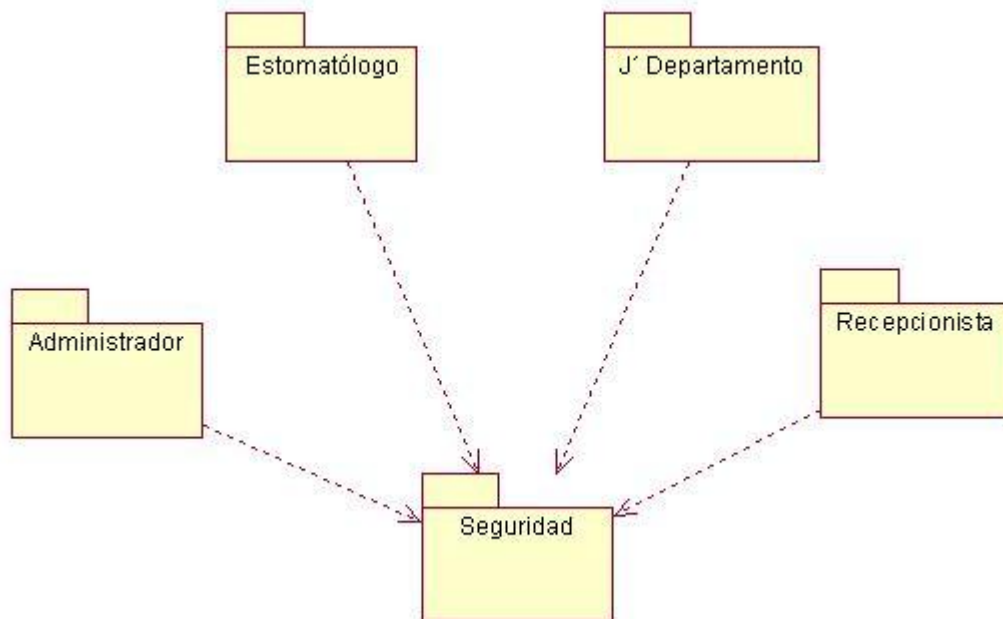
Actores	Justificación
Recepcionista	Es el que asigna turnos médicos a los pacientes.
Estomatólogo	Es el que atiende al paciente en la consulta, donde crea a su vez la historia clínica de cada paciente y luego la archiva.
Jefe de Departamento	Es una especialización del estomatólogo, que además de dirigir el departamento de estomatología puede asignar población y publicar

	materiales educativos, y se comporta como un estomatólogo más.
Administrador	Es el que va a administrar la aplicación.

**Tabla 9. Descripción de los Actores del Sistema**

### 3.5.2 Casos de Uso del Sistema

El diagrama de casos de uso del sistema representa la forma en cómo un actor opera con el sistema en desarrollo, además de la forma, tipo y orden en como los elementos interactúan. Un diagrama de casos de uso muestra, por tanto, los distintos requisitos funcionales que se esperan de una aplicación o sistema y cómo se relaciona con su entorno (usuarios u otras aplicaciones).



**Figura 8. Paquetes del Modelo de Casos de Usos del Sistema**

### 3.5.2.1 Paquete Seguridad

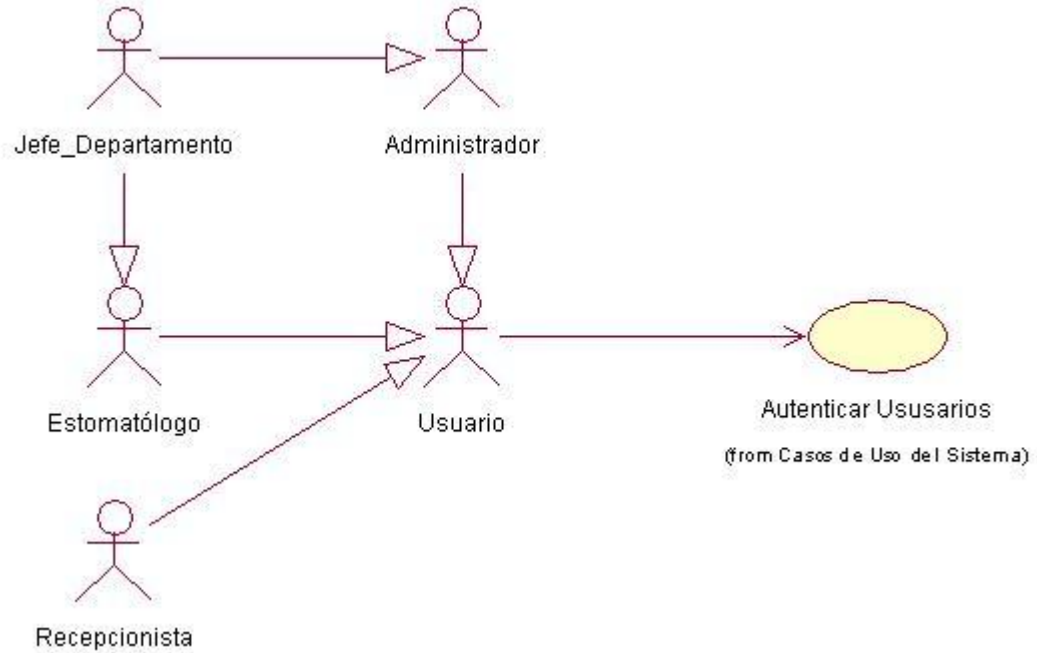


Figura 9. Caso de uso asociado al Paquete de Seguridad del Sistema

### 3.5.2.2 Paquete Administrador

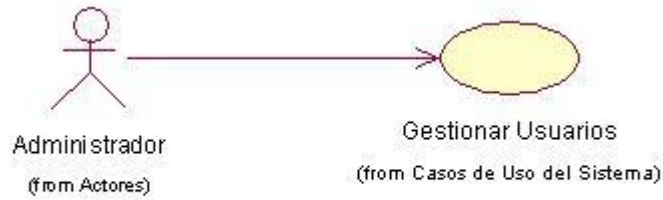


Figura 10. Caso de uso asociado Paquete de Administrador del Sistema

### 3.5.2.3 Paquete Jefe de Departamento

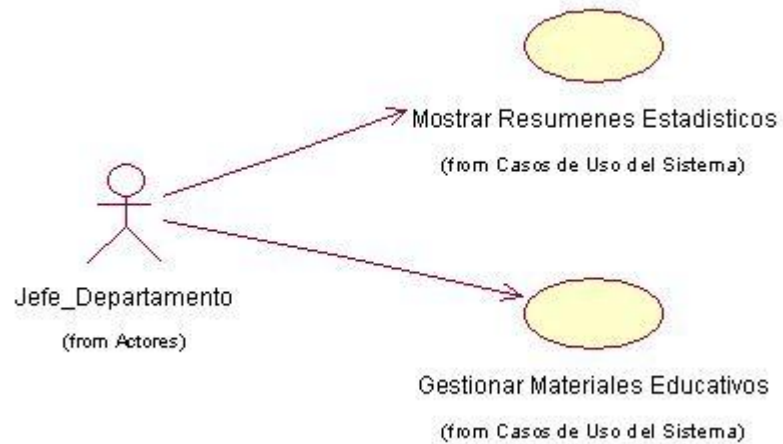


Figura 11. Casos de uso asociado al Paquete de Jefe de Departamento del Sistema

### 3.5.2.4 Paquete Recepcionista

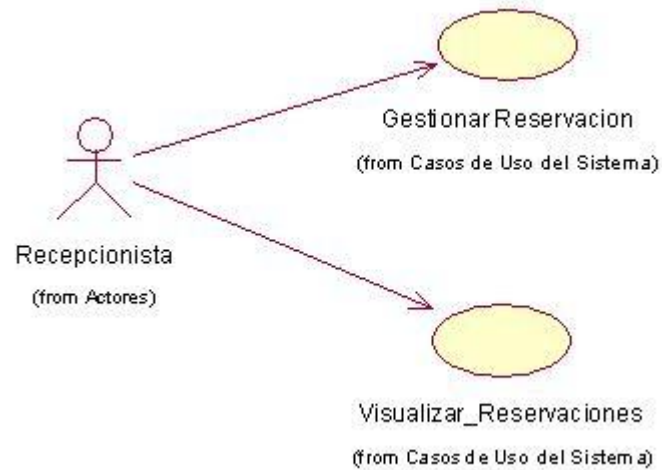


Figura 12. Casos de uso asociado al Paquete Recepcionista del Sistema

### 3.5.2.5 Paquete Estomatológico

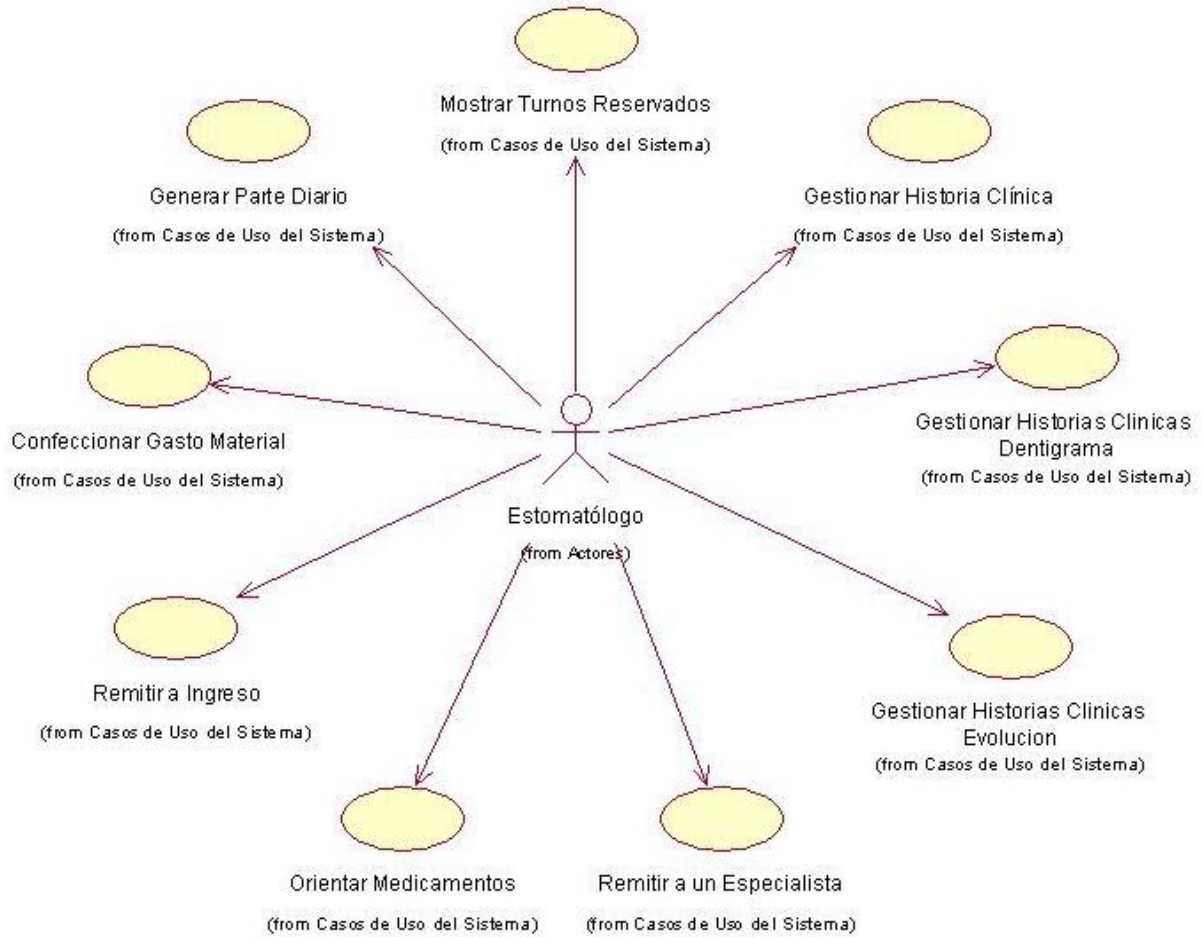


Figura 13. Casos de uso asociado al Paquete Estomatológico del Sistema

### 3.5.3 Descripción textual de casos de usos del sistema

#### 3.5.3.1 Caso de Uso “Autenticar Usuarios”

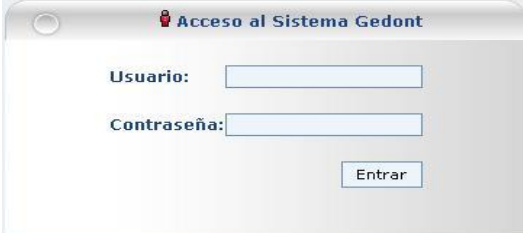
<b>Nombre del Caso de Uso</b>	<b>Autenticar Usuarios.</b>	
<b>Actores</b>	Usuario (Inicia).	
<b>Propósito</b>	Permitir a los usuarios autenticarse en el sistema.	
<b>Resumen</b>	Comienza cuando el usuario accede al sistema para poder navegar en las páginas a las cuales tienen permiso, mostrándole el sistema la opción de autenticarse.	
<b>Referencias</b>	R-1	
<b>Precondiciones</b>	El usuario debe estar registrado en el sistema.	
<b>Pos condiciones</b>	Para las siguientes acciones: Autenticar: El usuario esta autenticado.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El usuario accede al sistema para entrar en él.	1.1 El sistema muestra una interfaz para entrar los datos de autenticación.	
2. El usuario entra los datos para autenticarse.	2.1 El sistema valida los datos entrados y muestra una ventana de bienvenida al sistema.	
<b>Curso alternativo</b>		
	2.1 Emite un mensaje indicando que el usuario o la contraseña entrados están incorrectos.	
<b>Prototipo</b>		
		
<b>Prioridad:</b>	Crítico	

Tabla 10. Descripción detallada del Caso de Uso del sistema “Autenticar Usuarios”



## 3.5.3.2 Caso de Uso “Gestionar Usuarios”

<b>Nombre del Caso de Uso</b>	<b>Gestionar Usuarios.</b>	
<b>Actores</b>	Administrador (Inicia).	
<b>Propósito</b>	Permitir al administrador gestionar los usuarios del sistema.	
<b>Resumen</b>	Comienza cuando el administrador selecciona la opción de gestionar usuarios, mostrándole el sistema un conjunto de acciones que puede realizar sobre el usuario, de estas acciones puede seleccionar registrar, modificar o eliminar usuarios.	
<b>Referencias</b>	R-2	
<b>Precondiciones</b>	El administrador debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para las siguientes acciones: Registrar: Se registra un nuevo usuario en el sistema. Modificar: Se modifica un usuario del sistema. Eliminar: Se elimina un usuario del sistema.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El administrador selecciona registrar, modificar o eliminar usuarios.	1.1 El sistema ejecuta alguna de las siguientes opciones: a) Si selecciona la opción de registrar usuario ir a la sección de “Registrar Usuario” b) Si selecciona la opción de modificar usuario ir a la sección de “Modificar Usuarios” c) Si selecciona la opción de eliminar usuario ir a la sección de “Eliminar Usuario”	
<b>Sección “Registrar Usuario”</b>		
<b>Acciones del actor</b>	<b>Respuesta del Sistema</b>	
2. El administrador entra los datos necesarios para registrar un nuevo usuario.	2.1 El sistema valida los datos y muestra el resultado de la validación.	
3. El administrador pulsa el botón “Adicionar” para adicionar un usuario al sistema	3.1 El sistema inserta un nuevo usuario en el sistema	
<b>Curso alternativo</b>		
	2.1 Emite un mensaje indicando que el usuario no existe en el dominio.	
<b>Prototipo</b>		

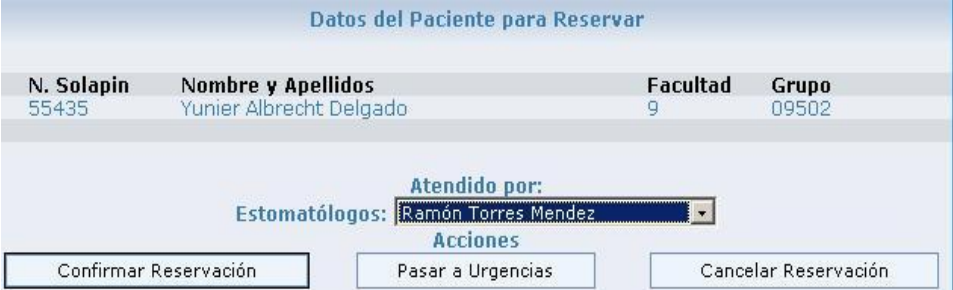


<p style="text-align: center;"><b>Registrar Usuarios al Sistema</b></p> <p>Solapín: <input type="text"/> </p> <p>Nombre: <input type="text"/></p> <p>P Apellido: <input type="text"/></p> <p>S Apellido: <input type="text"/></p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 10%;">Nombre</th> <th style="width: 20%;">P Apellido</th> <th style="width: 20%;">S Apellido</th> <th style="width: 20%;">Usuario</th> <th style="width: 20%;">Privilegios</th> </tr> </thead> <tbody> <tr> <td> Yunier</td> <td>Albrecht</td> <td>Delgado</td> <td>yalbrecht</td> <td>Jefe Dpto </td> </tr> </tbody> </table>		Nombre	P Apellido	S Apellido	Usuario	Privilegios	Yunier	Albrecht	Delgado	yalbrecht	Jefe Dpto					
Nombre	P Apellido	S Apellido	Usuario	Privilegios												
Yunier	Albrecht	Delgado	yalbrecht	Jefe Dpto												
<b>Sección "Modificar Usuario"</b>																
<b>Acciones del Actor</b>	<b>Respuesta del sistema</b>															
2. El administrador cambia los privilegios del usuario que desea modificar.	2.1 El sistema valida los datos modificados y guarda la información modificada del usuario.															
<b>Curso Alternativo</b>																
<b>Prototipo</b>																
<p style="text-align: center;"><b>Modificar Usuarios del Sistema</b></p> <p style="text-align: center;">Introduzca el usuario a modificar:</p> <p style="text-align: center;"><input type="text"/> </p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Nombre y Apellidos</th> <th style="width: 20%;">Ocupación</th> <th style="width: 30%;">Acción</th> </tr> </thead> <tbody> <tr> <td>Yunier Albrecht Delgado</td> <td>Jefe Dpto </td> <td></td> </tr> <tr> <td></td> <td>Jefe Dpto</td> <td></td> </tr> <tr> <td></td> <td>Estomatólogo</td> <td></td> </tr> <tr> <td></td> <td>Recepción</td> <td></td> </tr> </tbody> </table>		Nombre y Apellidos	Ocupación	Acción	Yunier Albrecht Delgado	Jefe Dpto			Jefe Dpto			Estomatólogo			Recepción	
Nombre y Apellidos	Ocupación	Acción														
Yunier Albrecht Delgado	Jefe Dpto															
	Jefe Dpto															
	Estomatólogo															
	Recepción															
<b>Sección "Eliminar Usuario"</b>																
<b>Acciones del Actor</b>	<b>Respuesta del sistema</b>															
3. El Administrador entra los datos para buscar el usuario que desea eliminar	3.1 El sistema muestra una interface donde aparece el usuario y la opción de eliminar															
4. El administrador elimina al usuario buscado.	4.1 El sistema elimina al usuario del sistema y muestra un mensaje indicando que la operación se ha realizado satisfactoriamente.															
<b>Curso Alternativo</b>																
	3.1 El sistema muestra un mensaje indicando que el usuario no existe dentro del sistema.															
<b>Prototipo</b>																
<p style="text-align: center;"><b>Eliminar Usuarios del Sistema</b></p> <p style="text-align: center;">Buscar usuario a eliminar</p> <p style="text-align: center;"><input type="text"/> </p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="width: 50%;">Nombre y Apellidos</th> <th style="width: 20%;">Ocupación</th> <th style="width: 30%;">Acción</th> </tr> </thead> <tbody> <tr> <td>Yunier Albrecht Delgado</td> <td>Recepción</td> <td></td> </tr> </tbody> </table>		Nombre y Apellidos	Ocupación	Acción	Yunier Albrecht Delgado	Recepción										
Nombre y Apellidos	Ocupación	Acción														
Yunier Albrecht Delgado	Recepción															
<b>Prioridad:</b>	Crítico															

Tabla 11. Descripción detallada del caso de Uso del sistema “Gestionar Usuario”

## 3.5.3.3 Caso de Uso “Gestionar Reservación”

<b>Nombre del Caso de Uso</b>		<b>Gestionar Reservación.</b>
<b>Actores</b>	Recepcionista (Inicia).	
<b>Propósito</b>	Permitir a la recepcionista gestionar las reservaciones de turnos médicos.	
<b>Resumen</b>	Comienza cuando la recepcionista selecciona la opción de “Reservación”, mostrándole el sistema una interface para registrar la reservación de turnos.	
<b>Referencias</b>	R-3	
<b>Precondiciones</b>	La recepcionista debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para las siguientes acciones: Confirmar Reservación: Se registra una nueva reservación en el sistema. Pasar a Urgencias: Se registra una nueva reservación de urgencia en el sistema Cancelar Reservación: Se cancela la reservación.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. La recepcionista selecciona la opción de “Reservación”.	1.1 El sistema muestra una interfaz para escoger el tipo de reservación que quiere hacer.	
2. La recepcionista escoge la opción de “Paciente UCI”, “Paciente Externo”.	2.1 El sistema ejecuta una de las siguientes acciones: a) Si selecciona Paciente UCI ir a la sección Paciente UCI. b) Si selecciona Paciente Externo ir a la sección de Paciente Externo.	
<b>Sección “Paciente UCI”</b>		
	2.1 El sistema muestra una interfaz para buscar un paciente uci por solapín.	
3. La recepcionista entra el solapín del paciente para buscarlo.	3.1 El sistema muestra una interfaz para reservar turno con los datos del paciente encontrados.	

<p>4. La recepcionista chequea que médico atenderá la nueva reservación y selecciona la opción de “Confirmar Reservación”, “Pasar a Urgencias”, “Cancelar Reservación”.</p>	<p>4.1 El sistema ejecuta alguna de las siguientes acciones:  a) Si selecciona Confirmar Reservación ir a la subsección “Confirmar Reservación”.  b) Si selecciona Pasar a Urgencias ir a la subsección “Pasar a Urgencias”.  c) Si selecciona Cancelar Reservación ir a la subsección “Cancelar Reservación”.</p>
<p><b>Subsección “Confirmar Reservación”</b></p>	
	<p>4.1 El sistema inserta una nueva reservación y muestra un mensaje indicando operación satisfactoria</p>
<p><b>Subsección “Pasar a Urgencias”</b></p>	
	<p>4.1 El sistema inserta una nueva reservación de urgencia y muestra un mensaje indicando que el paciente ha sido pasado a Urgencias.</p>
<p><b>Subsección “Cancelar Reservación”</b></p>	
	<p>4.1 El sistema no inserta ninguna reservación y muestra nuevamente la interfaz inicial para hacer una nueva reservación.</p>
<p><b>Curso Alternativo Sección “Paciente UCI”</b></p>	
	<p>3.1 El sistema muestra un mensaje indicando que el paciente no existe en el dominio UCI y debe ser procesado como Paciente externo.</p>
<p><b>Prototipo</b></p>	
	
<p><b>Sección “Paciente Externo”</b></p>	
	<p>2.1 El sistema muestra una interfaz para entrar los datos de un paciente externo</p>
<p>3. La recepcionista entra los datos requeridos para la reservación</p>	<p>3.1 El sistema muestra una interfaz para reservar turno con los datos del paciente ya insertados.</p>
<p>4. La recepcionista chequea que médico atenderá la nueva reservación y selecciona la opción de “Confirmar Reservación”, “Pasar a Urgencias”, “Cancelar Reservación”.</p>	<p>4.1 El sistema ejecuta alguna de las siguientes acciones:  a) Si selecciona Confirmar Reservación ir a la subsección “Confirmar Reservación”.  b) Si selecciona Pasar a Urgencias ir a la subsección “Pasar a Urgencias”.</p>


	c) Si selecciona Cancelar Reservación ir a la subsección "Cancelar Reservación".
<b>Subsección "Confirmar Reservación"</b>	
	4.1 El sistema inserta una nueva reservación en el sistema y muestra un mensaje indicando operación satisfactoria.
<b>Subsección "Pasar a Urgencias"</b>	
	4.1 El sistema inserta una nueva reservación de urgencia y muestra un mensaje indicando que el paciente ha sido pasado a Urgencias.
<b>Subsección "Cancelar Reservación"</b>	
	4.1 El sistema no inserta ninguna reservación y muestra nuevamente la interfaz inicial para hacer una nueva reservación.
<b>Curso Alternativo Sección "Paciente Externo "</b>	
	3.1 El sistema muestra un mensaje indicando que debe llenar los campos obligatorios.
<b>Prototipo</b>	
	
<b>Prioridad:</b>	Crítico

Tabla 12. Descripción detallada del caso de Uso del sistema "Gestionar Reservación"

### 3.5.3.4 Caso de Uso "Visualizar Turnos Reservados"

<b>Nombre del Caso de Uso</b>	<b>Visualizar Turnos Reservados.</b>
<b>Actores</b>	Estomatólogo (inicia).
<b>Propósito</b>	Permitir al estomatólogo visualizar listado de turnos reservados.
<b>Resumen</b>	Comienza cuando el estomatólogo selecciona la opción de ver listado de turnos asignado a pacientes.
<b>Referencias</b>	R-4
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.
<b>Pos condiciones</b>	Para las siguiente acción: Visualizar Turnos: Se muestra un listado de turnos reservados por doctor.

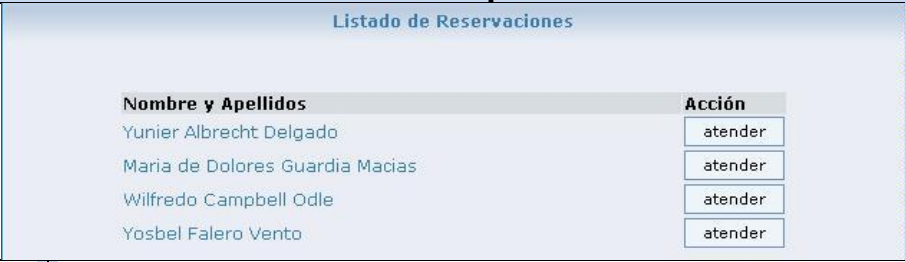
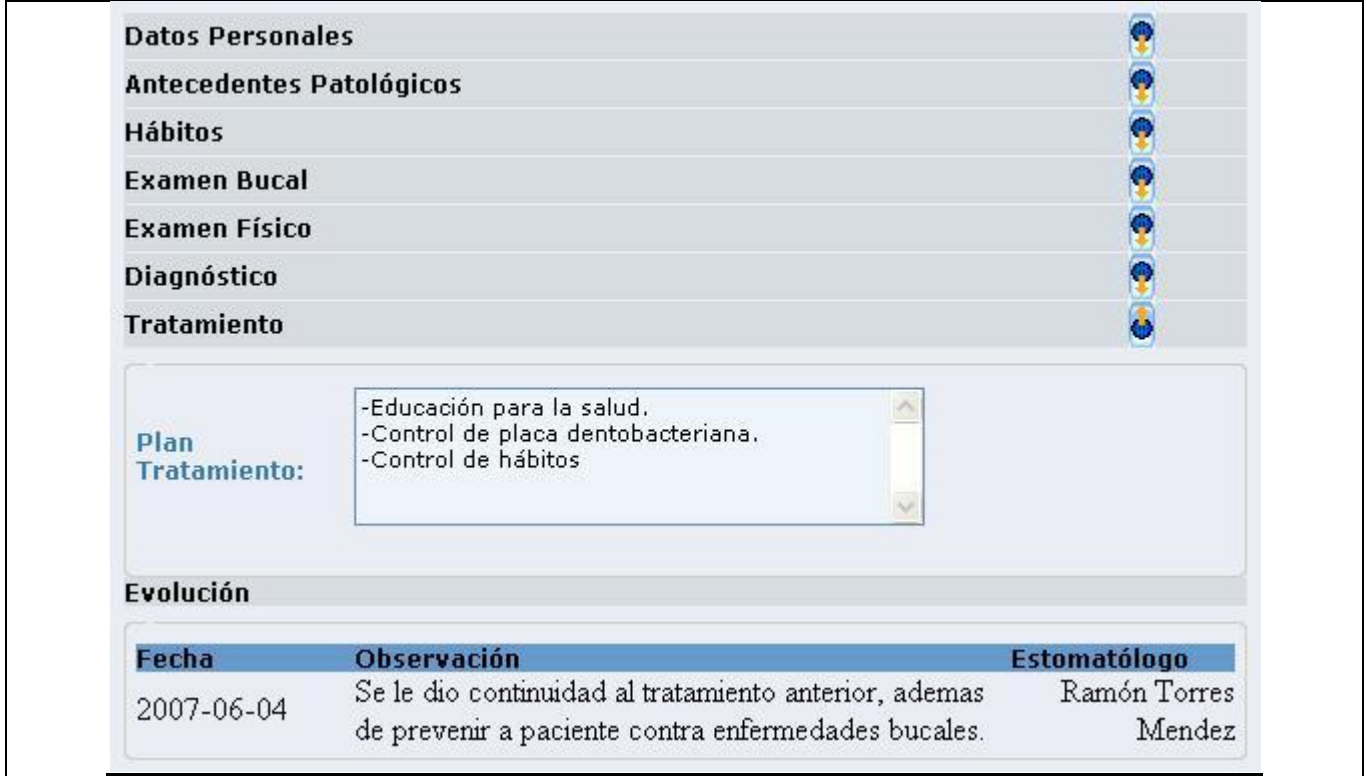
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El estomatólogo solicita ver listado de turnos reservados.	1.1 El sistema muestra una interfaz con el listado de turnos reservados para ese día con ese doctor.
Curso alternativo	
	2.1 De no existir reservación el sistema emite un mensaje indicando que no hay turnos reservados.
Prototipo	
	
<b>Prioridad:</b>	Crítico

Tabla 13. Descripción detallada del caso de Uso del sistema “Visualizar Turnos Reservados”

### 3.5.3.5 Caso de Uso “Gestionar Historias Clínicas”

Nombre del Caso de Uso	Gestionar Historias Clínicas
<b>Actores</b>	Estomatólogo (inicia)
<b>Propósito</b>	Permitir al estomatólogo gestionar historias clínicas de pacientes.
<b>Resumen</b>	Comienza cuando el estomatólogo selecciona la opción de gestionar historias clínicas, mostrándole el sistema un conjunto de acciones que puede realizar, de estas acciones puede seleccionar crear, visualizar o eliminar historias clínicas.
<b>Referencias</b>	R-5
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.
<b>Pos condiciones</b>	Para las siguientes acciones: Crear HC: Se crea una nueva HC en el sistema. Modificar HC: Se modifica el campo de la HC que el estomatólogo haya seleccionado para modificar. Eliminar HC: Se elimina la HC del paciente especificado.
Curso Normal de los Eventos	
Acciones del Actor	Respuesta del Sistema
1. El estomatólogo selecciona la opción Historias	1.1 El sistema muestra una interfaz donde brinda

Clínicas.	la posibilidad de crear, modificar, eliminar o visualizar la HC del paciente.
2. El estomatólogo solicita crear, modificar, eliminar o visualizar la HC de un paciente.	2.1 El sistema ejecuta alguna de las siguientes acciones: a) Si se selecciona la opción de crear HC ir a la sección "Crear HC". b) Si se selecciona la opción de modificar HC ir a la sección "Modificar HC". c) Si se selecciona la opción de visualizar HC ir a la sección "Visualizar HC". d) Si se selecciona la opción de eliminar HC ir a la sección "Eliminar HC".
<b>Sección "Crear HC"</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
3. El estomatólogo entra los datos necesarios para crear la HC.	3.1 El sistema valida los datos insertados y crea la HC almacenando los datos en la base de datos. 3.2 Posteriormente el sistema muestra un mensaje indicando que la HC se ha creado con éxito.
<b>Curso alternativo</b>	
	3.1 El sistema muestra un mensaje indicando llenar los campos obligatorios correctamente.
<b>Prototipo</b>	
<div style="border: 1px solid #ccc; padding: 10px;"> <p><b>Datos Personales</b></p> <p>Fecha: <input type="text" value="2007/06/04"/> <input type="button" value="C"/> <span style="float: right;">Unidad: <input type="text" value="UCI"/></span></p> <p>Paciente: <input type="text" value="Yaily González Fajardo"/> <span style="float: right;">No HC: <input type="text" value="56023"/></span></p> <p>Edad: <input type="text" value="22"/> Sexo: <input type="radio"/> Masc <input checked="" type="radio"/> Fem</p> <p>Motivo: <input type="text" value="Caries"/></p> <p>Historia Enf: <input type="text" value="Presenta caries, lo que ha probocado dolor al paciente."/></p> <p style="text-align: center;"><input type="button" value="Guardar Datos"/></p> </div>	
<b>Sección "Visualizar HC"</b>	

	4.1 El sistema muestra la interfaz que visualiza la historia clínica del paciente con todos sus datos.						
<b>Prototipo</b>							
 <p>The screenshot shows a patient history interface with the following sections:</p> <ul style="list-style-type: none"> <li><b>Datos Personales</b></li> <li><b>Antecedentes Patológicos</b></li> <li><b>Hábitos</b></li> <li><b>Examen Bucal</b></li> <li><b>Examen Físico</b></li> <li><b>Diagnóstico</b></li> <li><b>Tratamiento</b></li> <li><b>Plan Tratamiento:</b> <ul style="list-style-type: none"> <li>-Educación para la salud.</li> <li>-Control de placa dentobacteriana.</li> <li>-Control de hábitos</li> </ul> </li> <li><b>Evolución</b> <table border="1"> <thead> <tr> <th>Fecha</th> <th>Observación</th> <th>Estomatólogo</th> </tr> </thead> <tbody> <tr> <td>2007-06-04</td> <td>Se le dio continuidad al tratamiento anterior, además de prevenir a paciente contra enfermedades bucales.</td> <td>Ramón Torres Mendez</td> </tr> </tbody> </table> </li> </ul>		Fecha	Observación	Estomatólogo	2007-06-04	Se le dio continuidad al tratamiento anterior, además de prevenir a paciente contra enfermedades bucales.	Ramón Torres Mendez
Fecha	Observación	Estomatólogo					
2007-06-04	Se le dio continuidad al tratamiento anterior, además de prevenir a paciente contra enfermedades bucales.	Ramón Torres Mendez					
<b>Sección "Modificar HC"</b>							
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>						
5. El estomatólogo modifica los datos de la HC.	5.1 El sistema valida los datos modificados y actualiza los datos en la base de datos. 5.2 Luego el sistema muestra un mensaje indicando que la HC se ha actualizado.						
<b>Curso alternativo</b>							
	5.1 El sistema muestra un mensaje indicando que los campos obligatorios deben ser llenados correctamente.						
<b>Prototipo</b>							



**Hábitos**

Succión Lengua:  F.Cepillado:

Dir.Cepillado:  T.Cepillado:

Bruxismo:  Tabaquismo:  Si  No

Alcoholismo:  Si  No Café:  Si  No

Otros Hábitos:

<b>Sección "Eliminar HC"</b>							
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>						
	6.1 El sistema muestra la interface para eliminar HC.						
7. El estomatólogo pulsa el botón "Eliminar HC".	7.1 El sistema elimina la HC y muestra un mensaje indicando que la HC ha sido eliminada.						
<b>Curso alternativo</b>							
<b>Prototipo</b>							
<table style="width: 100%; border-collapse: collapse;"> <tr style="background-color: #e1eef6;"> <th style="width: 30%; padding: 5px;">No Historia</th> <th style="width: 40%; padding: 5px;">Nombre y Apellidos</th> <th style="width: 30%; padding: 5px;">Acciones</th> </tr> <tr> <td style="padding: 5px;">55457</td> <td style="padding: 5px;">Ramón Torres Méndez</td> <td style="padding: 5px; text-align: center;"> <input type="button" value="Ver"/> <input type="button" value="Editar"/> <input type="button" value="Eliminar"/> </td> </tr> </table>		No Historia	Nombre y Apellidos	Acciones	55457	Ramón Torres Méndez	<input type="button" value="Ver"/> <input type="button" value="Editar"/> <input type="button" value="Eliminar"/>
No Historia	Nombre y Apellidos	Acciones					
55457	Ramón Torres Méndez	<input type="button" value="Ver"/> <input type="button" value="Editar"/> <input type="button" value="Eliminar"/>					
<b>Prioridad:</b>	Crítico						

Tabla 14. Descripción detallada del caso de Uso del sistema "Gestionar Historias Clínicas"

## 3.5.3.6 Caso de Uso “Gestionar HC - Dentigrama”

<b>Nombre del Caso de Uso</b>	<b>Gestionar Historias Clínicas – Dentigrama.</b>	
<b>Actores</b>	Estomatólogo (inicia)	
<b>Propósito</b>	Permitir al estomatólogo gestionar dentigrama de pacientes.	
<b>Resumen</b>	Comienza cuando el estomatólogo selecciona la opción de gestionar dentigrama, mostrándole el sistema un conjunto de acciones que puede realizar, de estas acciones puede seleccionar crear, modificar o visualizar dentigrama.	
<b>Referencias</b>	R-6	
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para las siguientes acciones: Crear Dentigrama: Se registra un nuevo dentigrama en el sistema. Modificar Dentigrama: Se modifica el dentigrama del paciente seleccionado. Visualizar Dentigrama: Se muestra el dentigrama del paciente especificado.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El estomatólogo selecciona la opción Dentigrama durante la gestión de la HC.	1.1 El sistema muestra una interfaz donde brinda la posibilidad de crear o modificar el dentigrama del paciente.	
2. El estomatólogo solicita crear o modificar el dentigrama de un paciente.	2.1 El sistema ejecuta alguna de las siguientes acciones: a) Si se selecciona la opción de crear evolución ir a la sección “Crear HC - Dentigrama”. b) Si se selecciona la opción de modificar evolución ir a la sección “Modificar HC - Dentigrama”.	
<b>Sección “Crear HC - Dentigrama”</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
3. El estomatólogo entra los datos necesarios para crear el dentigrama.	3.1 El sistema valida los datos insertados y crea el dentigrama almacenando los datos en la base de datos.	

	3.2 Posteriormente el sistema muestra un mensaje indicando que el dentigrama se ha creado con éxito.
<b>Curso alternativo</b>	
	3.1 El sistema muestra un mensaje indicando llenar los campos obligatorios correctamente.
<b>Prototipo</b>	
<div style="border: 1px solid black; padding: 10px;"> <p style="text-align: center;"><b>DENTIGRAMA</b></p> <p style="text-align: center;">Paciente: Isledys Hernández Bermúdez <span style="float: right;">Guardar Dentigrama</span></p> <p style="text-align: center;">SUPERIOR LABIAL</p> <div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> <p>DERECHO</p> <p>18 17 16 15 14 13 12 11</p> <p>55 54 53 52 51 50 49 48 47 46 45 44 43 42 41</p> <p>85 84 83 82 81 80 79 78 77 76 75</p> </div> <div style="width: 10%; text-align: center;"> <p>L</p> </div> <div style="width: 45%;"> <p>21 22 23 24 25 26 27 28</p> <p>61 62 63 64 65</p> <p>71 72 73 74 75</p> <p>IZQUIERDO</p> </div> </div> <div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: 80%;"> <p><b>PIEZA 14:</b></p> <ul style="list-style-type: none"> <li><input type="radio"/> - Sano</li> <li><input checked="" type="radio"/> - Retenido Rx</li> <li><input type="radio"/> - Oligodoncia Rx</li> <li><input type="radio"/> - Corona Funda</li> <li><input type="radio"/> - Corona con Espiga TPR</li> <li><input type="radio"/> - Protesis</li> <li><input checked="" type="checkbox"/> - Ausente</li> <li><input checked="" type="checkbox"/> - Extracción Indicada</li> <li><input type="checkbox"/> - TPR</li> </ul> <p><b>REGION 5:</b></p> <ul style="list-style-type: none"> <li><input type="radio"/> - Sano</li> <li><input type="radio"/> - Caries</li> <li><input type="radio"/> - Obturado</li> <li><input type="radio"/> - Residiva</li> </ul> </div> </div>	
<b>Sección "Modificar HC - Dentigrama"</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
4. El estomatólogo selecciona el dentigrama a modificar.	4.1 El sistema brinda la posibilidad de modificar los datos del dentigrama.
5. El estomatólogo modifica los datos del dentigrama.	5.1 El sistema valida los datos modificados y actualiza los datos en la base de datos. 5.2 Luego el sistema muestra un mensaje indicando que el dentigrama se ha actualizado.
<b>Curso alternativo</b>	
	5.1 El sistema muestra un mensaje indicando que los campos obligatorios deben ser llenados correctamente.
<b>Prototipo</b>	

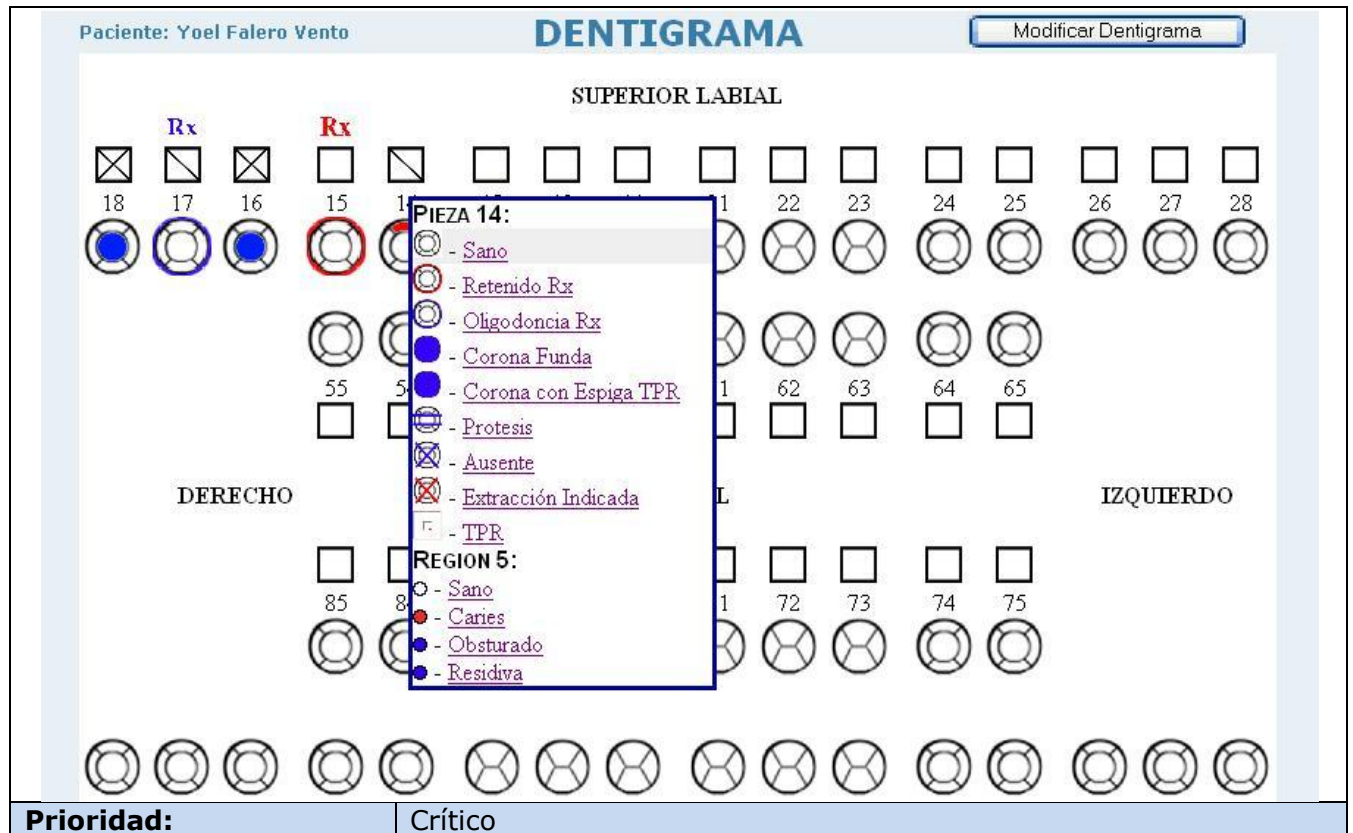


Tabla 15. Descripción detallada del caso de Uso del sistema “Gestionar HC - Dentigrama”

### 3.5.3.7 Caso de Uso “Gestionar HC - Evolución”

Nombre del Caso de Uso	Gestionar Historias Clínicas – Evolución.
Actores	Estomatólogo (inicia)
Propósito	Permitir al estomatólogo gestionar evolución de un paciente.
Resumen	Comienza cuando el estomatólogo selecciona un paciente para gestionar la evolución del mismo, mostrándole el sistema un conjunto de acciones que puede realizar, de estas acciones puede seleccionar crear, visualizar o modificar evolución.
Referencias	R-7
Precondiciones	El estomatólogo debe haberse autenticado en el sistema.

<b>Pos condiciones</b>	<p>Para las siguientes acciones:</p> <p>Crear Evolución: Se registra una nueva evolución.</p> <p>Visualizar Evolución: Se muestra la evolución del paciente especificado.</p> <p>Modificar Evolución: Se modifica la evolución del paciente seleccionado.</p>
------------------------	---

**Curso Normal de los Eventos**

Acciones del Actor	Respuesta del Sistema
1. El estomatólogo durante la gestión de la HC accede a la opción de evolución.	1.1 El sistema muestra una interfaz donde brinda la posibilidad de visualizar, crear o modificar la evolución del paciente.
2. El estomatólogo solicita visualizar, crear o modificar la evolución de un paciente.	2.1 El sistema ejecuta alguna de las siguientes acciones: a) Si se selecciona la opción de crear evolución ir a la sección "Crear HC - Evolución". b) Si se selecciona la opción de modificar evolución ir a la sección "Modificar HC - Evolución". c) Si se selecciona la opción de visualizar evolución ir a la sección "Visualizar HC - Evolución".

**Sección "Crear HC - Evolución"**

Acciones del Actor	Respuesta del Sistema
3. El estomatólogo entra los datos necesarios para crear la evolución.	3.1 El sistema valida los datos insertados y crea la evolución almacenando los datos en la base de datos. 3.2 Posteriormente el sistema muestra un mensaje indicando que la evolución se ha creado con éxito.

**Curso alternativo**

	3.1 El sistema muestra un mensaje indicando llenar los campos obligatorios correctamente.
--	---

**Prototipo**

The screenshot shows a web interface for creating a patient evolution. It features the following elements:

- Fecha:** A date input field containing "2007/06/11" and a calendar icon.
- Estomatólogo:** A text input field containing "Yunier Albrecht Delgado".
- Evolución:** A text area containing "Se realizó una extracción al paciente" with scroll arrows on the right side.
- Guardar Evolucion:** A button at the bottom center of the form.

<b>Sección "Modificar HC - Evolución"</b>										
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>									
4. El estomatólogo selecciona la evolución a modificar.	4.1 El sistema brinda la posibilidad de modificar los datos.									
5. El estomatólogo modifica los datos de la evolución seleccionada.	5.1 El sistema valida los datos modificados y actualiza los datos en la base de datos. 5.2 Luego el sistema muestra un mensaje indicando que la evolución ha sido actualizada.									
<b>Curso alternativo</b>										
	5.1 El sistema muestra un mensaje indicando que los campos obligatorios deben ser llenados correctamente.									
<b>Prototipo</b>										
<b>Sección "Visualizar HC - Evolución"</b>										
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>									
6. El estomatólogo selecciona mostrar evolución.	6.1 Si la evolución del paciente existe el sistema la muestra con todos sus datos.									
<b>Curso alternativo</b>										
	6.1 Si no existe la evolución del paciente el sistema muestra un mensaje informando que no existe.									
<b>Prototipo</b>										
<table border="1"> <thead> <tr> <th colspan="3"><b>Evolución</b></th> </tr> <tr> <th><b>Fecha</b></th> <th><b>Observación</b></th> <th><b>Estomatólogo</b></th> </tr> </thead> <tbody> <tr> <td>2007-06-11</td> <td>Se realizó una extracción al paciente</td> <td>Yunier Albrecht Delgado</td> </tr> </tbody> </table>		<b>Evolución</b>			<b>Fecha</b>	<b>Observación</b>	<b>Estomatólogo</b>	2007-06-11	Se realizó una extracción al paciente	Yunier Albrecht Delgado
<b>Evolución</b>										
<b>Fecha</b>	<b>Observación</b>	<b>Estomatólogo</b>								
2007-06-11	Se realizó una extracción al paciente	Yunier Albrecht Delgado								
<b>Prioridad:</b>	Crítico									

Tabla 16. Descripción detallada del caso de Uso del sistema "Gestionar HC – Evolución"

### 3.5.3.8 Caso de Uso "Generar Parte Diario"

<b>Nombre del Caso de Uso</b>	<b>Generar Parte Diario.</b>
<b>Actores</b>	Estomatólogo (inicia).
<b>Propósito</b>	Permitir al estomatólogo confeccionar el parte diario.
<b>Resumen</b>	Inicia cuando el estomatólogo selecciona la opción para crear el parte diario de su trabajo individual.

<b>Referencias</b>	R-11
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.
<b>Pos condiciones</b>	Para las siguiente acción: Generar Parte: Se crea un nuevo parte referente al personal médico.
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1. El estomatólogo solicita la opción Parte Diario.	1.1 El sistema muestra una interfaz para entrar los datos necesarios.
2. El estomatólogo entra los datos para crear el parte diario.	2.1 El sistema valida los datos insertados y los archiva en la base de datos. 2.2 Posteriormente el sistema muestra un mensaje indicando que los datos fueron insertados correctamente.
<b>Curso alternativo</b>	
	2.1 Emite un mensaje indicando que debe llenar los campos obligatorios.
<b>Prototipo</b>	
<b>Prioridad:</b>	Crítico

Tabla 17. Descripción detallada del caso de Uso del sistema “Generar Parte Diario”

### **3.6 Conclusiones.**

En este capítulo se describió el Negocio para la comprensión de cómo ocurren los procesos en la realidad dentro de la instalación estudiada, así también se desglosó el sistema en casos de usos del sistema, atendiendo al criterio de empaquetamiento de Casos de Usos del sistema también detallado en este capítulo, para mejorar el entendimiento de cómo funciona el sistema dando respuesta al problema que trata este trabajo.



## CAPÍTULO 4

# Construcción de la Solución Propuesta

### 4.1 Introducción

En el presente capítulo se muestran los resultados de la elaboración de los diagramas de la etapa de Diseño, como los diagramas de Clases del Diseño Web, Modelo de Datos, que forman la base para entrar en la fase de Implementación orientada en la metodología de RUP. También están las pautas de Diseño e Interfaz de usuario utilizadas para el desarrollo de la implementación del sistema así como generalidades que ayudarán al entendimiento de dicha implementación, así como la distribución física del sistema en su instalación y el plan de pruebas y casos de prueba diseñados para probar la aplicación.

### 4.2 Diagramas de Clases

Un diagrama de clases presenta las clases del sistema con sus relaciones estructurales y de herencia, todo el código que irá creando las páginas, así como el contenido dinámico de estas una vez que estén en el navegador del cliente. En el caso de las aplicaciones Web, el diagrama de clases representa las colaboraciones que ocurren entre las páginas, donde cada página lógica puede ser representada como una clase, es muy importante pues estos son los artefactos que se necesitan modelar para que el desarrollador los implemente y obtener así el producto final con la calidad requerida.

El diagrama de clases Web, fue definido, a partir de los diferentes casos de uso del sistema y empleando las extensiones de UML para Web, a continuación se muestran los diagramas de clases.

### 4.2.1 Diagrama de Clases Web del CUS “Autenticar Usuario”

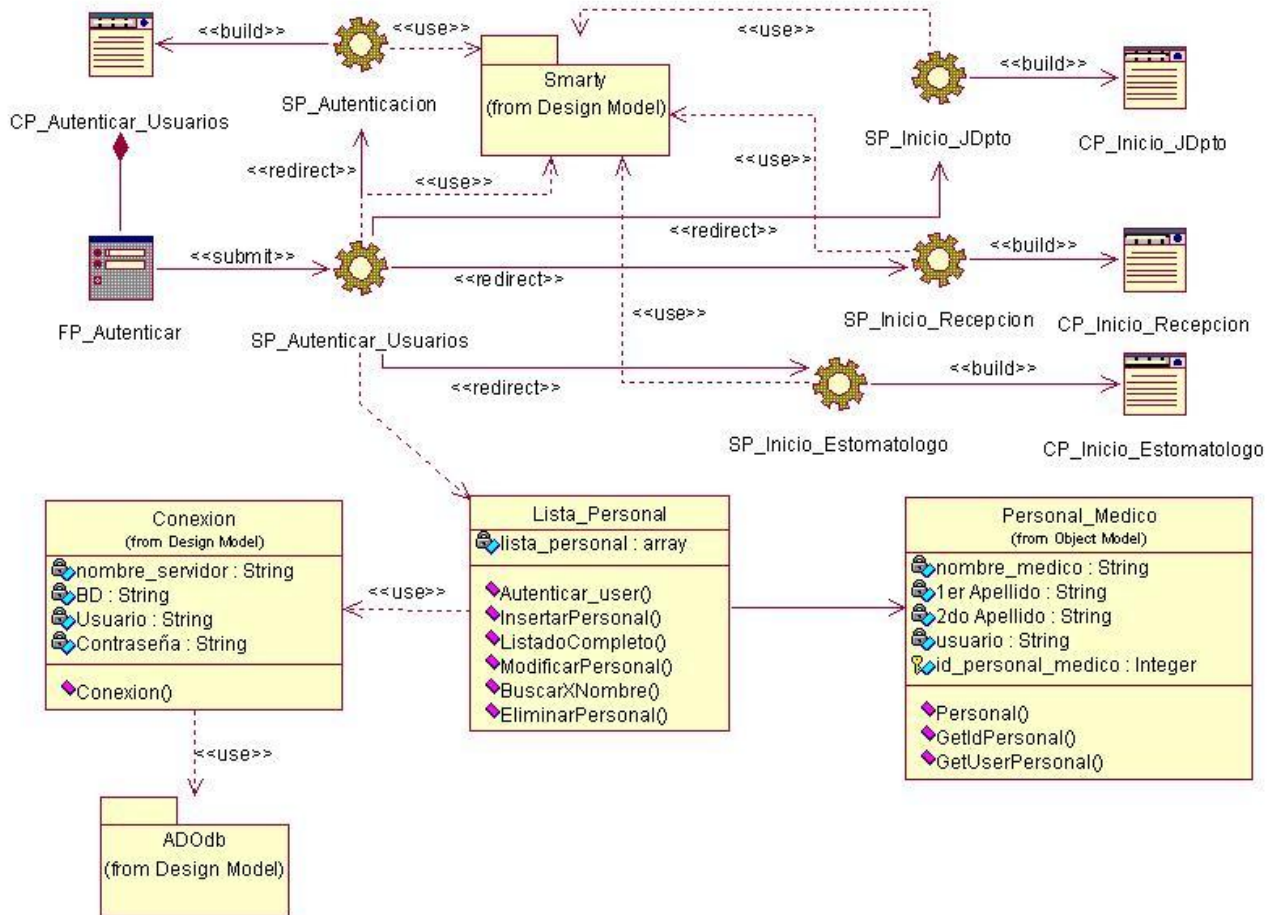


Figura 14. Diagrama de Clases del Diseño del CUS Autenticar Usuario

### 4.2.2 Diagrama de Clases Web del CUS “Gestionar Usuario”

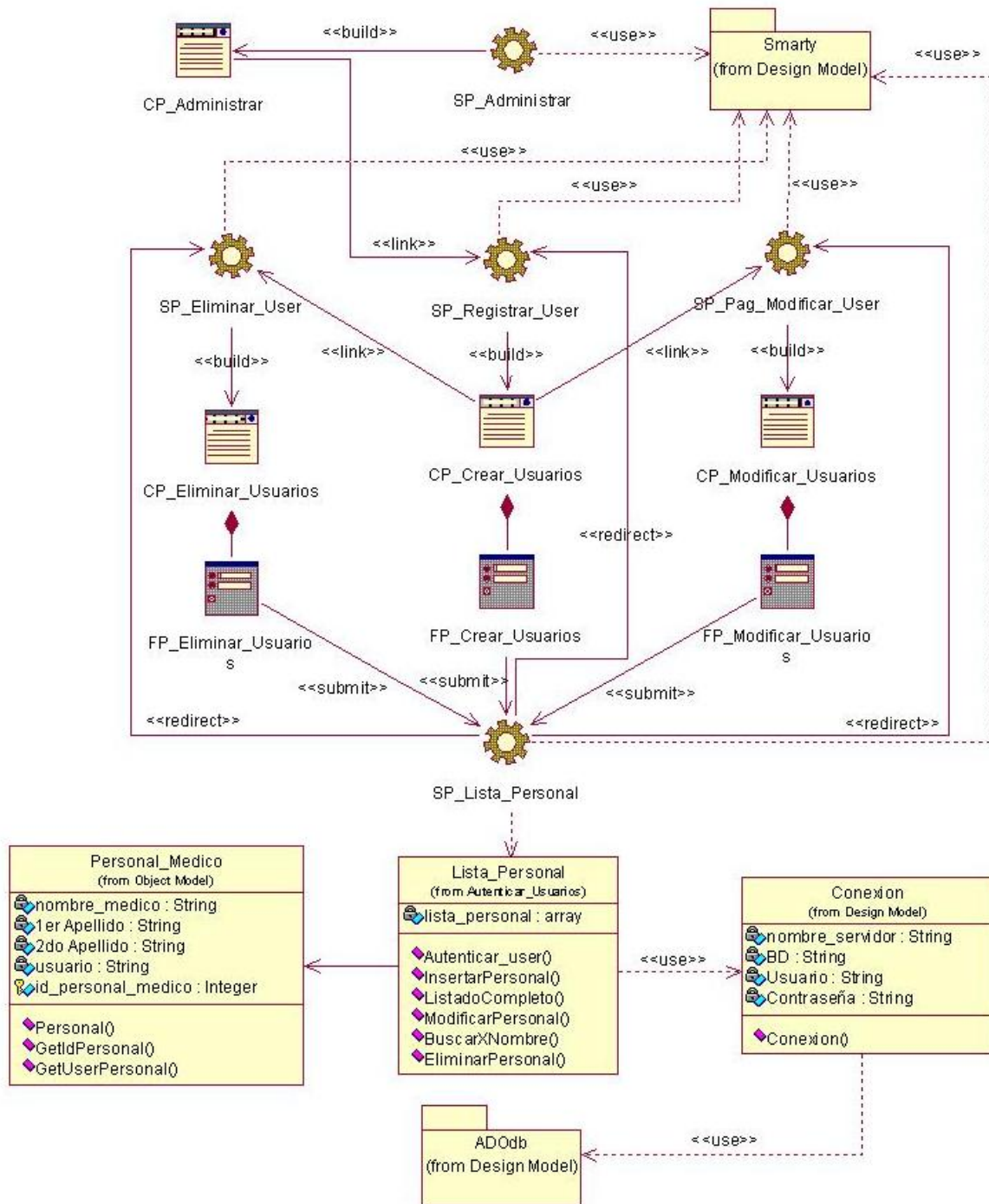


Figura 15. Diagrama de Clases del Diseño del CUS Gestionar Usuarios

### 4.2.3 Diagrama de Clases Web del CUS “Gestionar Reservación”

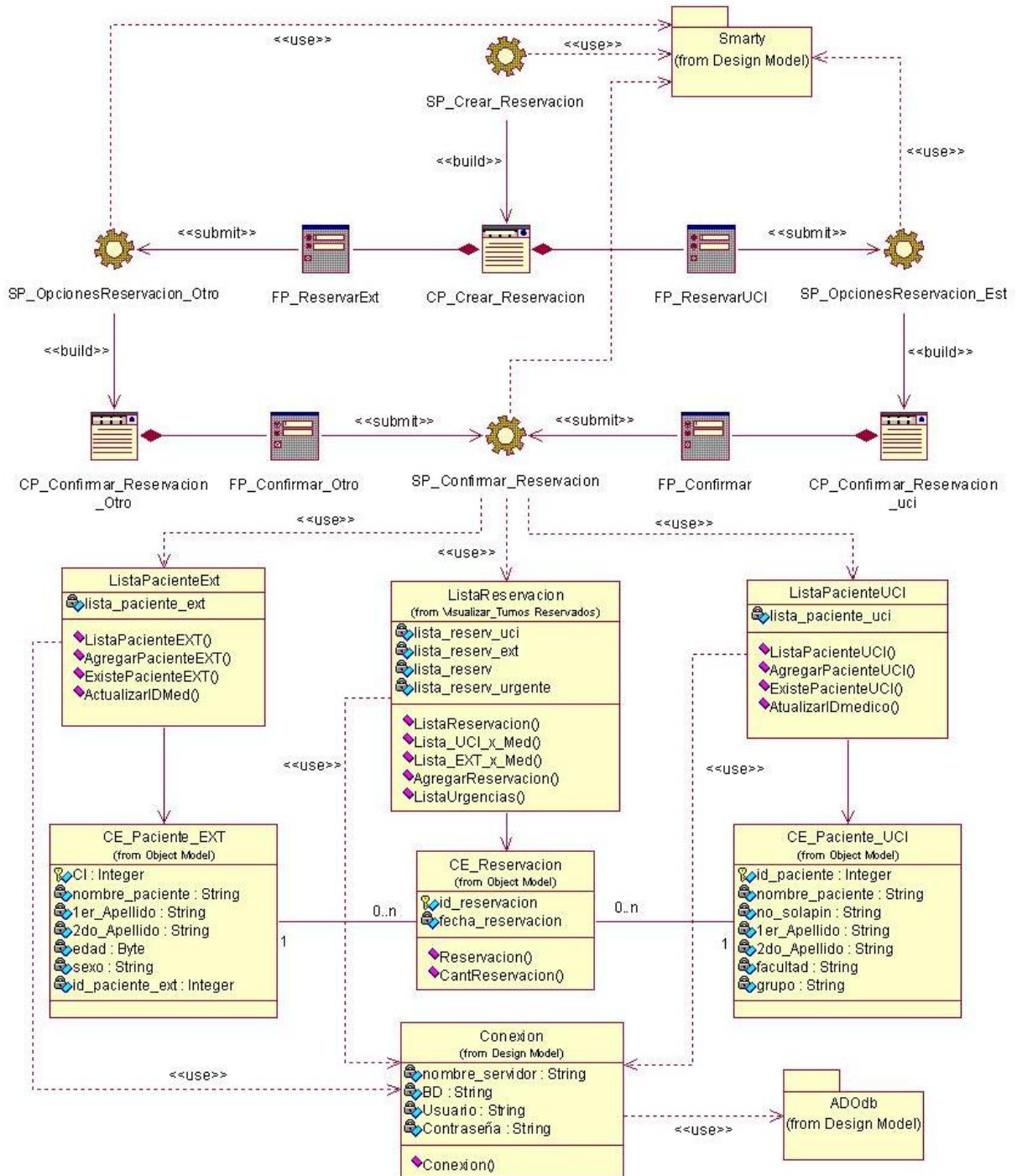


Figura 16. Diagrama de Clases del Diseño del CUS Gestionar Reservación

### 4.2.4 Diagrama de Clases Web del CUS “Gestionar HC”

#### 4.2.4.1 Diagrama de Clases Web del CUS Gestionar HC. “Sección Crear HC”

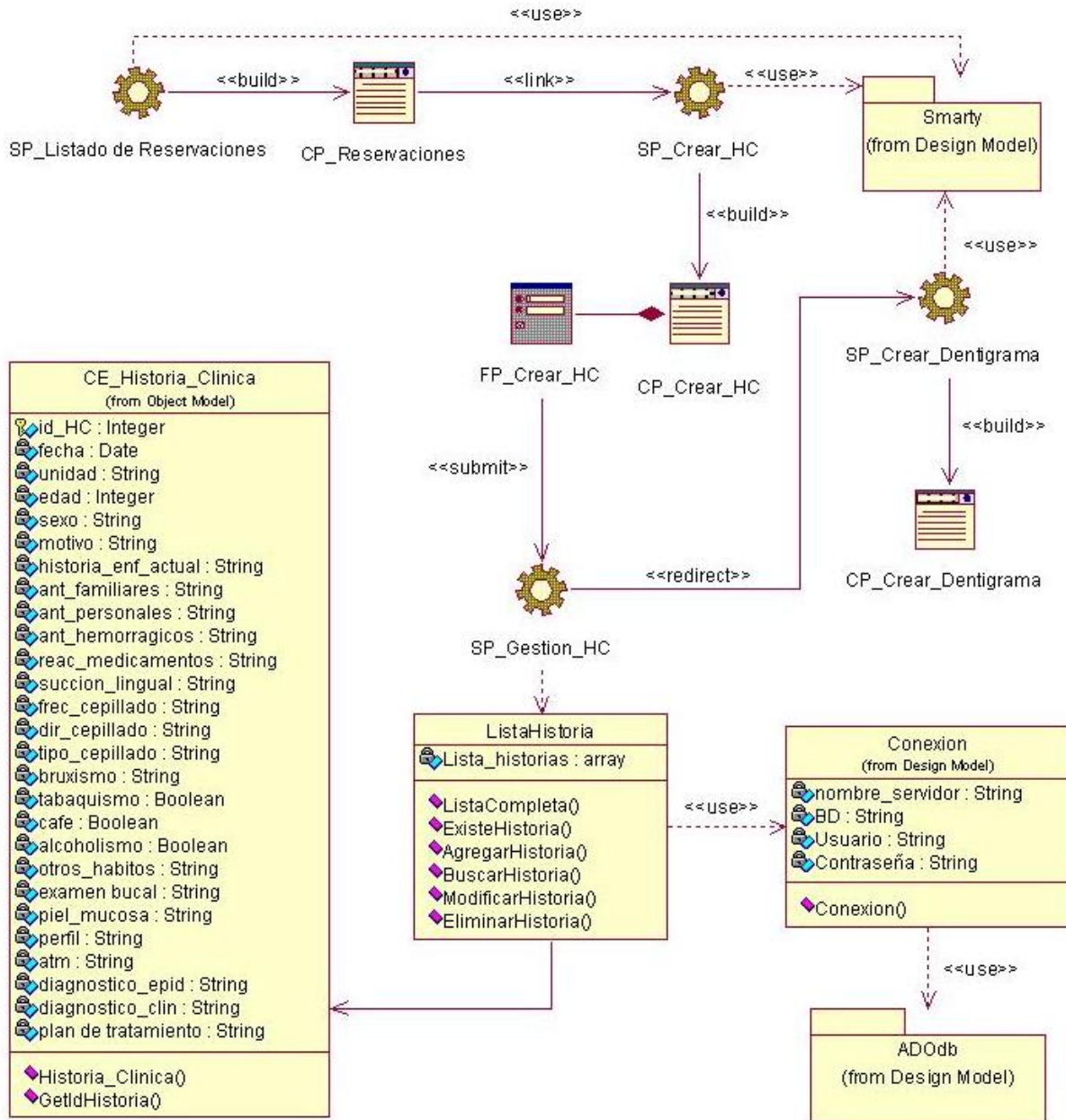


Figura 17. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Crear HC

4.2.4.2 Diagrama de Clases Web del CUS Gestionar HC. “Sección Modificar HC”

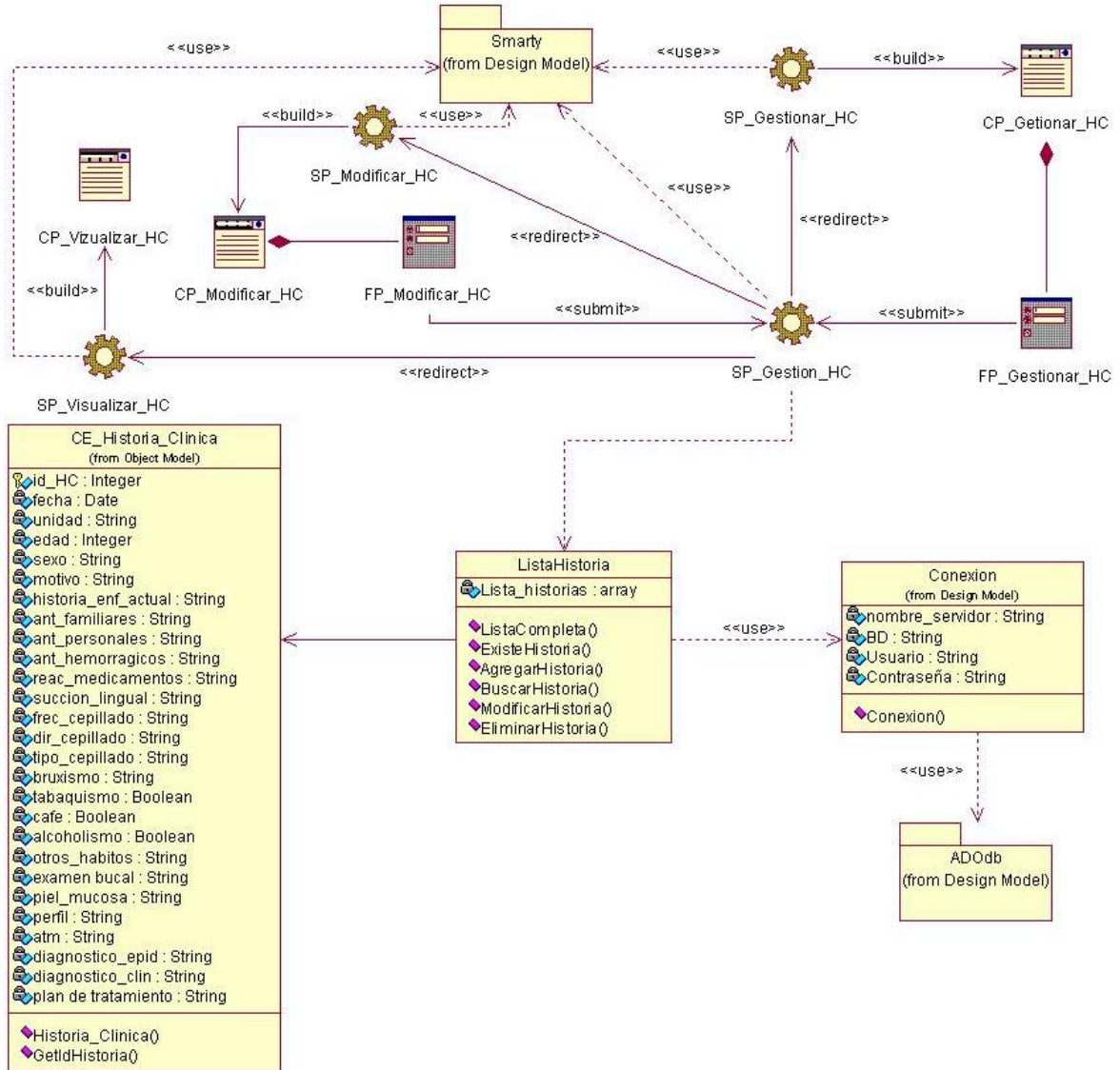


Figura 18. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Modificar HC

4.2.4.3 Diagrama de Clases Web del CUS Gestionar HC. “Sección Eliminar HC”

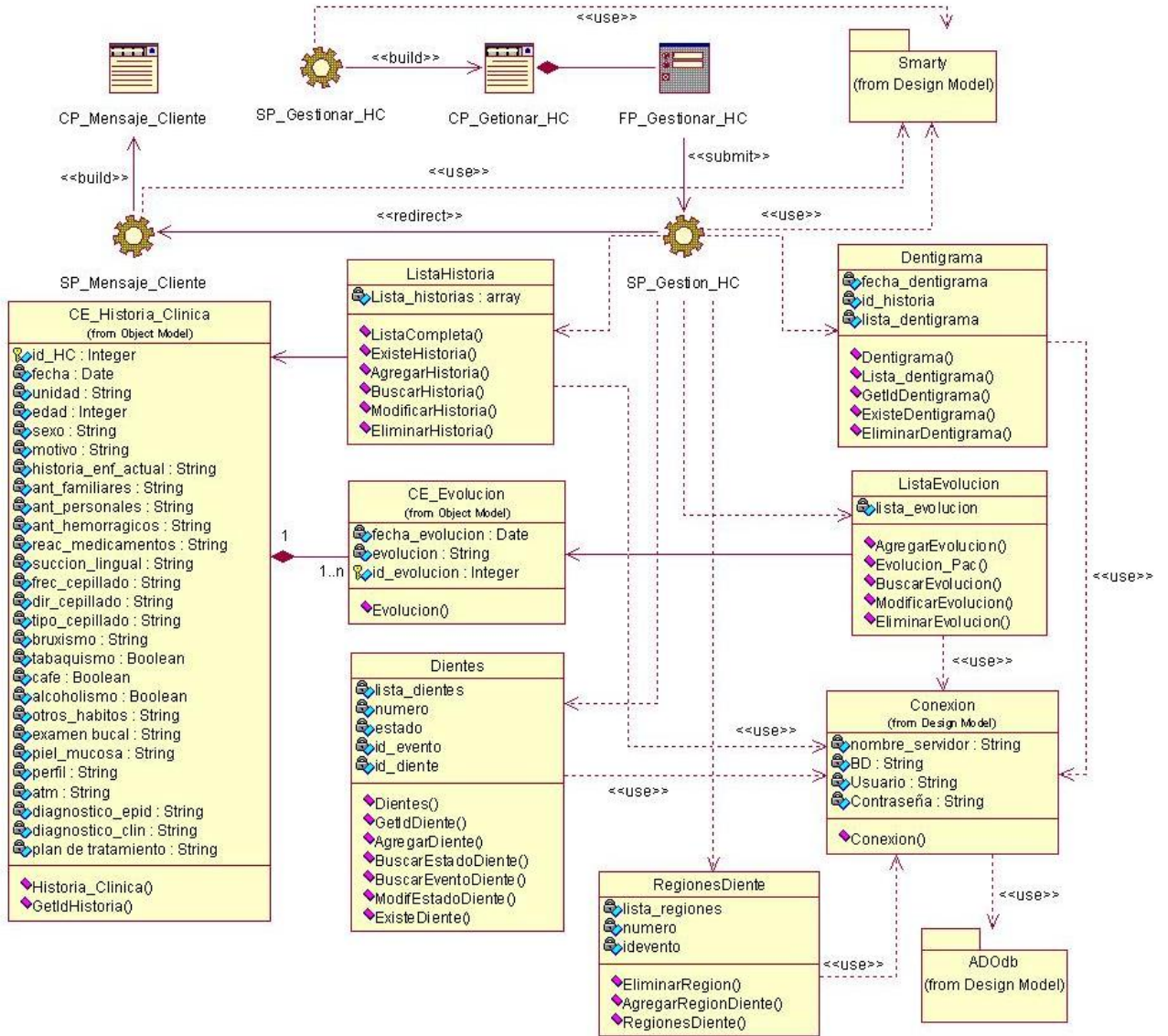


Figura 19. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Eliminar HC

4.2.4.4 Diagrama de Clases Web del CUS Gestionar HC. “Sección Visualizar HC”

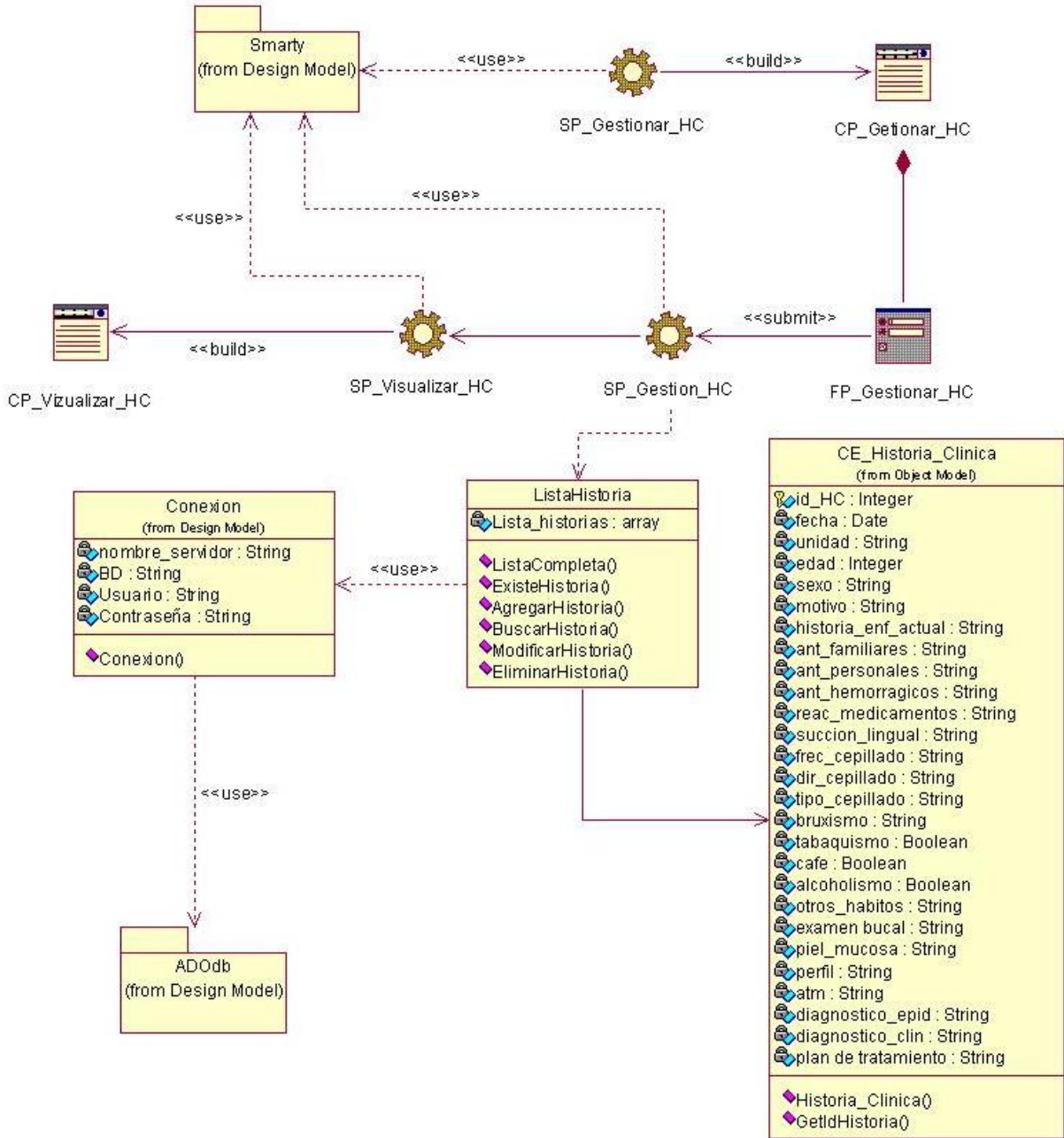


Figura 20. Diagrama de Clases del Diseño del CUS Gestionar HC. Flujo Visualizar HC



### 4.2.5 Diagrama de Clases Web del CUS “Gestionar Dentigrama”

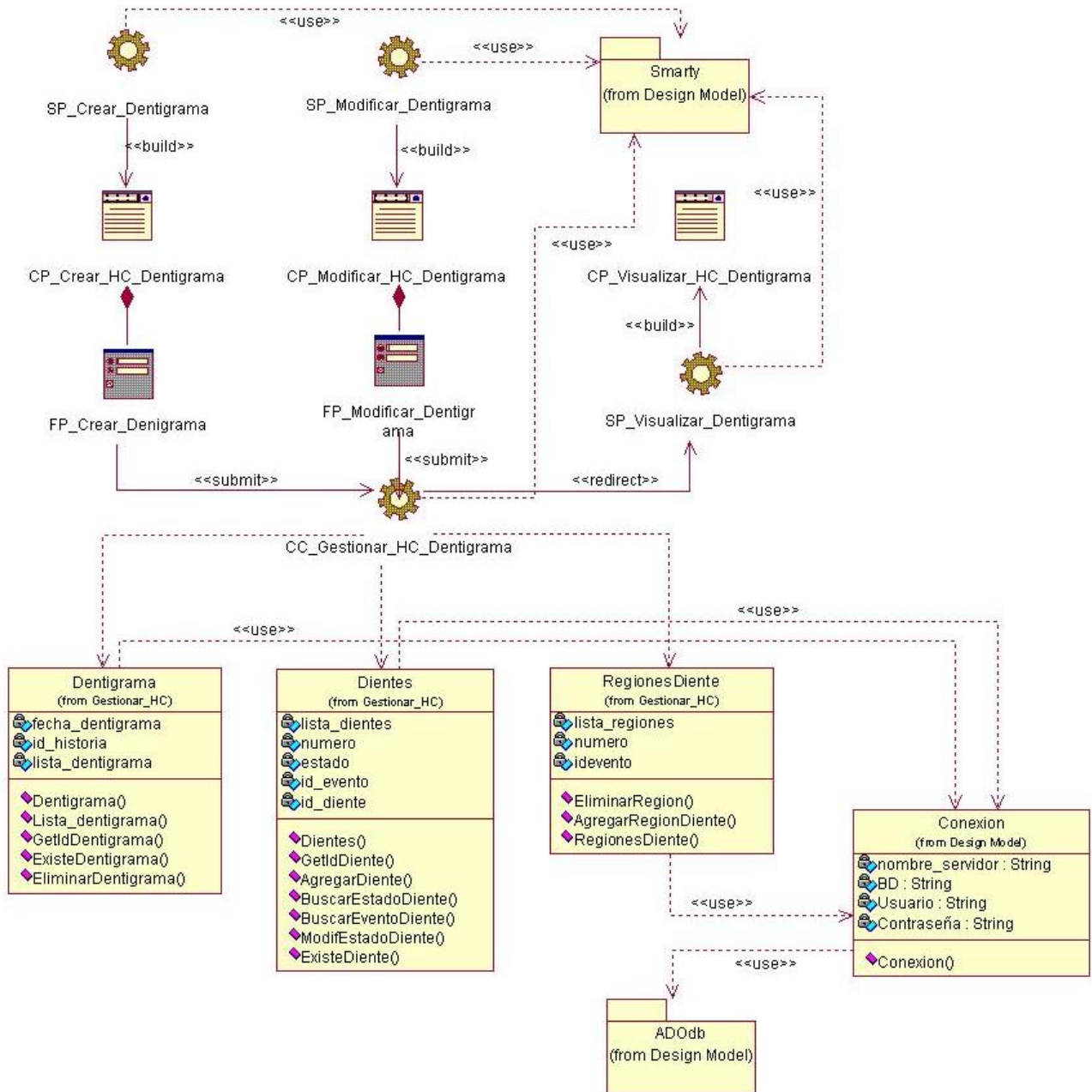


Figura 21. Diagrama de Clases del Diseño del CUS Gestionar Dentigrama

### 4.2.6 Diagrama de Clases Web del CUS “Gestionar Evolución”

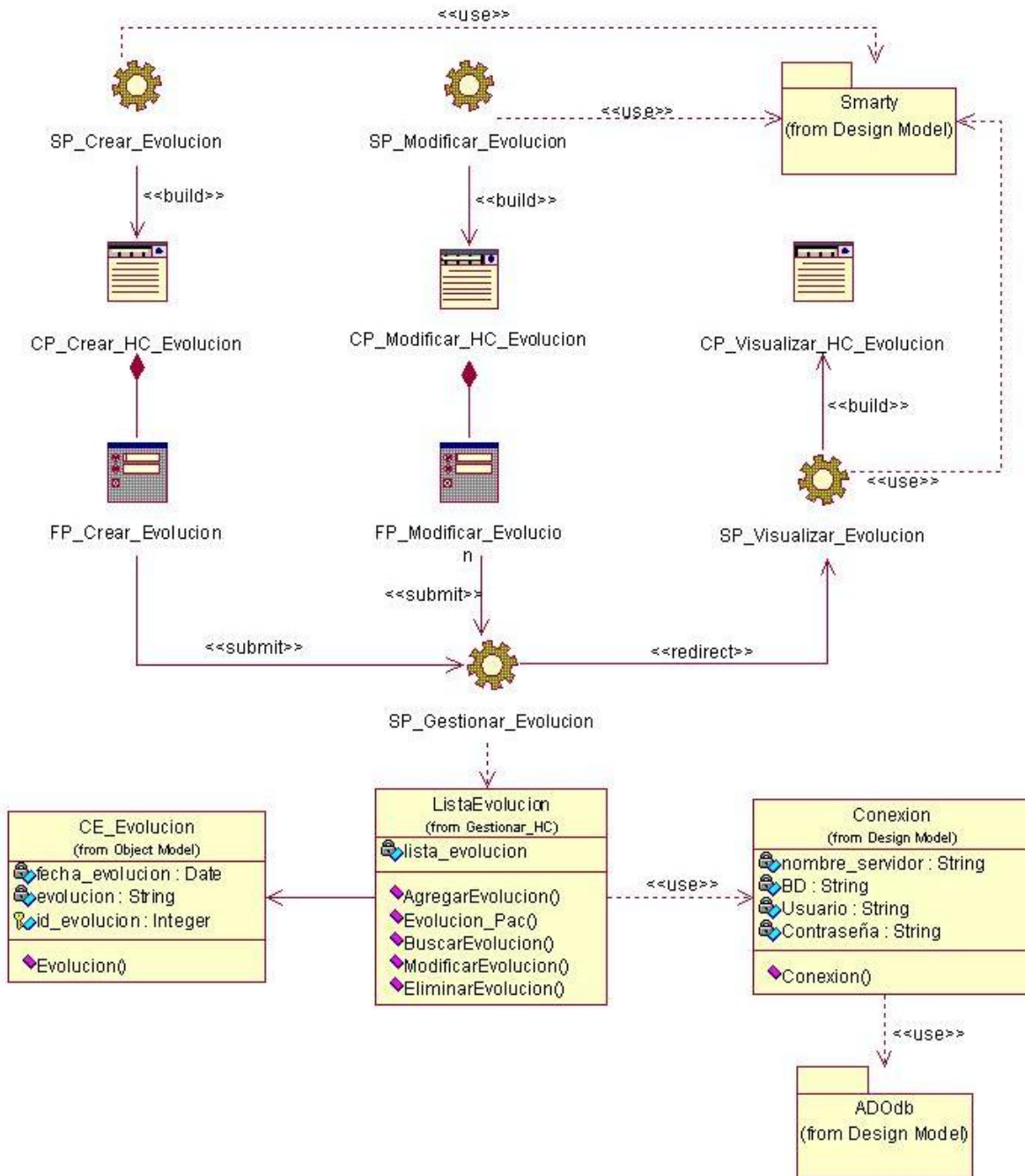


Figura 22. Diagrama de Clases del Diseño del CUS Gestionar Evolución

### 4.2.7 Diagrama de Clases Web del CUS “Visualizar Turnos Reservados”

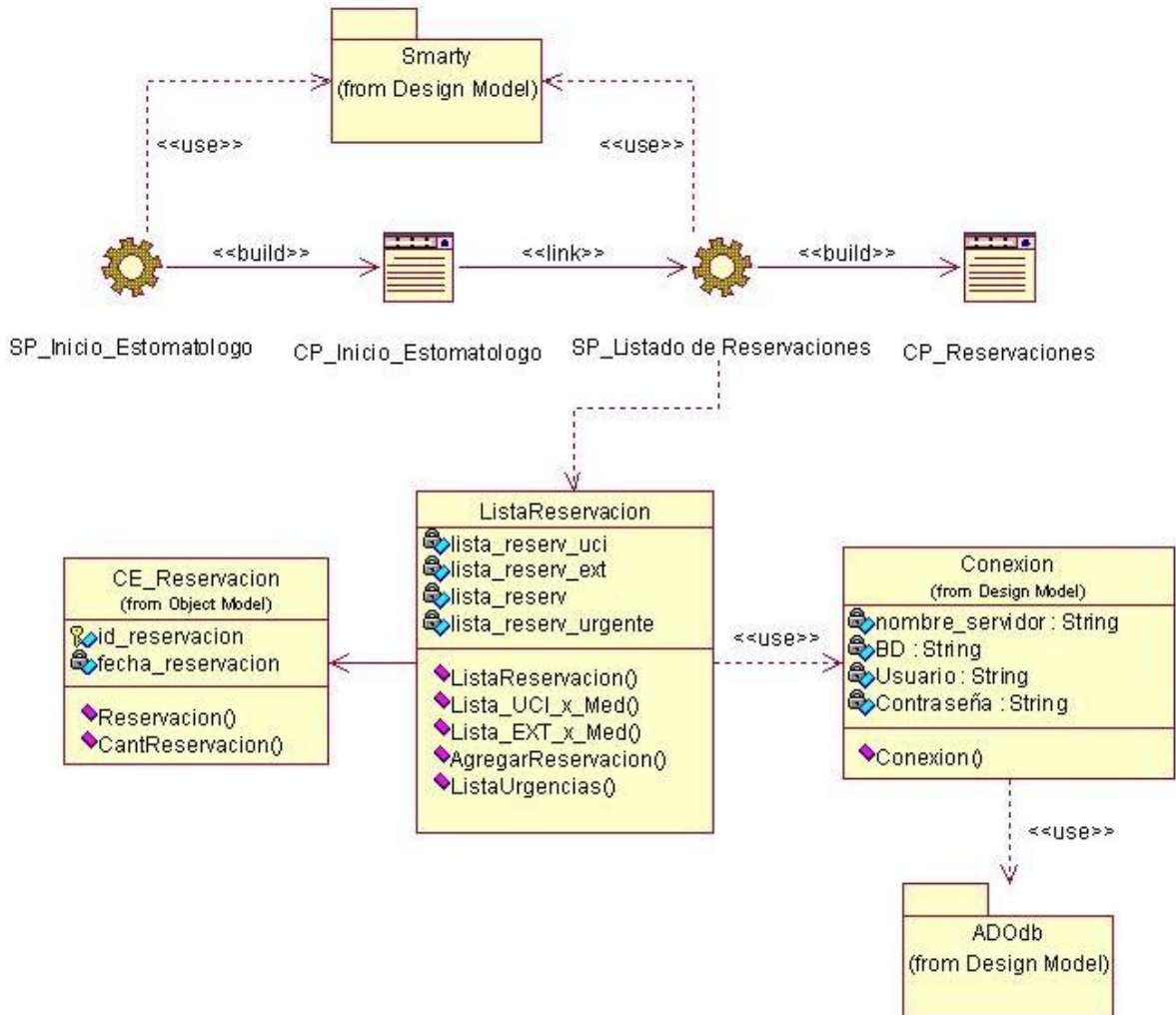


Figura 23. Diagrama de Clases del Diseño del CUS Visualizar Turnos Reservados

### 4.2.8 Diagrama de Clases Web del CUS “Generar Parte Diario”

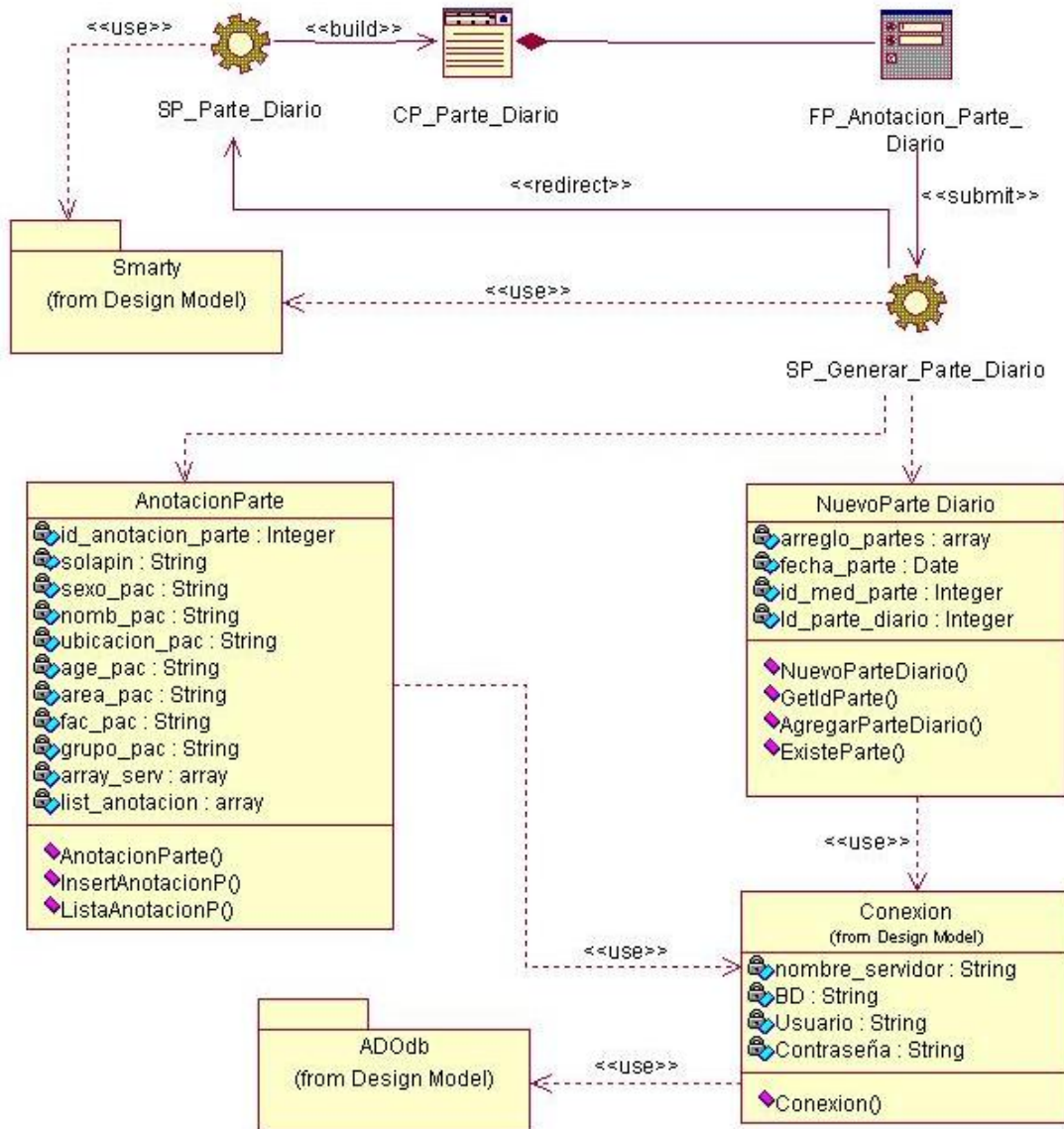


Figura 24. Diagrama de Clases del Diseño del CUS Generar Parte Diario

### 4.2.9 Diagrama de Clases Web del CUS “Generar Parte Gasto Material”

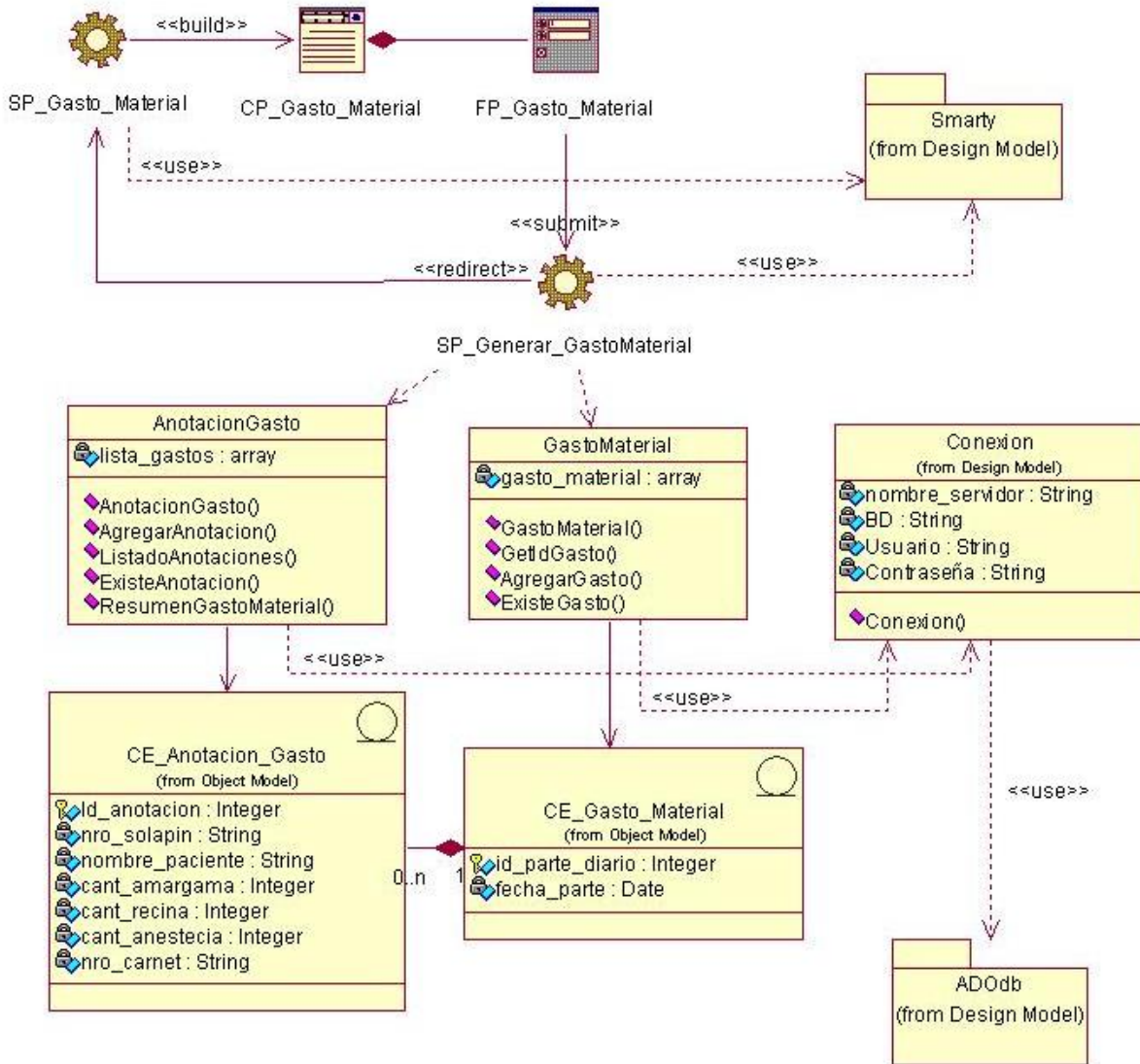


Figura 25. Diagrama de Clases del Diseño del CUS Generar Parte Gasto Material

### 4.3 Principios de diseño.

La calidad de la interfaz de usuario puede ser uno de los motivos que conduzca a un sistema al éxito o al fracaso, es por eso que uno de los aspectos más relevantes de la usabilidad de un sistema es la consistencia de su interfaz de usuario.

Para el desarrollo de la interfaz se tuvo en cuenta los siguientes aspectos:

- 1) Reducir la carga a la memoria.
- 2) Obtener información de retroalimentación.
- 3) Diseño de diálogos que conducen a una conclusión.
- 4) Previsión de errores y manejo de errores simples.
- 5) Que el usuario sintiera la sensación de control.

Una de las premisas fundamentales de la aplicación es la ventaja que proporciona las interfaces Web sobre las interfaces de comando. Ya que las interfaces Web:

- Proporcionan un ambiente amigable.
- Conducen a un aprendizaje más natural.
- Establecen un “sentimiento” (sobre todo en la uniformidad del ambiente) al usuario que enriquece su experiencia en el uso de la aplicación.

Además de estos principios, se tuvieron en cuenta las siguientes características:

- Utilizar una misma tipografía, forma y estilo en todas las páginas.
- La facilidad del usuario de poder navegar desde cualquier punto a otro dentro de la aplicación.
- Se tuvo presente siempre el ancho de banda y por ello se utilizaron formato de imágenes de compresión favorables.

- La simplicidad y consistencia, favoreciendo la usabilidad de la aplicación.
- Navegación simple en todas las páginas de la aplicación, de forma tal que siempre sea accesible por el usuario.
- Estabilidad y uniformidad del diseño, para así poder ubicar al usuario dentro del mismo y hacerlo sentir parte de él.

Se realizan múltiples operaciones en cada página, de forma que el usuario no tenga que moverse tanto dentro de la aplicación, para completar una operación.

### **4.3.1 Estándares de la interfaz de la aplicación**

Para el diseño de la interfaz de usuario se utilizan varias plantillas basadas en los roles de los usuarios del sistema y una hoja de estilos para guardar la configuración del diseño para todas las páginas. Para los botones y las líneas se utilizaron estos estilos, eliminando así el número de imágenes que demoren la presentación de la página. Esta hoja de estilos establece el tipo y tamaño de fuente de cada elemento de la página predominando la familia de fuentes Verdana, Arial, Helvetica, sans-serif de tamaño entre 10 y 12 pixeles para el texto en general.

La plantilla está estructurada de la siguiente manera: en la parte superior se encuentra el header de la página conteniendo en su parte izquierda el logo de la institución cliente y en el centro el nombre del Sistema, además de vínculos a paginas de uso común para todos los usuarios. En la parte central izquierda están colocados los menús de navegabilidad del usuario que varían en dependencia del rol que desempeñen y contienen las operaciones a realizar y el centro de la plantilla estará conformado por todos los formularios. En la parte inferior está ubicado el footer que va a contener la identidad de la empresa para la que ha sido desarrollado el sistema.

A continuación se muestra la plantilla en general:

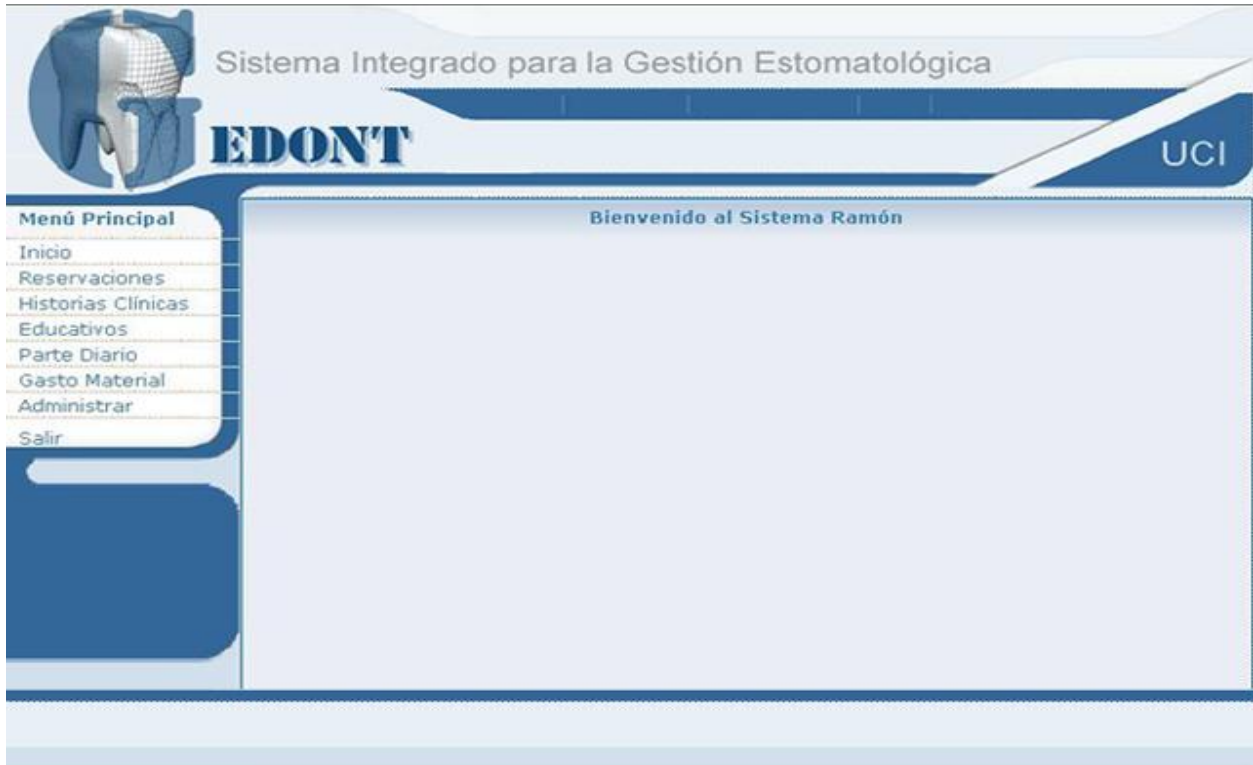


Figura 26. Plantilla del Sistema en General

### 4.3.2 Estándares de Codificación

Usar técnicas de codificación sólidas y realizar buenas prácticas de programación con vistas a generar un código de alta calidad es de gran importancia para la calidad del software y para obtener un buen rendimiento. Además, si se aplica de forma continuada un estándar de codificación bien definido, se utilizan técnicas de programación apropiadas, y, posteriormente, se efectúan revisiones del código de rutinas, esto trae consigo que la aplicación se convierta en un sistema de software fácil de comprender y de mantener. (24)



Con el objetivo de facilitar el mantenimiento del software se emplearon los siguientes estándares de codificación:

- Empleo de comentarios en todas las declaraciones de clases y funciones más complejas.
- Utilización de nombres de clases significativos, que expresen total o parcialmente su significado.
- Utilización del carácter “\_”, para los nombres de las clases y las variables, en substitución del carácter espacio para hacer el código más legible.
- Los nombres de las clases se escribieron con mayúsculas.
- Organización del código de forma estructurada, en bloques de código, para una mejor lectura del mismo.

### **4.3.3 Tratamiento de Errores**

En el sistema propuesto se evitan, minimizan y tratan los posibles errores, con el fin de garantizar la integridad y confiabilidad de la información que en este se registra y muestra. Los errores se tratan en una página especial que incluye el fichero de configuración general, y está preparada para recoger el número del error y presentar la pantalla con el error que le corresponde a ese código.

Los mensajes de error que emite el sistema se muestran en un lenguaje de fácil comprensión para los usuarios. Cuando se introduce información en un formulario y faltan datos, sale un cuadro de alerta indicando el campo o dato que falta.

## **4.4 Diseño de la Base de Datos**

La base de datos es el sistema utilizado para el almacenamiento de datos y acceso controlado a los datos almacenados. Es la colección de datos organizada para dar servicio a muchas aplicaciones al mismo tiempo al combinar los datos de manera que aparezcan estar en una sola ubicación. En este epígrafe se muestra el diseño de la base de datos del sistema propuesto a través del diagrama de clases persistentes y el esquema de la base de datos generados a partir de este.

### 4.4.1 Diagrama de Clases Persistentes

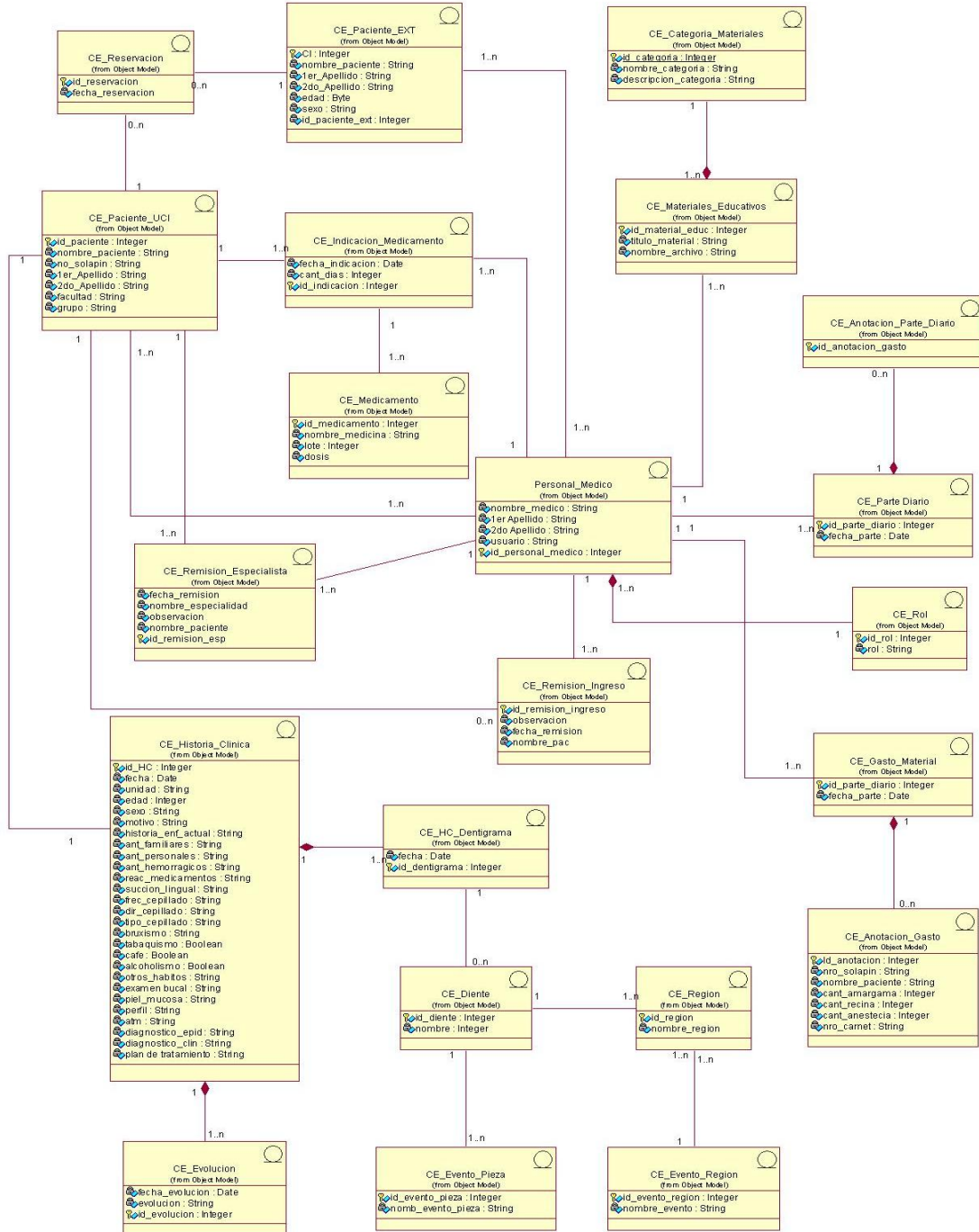


Figura 27. Diagrama de Clases Persistentes

### 4.4.2 Modelo de Datos

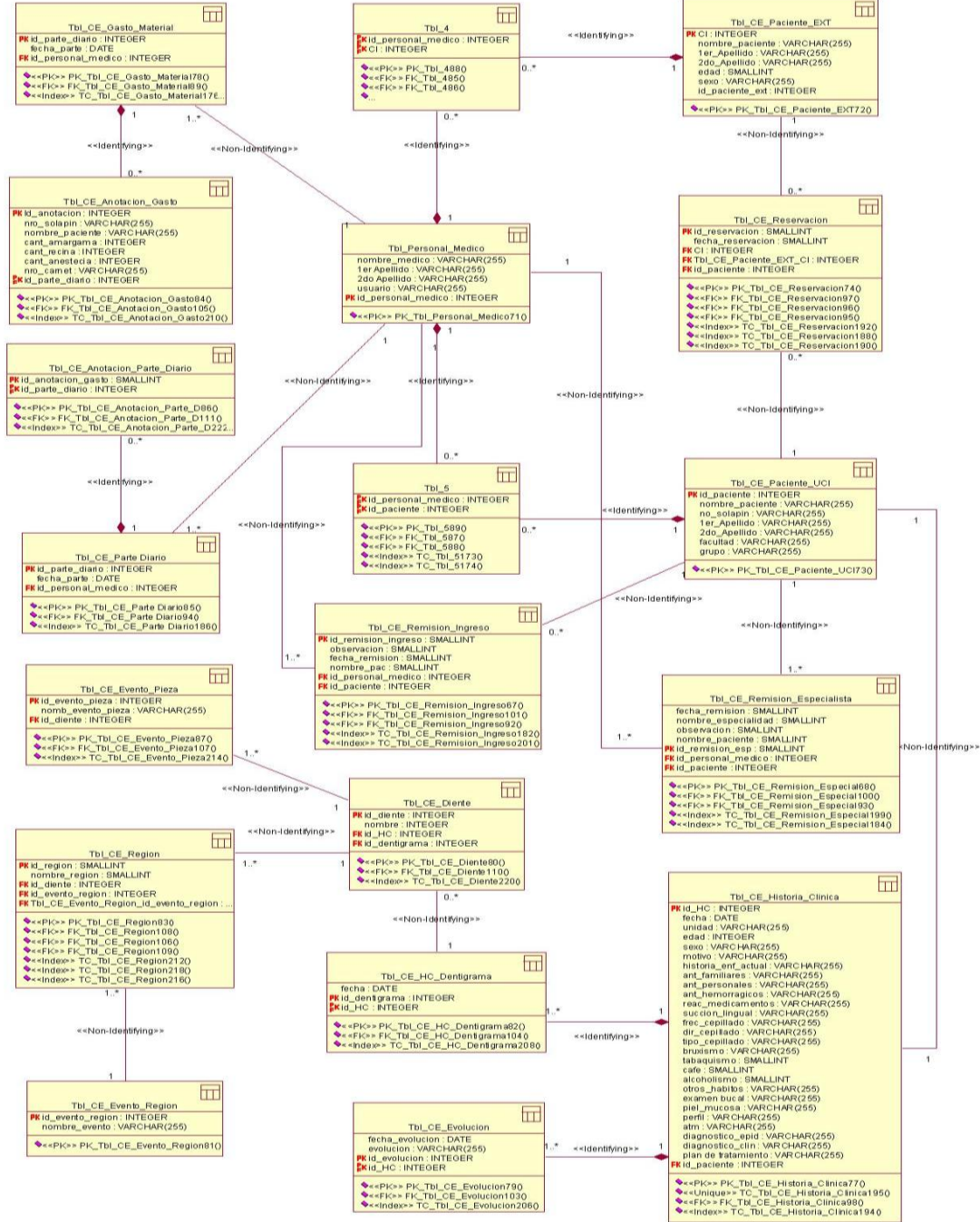


Figura 28. Diagrama de Modelo de Datos

## 4.5 Generalidades de la Implementación

Para el desarrollo del sistema propuesto se utilizó la librería de acceso a datos adodb que permite implementar fácilmente una capa de abstracción de base de datos, para realizar la página una única vez y que permita utilizar cualquier sistema gestor de base de datos sin cambiar el código, siendo compatible con una multitud de Bases de Datos como son: MySQL, Oracle, Microsoft SQL Server, Sybase, Sybase SQL Anywhere, Informix, PostgreSQL, FrontBase, SQLite, Interbase, Foxpro, Access, ADO, DB2, SAP DB y ODBC). Además el uso de estas librerías o clases conlleva una mejora en nuestro modo de trabajo, siendo más rápidos y organizados. (26)

La aplicación fue concebida utilizando el motor de plantillas Smarty ya que con el podemos realizar aplicaciones web de calidad separando el código PHP de la presentación HTML, además brinda un número de funcionalidades de las cuales podemos disfrutar, estas son: (27)

- Plantillas limpias fáciles de usar por los diseñadores.
- Escalabilidad.
- Mantenimiento más sencillo (al igual que la escalabilidad, únicamente con la separación de código y presentación no se consigue un mantenimiento más sencillo, también se requerirá de una buena codificación).
- Depuración óptima del código, al tener ficheros pequeños únicamente con código PHP.
- Posibilidad de introducir comentarios dentro de las plantillas que no se enviarán al servidor.
- Funciones integradas que facilitan el tratamiento de variables.
- Funciones asistentes para generación de código HTML.

La arquitectura usada para el desarrollo de la aplicación está basada en el modelo de tres capas: la primera de las capas está dirigida a los usuarios, la segunda de las capas está destinada a la aplicación web que se encuentra instalada y procesada en el servidor y a ella accede cada uno de los usuarios (tecnología cliente-servidor) y la última de las capas es la encargada de suministrar los datos, la base de datos.

La solución técnica de la aplicación está basada en la integración de una serie de tecnologías web como son:

**HTML:** Lenguaje necesario para conseguir los estándares Web que aparte de los beneficios que estos aportan son los que abren las puertas al futuro del Web, de una forma inteligente, regulada y en consenso.

**CSS:** Con él se define de una forma casi ilimitada el estilo y color del sitio Web ayudando en el proceso de estandarización y optimización del sitio.

**PHP:** Para llevar a la realidad el dinamismo del sitio mediante la gestión de contenidos, todo ello a través de herramientas programadas para poder interactuar con dicha información.

**MySQL:** Se usa como gestor de base de datos, destacable por su robustez y velocidad.

**Apache:** Es la tecnología empleada como servidor Web.

### **4.6 Modelo de Despliegue**

El diagrama de despliegue es un modelo de objetos que describe la distribución física del sistema en términos de cómo se distribuye la funcionalidad entre los nodos de cómputo. Es una colección de nodos y arcos; donde cada nodo representa un recurso de cómputo, normalmente un procesador o un dispositivo de hardware similar. Además muestra la topología del hardware sobre el que se ejecuta el sistema.

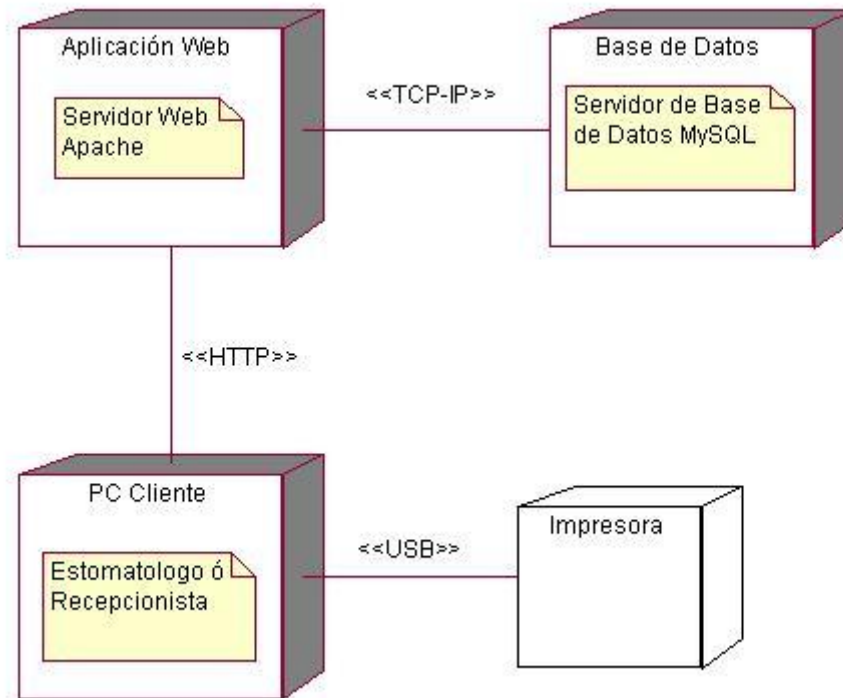


Figura 29. Diagrama de Despliegue

## 4.7 Modelo de Implementación

El modelo de implementación describe cómo los elementos de diseño se implementan en componentes y tiene como objetivo planificar integraciones del sistema, distribuir el sistema en nodos, implementar clases, subsistemas y probar componentes individualmente. Es una colección de componentes que incluyen ficheros de código fuente, binarios y ejecutables. Los componentes representan todos los tipos de elementos software que entran en la fabricación de aplicaciones informáticas. Pueden ser simples archivos, paquetes, bibliotecas cargadas dinámicamente, etc. (25)

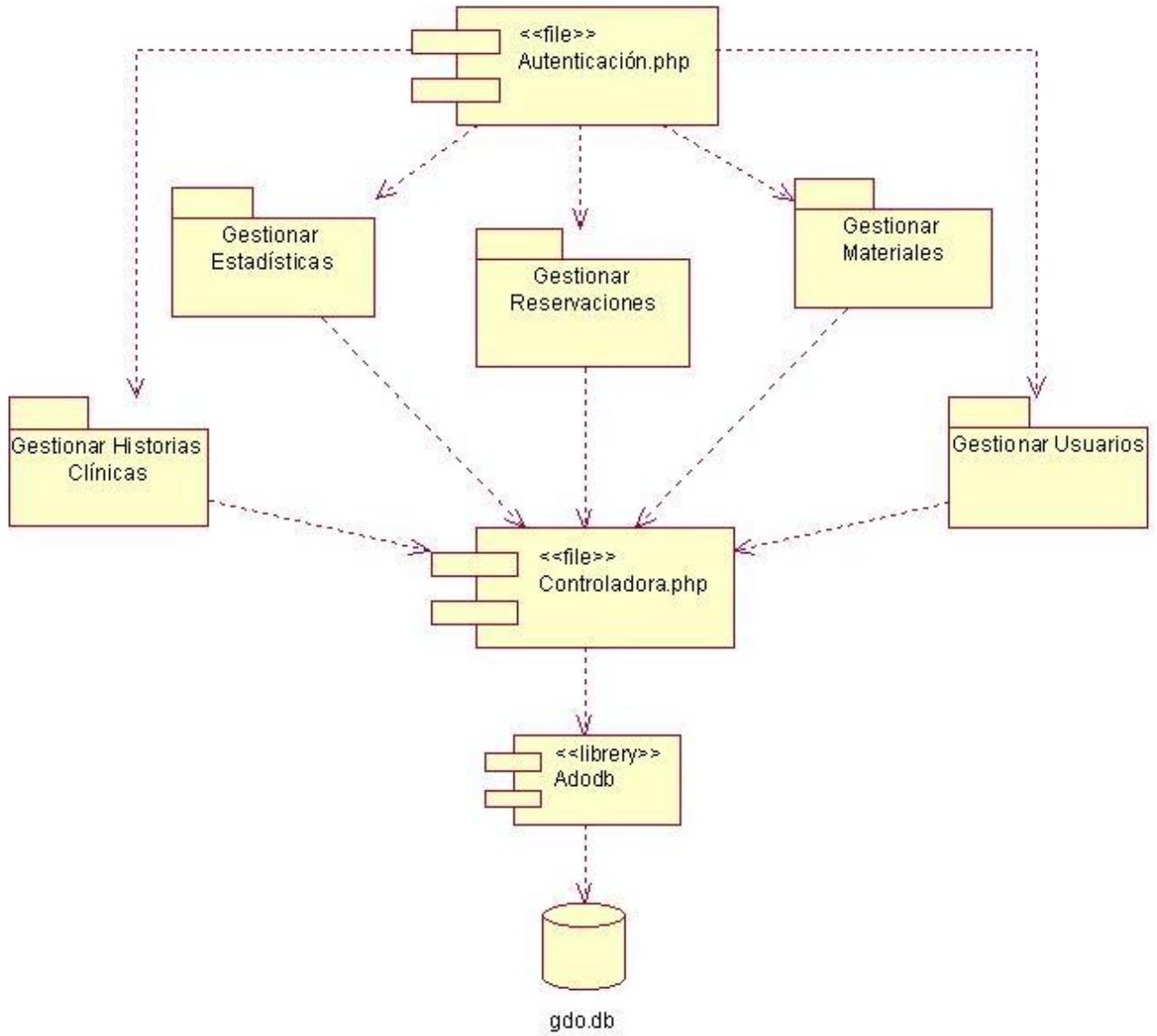


Figura 30. Diagrama de Implementación

## 4.8 Conclusiones

En el presente capítulo se desarrollaron los diagramas de clases de la aplicación y el diseño de la base de datos del sistema. Se describieron los principios de diseño seguidos, específicamente los temas de estándares de la interfaz, concepción del tratamiento de errores y sistema de ayuda, además de los diagramas de componentes y el modelo de despliegue.



## CAPÍTULO 5

### Estudio de Factibilidad

#### 5.1 Introducción

Para la realización de un proyecto es de gran importancia el análisis del costo y los beneficios que reportará. Como resultado de este análisis se obtiene el tiempo de desarrollo en meses, costo y la cantidad de personas que se necesitan para desarrollar el proyecto.

En este capítulo se describe la estimación de costos del sistema propuesto y sus beneficios.

#### 5.2 Planificación basada en casos de usos

Paso 1. Cálculo de los Puntos de casos de uso Desajustados.

$$UUCP = UAW + UUCW$$

Donde:

UUCP: Puntos de casos de uso sin ajustar.

UAW: Factor de peso de los actores sin ajustar.

UUCW: Factor de peso de los casos de uso sin ajustar.

Tipo de actor	Descripción	Factor de peso	Actores	Total
Simple	Sistema con sistema a través de interfaz de programación.	1	0	0
Medio	Sistema con sistema a través de interfaz basada en texto.	2	0	0
Complejo	Persona que interactúa con el sistema mediante interfaz gráfica	3	4	12
Total				12

Tabla 18. Factor de peso de los actores sin ajustar

$UAW = \Sigma \text{ cantidad de actores} * \text{peso}$

$UAW = 12$

Tipo de CU	Descripción	Peso	Cantidad de CU	Total
Simple	El caso de uso tiene de 1 a 3 transacciones.	5	8	40
Medio	El caso de uso tiene de 4 a 7 transacciones.	10	6	60
Complejo	El caso de uso tiene más de 8 transacciones.	15	0	0

**Tabla 19. Factor de peso de los casos de uso sin ajustar**

$UUCW = \Sigma \text{ cantidad de CU} * \text{Peso}$

$UUCW = 100$

$UUCP = 12 + 100$

$UUCP = 112$

Paso 2. Cálculo de los Puntos de casos de uso ajustados.

$UCP = UUCP * TCF * EF$

Donde:

UCP: Puntos de casos de uso ajustados.

UUCP: Puntos de casos de uso sin ajustar.

TCF: Factor de complejidad técnica.

EF: Factor de ambiente.

El factor de complejidad técnica (TCF) se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada factor se cuantifica en un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Significado de los valores:

0: No presente o sin influencia.

1: Influencia incidental o presencia incidental.

2: Influencia moderada o presencia moderada.

3: Influencia media o presencia media.

4: Influencia significativa o presencia significativa.

5: Fuerte influencia o fuerte presencia.

Factor	Descripción	Peso	Valor Asig.	Total
T1	Sistema distribuido	2	0	0
T2	Objetivos de performance o tiempo de respuesta	1	5	5
T3	Eficiencia del usuario final	1	5	5
T4	Procesamiento interno complejo	1	2	2
T5	El código debe ser reutilizable	1	5	5
T6	Facilidad de instalación	0.5	5	2.5
T7	Facilidad de uso	0.5	5	2.5
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	5	5
T10	Concurrencia	1	1	1
T11	Incluye objetivos especiales de seguridad	1	3	3
T12	Provee acceso directo a terceras partes	1	0	0
T13	Se requieren facilidades especiales de entrenamiento a los usuarios	1	2	2
Total				43

**Tabla 20. Factor de complejidad técnica**

$$TCF = 0.6 + 0.01 * \sum (\text{peso} * \text{valor asignado})$$

$$TCF = 0.6 + 0.01 * 43$$

$$TCF = 1.03$$

El factor de ambiente (EF) está relacionado con las habilidades y entrenamiento del grupo de desarrollo que realiza el sistema. Cada factor se cuantifica con un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante).

Factor	Descripción	Peso	Valor asignado	Total
E1	Familiaridad con el modelo de proyecto utilizado.	1.5	3	4.5
E2	Experiencia en la aplicación.	0.5	2	1
E3	Experiencia en la orientación a objetos	1	4	4
E4	Capacidad de analista líder.	0.5	4	2
E5	Motivación.	1	5	5
E6	Estabilidad de requerimientos.	2	4	8
E7	Personal Part - Time	-1	3	-3
E8	Dificultad del lenguaje de programación.	-1	3	-3
Total				17.5

**Tabla 21. Factor de ambiente**

$$EF = 1.4 - 0.03 * \Sigma (\text{peso} * \text{valor asignado})$$

$$EF = 1.4 - 0.03 * 17.5$$

$$EF = 1.4 - 0.53$$

$$EF = 0.87$$

$$UCP = UUCP * TCF * EF$$

$$UCP = 112 * 1.03 * 0.87$$

$$UCP = 100.36$$

Paso 3. Estimación de esfuerzo a través de los puntos de casos de uso.

$$E = UCP * CF$$

Donde:

E: Esfuerzo estimado en horas hombres.

UCP: Punto de casos de usos ajustados.

CF: Factor de conversión.

Para obtener el factor de conversión (CF) se cuentan cuantos valores de los que afectan el factor ambiente (E1...E6) están por debajo de la media (3), y los que están por arriba de la media para los restantes (E7, E8). Si el total es 2 o menos se utiliza el factor de conversión 20 Horas-Hombre / Punto de Casos de uso. Si el total es 3 o 4 se utiliza el factor de conversión 28 Horas-Hombre / Punto de Casos de uso. Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso del mismo es demasiado alto.

En este caso se puede decir que:

$CF = 20 \text{ Horas-Hombre} / \text{Punto de Casos de uso.}$

$E = 100.3634 * 20$

$E = 2007.268 \text{ Horas-Hombre}$

Paso 4. Calcular esfuerzo de todo el proyecto.

Actividad	Porcentaje %	Horas/Hombre
Análisis	10%	501.817 Horas-Hombre
Diseño	20%	1003.634 Horas-Hombre
Implementación	40%	2007.268 Horas-Hombre
Pruebas	15%	752.725 Horas-Hombre
Sobrecarga (otras actividades)	15%	752.725 Horas-Hombre
Total	100%	5018.17 Horas-Hombre

**Tabla 22. Esfuerzo del Proyecto**

Si ET = 5018.17 horas-hombre y cada mes tiene como promedio 192 horas, eso daría un

ET = 26.136 mes-hombre

Esto quiere decir que 1 persona puede realizar el problema analizado en 26 meses.

### 5.3 Costos

Se asume como salario promedio mensual \$289.00

CH: Cantidad de hombres

Tiempo: Tiempo total del Proyecto.

CH = 2 hombres.

CHM = 2 hombres \* Salario.

CHM = 578 \$/mes

Costo = CHM \* ET/2

Costo = 578 \* 26.136 /2

Costo = \$ 7553.304

#### Tiempo total del Proyecto:

Tiempo = ET / CH

Tiempo = 26.136 meses /2 hombres

Tiempo = 13.068 meses

### 5.4 Beneficios Tangibles e Intangibles

El sistema propuesto tiene como objetivo principal resolver los problemas que existen durante la gestión de la información que se genera en el Departamento de Estomatología de la UCI garantizando la integridad de la información almacenada.

El beneficio fundamental del sistema es contar con una aplicación Web flexible, dinámica y de interfaz agradable que permita registrar, actualizar y conocer de una forma más precisa y en el menor tiempo posible la información referente a las consultas.

Por tanto, los beneficios inmediatos son generalmente intangibles:

- Disminución del tiempo y esfuerzo que se invierte en esta tarea que se realiza, hasta ahora, de forma manual.
- Disminución de la acumulación de materiales impresos relacionados con los procesos de gestión de la información referente a las historias clínicas.
- Disminución de los gastos pues resulta menos costoso crear y procesar información digital que copias duras.
- Fácil detección de errores.
- Fácil y rápido acceso y publicación de la información actualizada.
- Fácil procesamiento de la información y obtención.

## 5.5 Análisis de costos y beneficios

Una vez implantado el sistema este contribuirá a aumentar la eficiencia de los servicios brindados en el Departamento de Estomatología durante la realización de una consulta estomatológica así como a su organización disminuyendo el espacio ocupado por el exceso de control documental, dando respuestas rápidas a pedidos administrativos, controlando la productividad individual y colectiva y mejorando la planificación de actividades.

La tecnología utilizada para el desarrollo del sistema es completamente libre por lo que contribuye con el ahorro monetario por concepto de software libre.

## 5.6 Conclusiones

En este capítulo se describió el estudio de factibilidad realizado correspondiente al sistema propuesto, teniendo en cuenta el costo estimado y los beneficios que reportará al ser implantado.

La herramienta propuesta reportará beneficios significativos e importantes para la gestión de las consultas estomatológicas que se desarrollan en la UCI, al contribuir a mejorar los procesos que se desarrollan en función de controlar la información de los mismos, lo que indica que es factible implementar la herramienta propuesta.

## CONCLUSIONES

Los sistemas de Gestión de la Información presentan soluciones eficaces al procesamiento de datos, ha sido objetivo de este trabajo mostrar una solución a esta problemática, enfocada fundamentalmente al procesamiento de información en las consultas estomatológicas.

El resultado es la obtención del Sistema para la Gestión Estomatológica GEDONT, el cual es adaptable a la sociedad cubana actual y su naciente desarrollo informático, que alcanzara magnitudes mayores en los próximos años. Se logró su implementación con herramientas de desarrollo libres, y la posibilidad de su despliegue en los Sistemas Operativos Linux y Windows.

El desarrollo de este trabajo de tesis está orientado a la concepción de una herramienta informática para la gestión de la información durante la realización de una consulta estomatológica en la UCI. El valor fundamental de esta herramienta se expresa en la contribución a simplificar el trabajo y la demora que produce el procesamiento manual de la información.

Con el estudio realizado a lo largo de todas las etapas de desarrollo y el sistema que se ha logrado diseñar e implementar, se cumple con el objetivo general del proyecto: desarrollar una solución robusta, flexible y única de software que dé soporte a los procesos relacionados con las consultas estomatológicas, reafirmando así la utilidad y validez de emplear las tecnologías informáticas para apoyar las labores que se desarrollan en cualquier tipo de esfera.

Con la implementación de esta aplicación se logra:

Viabilizar el almacenamiento de los datos y lograr disponibilidad a la hora de las revisiones y con esto la velocidad de la prestación de los servicios. Eliminar todo tipo de documentos a la hora de las comunicaciones entre los especialistas y con esto un ahorro de materiales, Portabilidad del sistema y su adaptabilidad a cambios de estructura.

Se ha demostrado la eficacia de los lenguajes y tecnologías utilizadas para el desarrollo del sistema, se realizó una base de datos, donde se almacena toda la información necesaria que se genera de los procesos relacionados, para de esta forma garantizar la veracidad y centralización de la misma, se realizó el análisis, diseño e implementación del sistema, se han seguido los principios básicos de diseño descritos





para el desarrollo del sistema y se logra una seguridad y protección de los datos consecuente con el nivel de seguridad requerido.

## RECOMENDACIONES

Se recomienda:

- Poner a prueba el sistema durante un período de tiempo significativo, para comprobar su desempeño y que las funcionalidades del sistema se correspondan con la actividad que se está gestionando.
- Continuar el estudio con el objetivo de añadir nuevas funcionalidades al sistema como la implementación de un módulo de reservaciones online permitiendo este servicio a la comunidad UCI como comunidad altamente informatizada que es. Así como también nuevas funcionalidades estadísticas para la gestión interna de la clínica.
- Proponer la utilización del sistema en las clínicas estomatológicas existentes en el país adaptando el sistema a las condiciones de las mismas.

## BIBLIOGRAFÍA

1. Arsys. *Programación Web*. [En línea] [Citado el: 20 de Marzo de 2007.] <http://www.arsys.es/soporte/programacion/windows.htm>.
2. Valido, Y. y Moreira, Y. SAImm: Sistema de Apoyo Integral a la Misión Milagro. Trabajo de Diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico “José Antonio Echeverría”, Ciudad de la Habana, junio 2005.
3. **Gil, Ríos**. Sistemas de Información. *Departamento de Ingeniería Telemática Escuela Politécnica Superior Universidad Carlos III de Madrid*. [En línea] [Citado el: 20 de Marzo de 2007.] [http://www.it.uc3m.es/mcfp/docencia/si/material/1\\_cli-ser\\_mcfp.pdf](http://www.it.uc3m.es/mcfp/docencia/si/material/1_cli-ser_mcfp.pdf).
4. Méndez, L. y Torres, A. Sistema de Promoción y Gestión Comercial para la oficina de Transferencia Tecnológica de la Universidad de Cienfuegos. Trabajo de Diploma para optar por el título de Ingeniero Informático, Instituto Superior Politécnico “José Antonio Echeverría”, Ciudad de la Habana, junio 2005.
5. **MÉNDEZ, DUQUE y D, Néstor**. *Conceptos de arquitectura Cliente/Servidor*. [En línea] [Citado el: 20 de Marzo de 2007.] [http://www.it.uc3m.es/mcfp/docencia/si/material/1\\_cli-ser\\_mcfp.pdf](http://www.it.uc3m.es/mcfp/docencia/si/material/1_cli-ser_mcfp.pdf).
6. Eliurkis Días Terrero: Sistema Automatizado para la Gestión de Información de la Misión Milagro. Trabajo de Diploma para optar por el título de Ingeniero Informático, “Universidad de las Ciencias Informáticas”, Ciudad de la Habana, junio 2006.
7. **Larman**. *UML y Patrones Introducción al análisis y diseño orientado a objetos*. La Habana : Editorial Félix Varela, 2004.
8. Webexperto. *PHP.Net*. [En línea] [Citado el: 25 de Marzo de 2007.] <http://www.webexperto.com/articulos/art/217/historia-de-php>.
9. DesarrolloWeb. *Historia de PHP*. [En línea] [Citado el: 25 de Marzo de 2007.] <http://www.desarrolloweb.com/articulos/436.php>.
10. **Der Henst S, Christian Van**. Maestro del Web. *¿Qué es el PHP?* [En línea] [Citado el: 25 de Marzo de 2007.] <http://www.maestrosdelweb.com/editorial/phpintro>.
11. CompletHost. *Servidores Web*. [En línea] [Citado el: 2 de Abril de 2007.] [http://www.complethost.com/servidor\\_web.php](http://www.complethost.com/servidor_web.php).



12. BarraPunto. *El servidor de web Apache*. [En línea] [Citado el: 2 de Abril de 2007.] <http://acs.barrapunto.org/articulos/trunk/LinuxActual/Apache/html/c20.html>.
13. DesarrolloWeb. *Arquitectura del servidor Apache*. [En línea] [Citado el: 2 de Abril de 2007.] <http://www.desarrolloweb.com/articulos/1112.php>.
14. Introducción a la Documática. *Los sistemas de gestión de bases de datos*. [En línea] [Citado el: 5 de Abril de 2007.] <http://tramullas.com/documatica/2-4.html>.
15. Error500. *Sistema Gestor de base de datos SGBD*. [En línea] [Citado el: 5 de Abril de 2007.] [http://www.error500.net/garbagecollector/archives/categorias/bases\\_de\\_datos/sistema\\_gestor\\_de\\_base\\_de\\_datos\\_sgbd.php](http://www.error500.net/garbagecollector/archives/categorias/bases_de_datos/sistema_gestor_de_base_de_datos_sgbd.php).
16. SoftWare Shop. *MySQL*. [En línea] [Citado el: 5 de Abril de 2007.] <http://www.software-shop.com/Productos/MySQL/mysql.html>.
17. *PostgreSQL vs MySQL*. [En línea] [Citado el: 6 de Abril de 2007.] [http://www.netpecos.org/docs/mysql\\_postgres/x57.html](http://www.netpecos.org/docs/mysql_postgres/x57.html).
18. *Proceso Unificado para Desarrollo de Software (RUP)*. [En línea] 7 de Mayo de 2001. [Citado el: 10 de Abril de 2007.] <http://atenea.ucauca.edu.co/~gramirez/archivos/AnotacionesRUP.pdf>.
19. **Molpeceres, Alberto**. *Procesos de desarrollo: RUP, XP y FDD*. [En línea] 15 de Diciembre de 2002. [Citado el: 10 de Abril de 2007.]
20. *EL LENGUAJE UNIFICADO DE MODELADO (UML)*. [En línea] [Citado el: 10 de Abril de 2007.] <http://www.fi-b.unam.mx/pp/profesores/carlos/aydoo/uml.html>.
21. El Rincón del Vago. *EL DESARROLLO DE SISTEMAS DE INFORMACIÓN EMPLEANDO EL LENGUAJE DE MODELADO UNIFICADO UML*. [En línea] [Citado el: 10 de Abril de 2007.] [http://html.rincondelvago.com/uml\\_5.html](http://html.rincondelvago.com/uml_5.html)
22. *Modelado de Sistemas con UML*. [En línea] [Citado el: 11 de Abril de 2007.] <http://es.tldp.org/Tutoriales/doc-modelado-sistemas-UML/multiple-html/c12.html>.
23. **José Ortín, María, y otros**. *El Modelo del Negocio como base del Modelo de Requisitos. Grupo de Investigación de Ingeniería de Software, Departamento de Informática y Sistemas, Facultad de Informática, Universidad de Murcia*. [En línea] [Citado el: 15 de Mayo de 2007.] [http://www.lsi.us.es/~amador/JIRA/Ponencias/JIRA\\_Ortin.pdf](http://www.lsi.us.es/~amador/JIRA/Ponencias/JIRA_Ortin.pdf).



24. *Aplicación de estándares web, accesibilidad y usabilidad*. [En línea] 21 de Septiembre de 2006 [Citado el: 7 de Junio de 2007.] <http://www.dosmilmastres.com/blog.php?anotacionid=4>
25. **I.T.I.S y I.T.I.G.** *Modelo de Implementación: Diagramas de Componentes y Despliegue*. [En línea] [Citado el: 11 de Abril de 2007.] <http://www.info-ab.uclm.es/assignaturas/42530/pdf/M2tema12.pdf>.
26. *ADODB, librería PHP para bases de datos*. [En línea] 22 de Abril de 2006 [Citado el: 11 de Abril de 2007.] <http://www.faq-mac.com/bitacoras/mmm/?p=166>
27. **Javier Perez**. *¿Qué es Smarty?*. [En línea] 17 de Mayo de 2007 [Citado el: 11 de Abril de 2007.] <http://www.maestrosdelweb.com/editorial/que-es-smarty/>



## GLOSARIO DE TÉRMINOS

- **UCI:** Universidad de las Ciencias Informáticas.
- **UML:** Unified Modeling Language. Es una notación estándar para modelar objetos del mundo real como primer paso en el desarrollo de programas orientados a objetos.
- **WEB:** Red de documentos HTML intercomunicados y distribuidos entre servidores del mundo entero.
- **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.
- **RUP:** Rational Unified Process (Proceso Unificado de desarrollo). Metodología para el desarrollo de Software.
- **PHP:** Hypertext Preprocessor. Es un ambiente script del lado del servidor que permite crear y ejecutar aplicaciones Web dinámicas e interactivas.
- **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.
- **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.
- **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.
- **HTTP:** HyperText Transfer Protocol. Protocolo de Transferencia de Hipertextos. Modo de comunicación para solicitar páginas Web.
- **HTML:** HyperText Markup Language. 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".

- **ADODB:** Es un conjunto de librerías de bases de datos para PHP y Python. Esta permite a los programadores desarrollar aplicaciones web de una manera portable, rápida y fácil.
- **Smarty:** Es un motor de plantillas para PHP, cuyo objetivo es separar el contenido de la presentación en una página web.
- **XML:** Extensible Markup Language. Es un lenguaje extensible de etiquetas desarrollado por el World Wide Web Consortium. Orientado principalmente al almacenamiento, procesamiento y transmisión de mensajes.
- **CUN:** Caso de Uso del Negocio.
- **CUS:** Caso de Uso del Sistema.
- **GEDONT:** Gestión odontológica.
- **COP:** Cariacos Obturados Perdidos.
- **IDEs:** Integrated Development Environment. Entorno Integrado de Desarrollo.
- **XHTML:** eXtensible HyperText Markup. Es un lenguaje de descripción pensado para destronar al HTML como estándar para hacer páginas web. Es la versión XML de HTML.
- **GTK:** GIMP Toolkit. Es una biblioteca que posibilita la creación de interfaces de usuario gráficas (Graphical User Interfaces, GUI). Está disponible bajo licencia GPL.
- **ASP:** Active Server Pages. Es una tecnología del lado servidor de Microsoft para páginas web generadas dinámicamente, que ha sido comercializada como un anexo a Internet Information Server (IIS).



## ANEXOS

Anexo 1. Precios del Odontosoft por ordenadores en uso .....	117
Anexo 2. Vista general de RUP .....	117
Anexo 3. Flujos de trabajo de RUP .....	118
Anexo 4. Descripción textual del caso de Uso del sistema “Remitir a Especialista” .....	118
Anexo 5. Descripción textual del caso de Uso del sistema “Indicar Medicamentos” .....	120
Anexo 6. Descripción textual del caso de Uso del sistema “Remitir a Ingreso” .....	121
Anexo 7. Descripción textual del caso de Uso del sistema “Parte Gasto Material” .....	122
Anexo 8. Descripción textual del caso de Uso del sistema “Gestionar Materiales” .....	123
Anexo 9. Descripción textual del caso de Uso del sistema “Visualizar Materiales” .....	124
Anexo 10. Opiniones y Avals .....	126



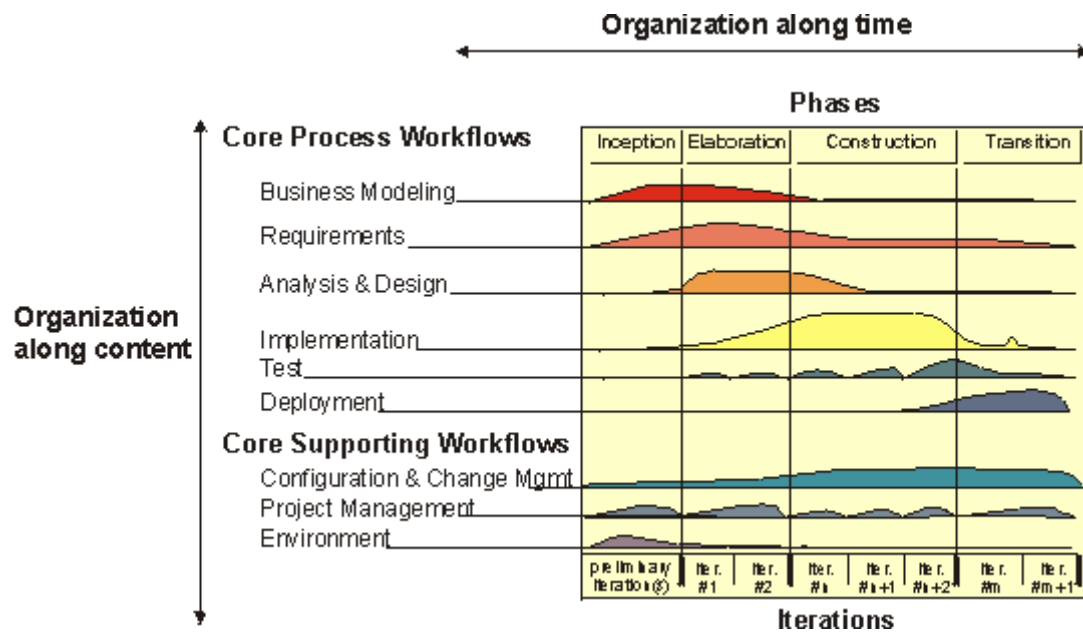


**Anexo 1. Precios del Odontosoft por ordenadores en uso**

<b>Versión</b> (todas sin límite de odontólogos)	<b>Precio</b>
Básico	u\$s 199
Mono usuario	u\$s 390
Multiusuario hasta 5 computadoras simultáneas	u\$s 780
Multiusuario hasta 10 computadoras simultáneas	u\$s 995
Multiusuario más de 10 computadoras simultáneas	Consultar

**Tabla 23. Precios de PC por versiones**

**Anexo 2. Vista general de RUP**



**Figura 31. Vista general de RUP**

**Anexo 3. Flujos de trabajo de RUP**



**Figura 32. Flujos de trabajo de RUP**

**Anexo 4. Descripción textual del caso de Uso del sistema “Remitir a Especialista”**

<b>Nombre del Caso de Uso</b>	<b>Remitir a Especialista.</b>
<b>Actores</b>	Estomatólogo (inicia).
<b>Propósito</b>	Permitir al estomatólogo remitir un paciente al especialista.
<b>Resumen</b>	Inicia cuando el estomatólogo selecciona la opción para crear una nueva orden de remisión de un paciente a un especialista.
<b>Referencias</b>	R-9
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.
<b>Pos condiciones</b>	Para las siguiente acción: Remisión Especialista: Se crea una nueva receta médica.
<b>Curso Normal de los Eventos</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
1. El estomatólogo solicita remitir un paciente a un especialista.	1.1 El sistema muestra una interfaz para entrar los datos

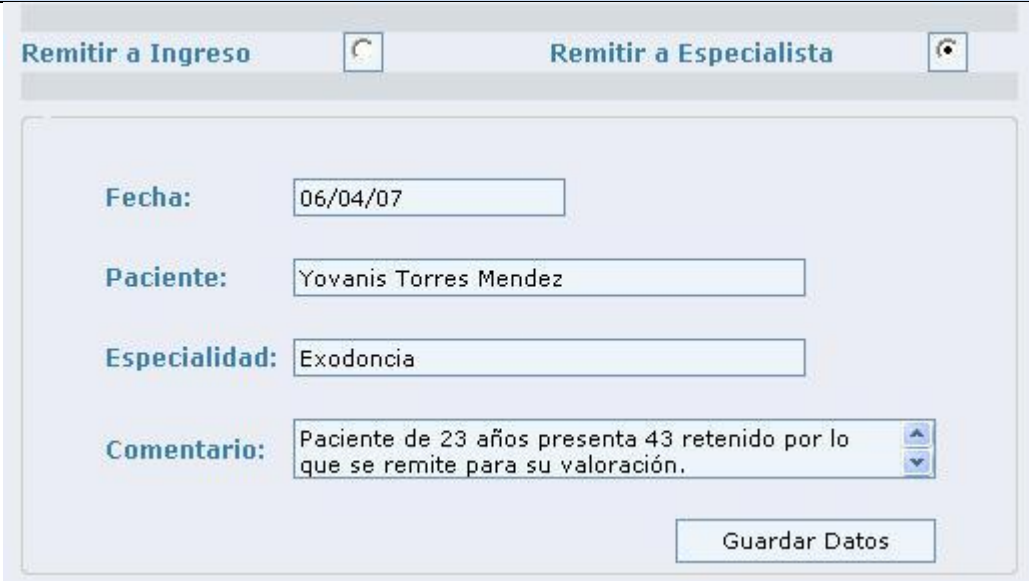
	necesarios.
2. El estomatólogo entra los datos para crear la orden de remisión.	2.1 El sistema valida los datos insertados y los archiva en la base de datos. 2.2 Posteriormente el sistema muestra un mensaje indicando que los datos fueron insertados correctamente.
<b>Curso alternativo</b>	
	2.1 Emite un mensaje indicando que debe llenar los campos obligatorios.
<b>Prototipo</b>	
	
<b>Prioridad:</b>	Secundario

Tabla 24. Descripción detallada del CUS “Remitir a Especialista”

## Anexo 5. Descripción textual del caso de Uso del sistema “Indicar Medicamentos”


<b>Nombre del Caso de Uso</b>	Indicar Medicamentos.	
<b>Actores</b>	Estomatólogo (inicia).	
<b>Propósito</b>	Permitir al estomatólogo indicar medicamentos a un paciente.	
<b>Resumen</b>	Inicia cuando el estomatólogo selecciona la opción para crear una nueva receta médica a un paciente.	
<b>Referencias</b>	R-8	
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para las siguientes acción: Crear Receta: Se crea una nueva receta médica.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El estomatólogo solicita crear receta médica.	1.1 El sistema muestra una interfaz para entrar los datos necesarios.	
2. El estomatólogo entra los datos para crear la receta médica.	2.1 El sistema valida los datos insertados y los archiva en la base de datos. 2.2 Posteriormente el sistema muestra un mensaje indicando que los datos fueron guardados satisfactoriamente.	
<b>Curso alternativo</b>		
	2.1 Emite un mensaje indicando que debe llenar los campos obligatorios.	
<b>Prototipo</b>		
		
<b>Prioridad:</b>	Secundario	

Tabla 25. Descripción detallada CUS “Indicar Medicamentos”



## Anexo 6. Descripción textual del caso de Uso del sistema “Remitir a Ingreso”

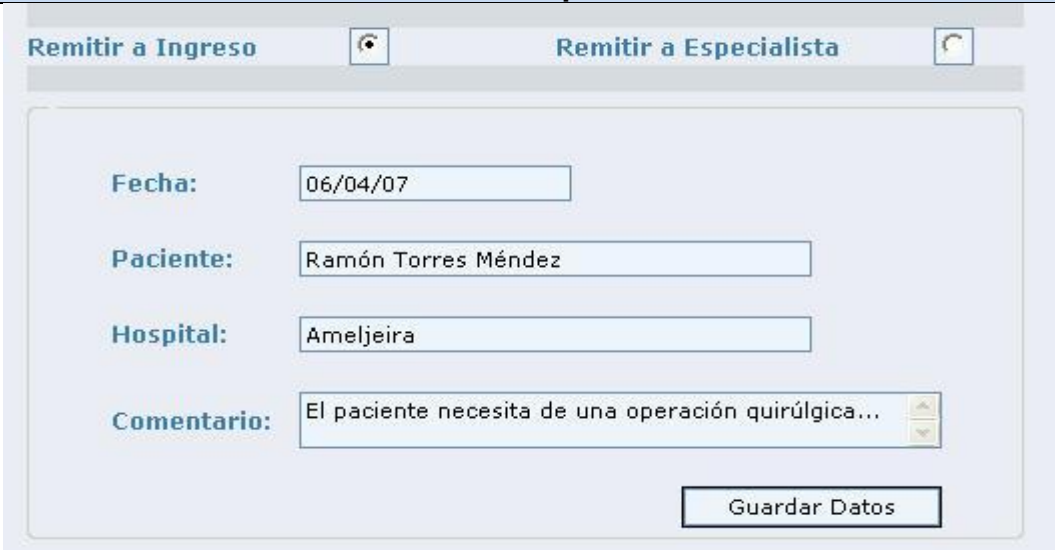
<b>Nombre del Caso de Uso</b>	<b>Remitir a Ingreso.</b>	
<b>Actores</b>	Estomatólogo (inicia).	
<b>Propósito</b>	Permitir al estomatólogo remitir un paciente a ingreso.	
<b>Resumen</b>	Inicia cuando el estomatólogo selecciona la opción para crear una nueva orden de remisión de un paciente a ingreso.	
<b>Referencias</b>	R-10	
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para la siguiente acción: Remisión Ingreso: Se crea una nueva orden de remisión a ingreso.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El estomatólogo solicita remitir un paciente a ingreso.	1.1 El sistema muestra una interfaz para entrar los datos necesarios.	
2. El estomatólogo entra los datos para crear la orden de remisión.	2.1 El sistema valida los datos insertados y los archiva en la base de datos. 2.2 Posteriormente el sistema muestra un mensaje indicando que los datos fueron guardados satisfactoriamente.	
<b>Curso alternativo</b>		
	2.1 Emite un mensaje indicando que debe llenar los campos obligatorios.	
<b>Prototipo</b>		
		
<b>Prioridad:</b>	Secundario	

Tabla 26. Descripción detallada del CUS “Remitir a Ingreso”



## Anexo 7. Descripción textual del caso de Uso del sistema “Parte Gasto Material”


<b>Nombre del Caso de Uso</b>	<b>Generar Parte de Gasto Material.</b>	
<b>Actores</b>	Estomatólogo (inicia).	
<b>Propósito</b>	Permitir al estomatólogo tener al final de la jornada laboral una parte del gasto material que ha tenido.	
<b>Resumen</b>	Inicia cuando el estomatólogo selecciona la opción para crear parte de gasto material.	
<b>Referencias</b>	R-12	
<b>Precondiciones</b>	El estomatólogo debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para la siguiente acción: Guardar Parte: Se guarda una anotación del parte de gasto material.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El estomatólogo selecciona la opción de Parte de Gasto Material	1.1 El sistema muestra una interfaz para entrar los datos de una nueva anotación de gasto material.	
2. El estomatólogo entra los datos de una nueva anotación.	2.1 El sistema valida los datos insertados y los guarda. 2.2 Posteriormente el sistema muestra una interfaz para seguir entrando datos y el listado de anotaciones ya insertadas además de un mensaje indicando que los datos fueron guardados satisfactoriamente.	
<b>Curso alternativo</b>		
	2.1 Emite un mensaje indicando que debe llenar los campos obligatorios.	
<b>Prototipo</b>		
		
<b>Prioridad:</b>	Secundario	

Tabla 27. Descripción detallada del CUS “Parte Gasto Material”



## Anexo 8. Descripción textual del caso de Uso del sistema “Gestionar Materiales”

<b>Nombre del Caso de Uso</b>	<b>Gestionar Materiales Educativos.</b>	
<b>Actores</b>	Jefe de Departamento (inicia).	
<b>Propósito</b>	Permitir al JDpto. Publicar o eliminar materiales educativos.	
<b>Resumen</b>	Inicia cuando el JDpto. Selecciona la opción para gestionar Materiales Educativos.	
<b>Referencias</b>	R-13	
<b>Precondiciones</b>	El JDpto. debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para la siguiente acción: Publicar Material: Se sube el archivo al servidor para que esté disponible su visualización. Eliminar Material: Se elimina el material el cual ya no estará publicado en el sistema.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El JDpto. selecciona la opción de gestionar materiales educativos	1.1 El sistema muestra una interfaz para publicar o eliminar materiales educativos.	
2. El JDpto. selecciona Publicar Materiales Educativos, Eliminar Materiales Educativos.	2.1 El sistema ejecuta una de las acciones siguientes: a) Si selecciona Publicar Materiales educativos ir a la sección “Publicar Material”. b) Si selecciona Eliminar Material ir a la sección “Eliminar Material”.	
<b>Sección “Publicar Materiales”</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
	2.1 El sistema muestra una interfaz para publicar materiales.	
3. El JDpto. elige la categoría del material y sube el archivo al servidor	3.1 El sistema guarda los datos y publica el material educativo.	
<b>Curso alternativo</b>		
3. El JDpto. inserta una nueva categoría de Materiales y además una breve descripción de la misma	3.1 El sistema guarda la una nueva categoría la cual estará disponible para elegir al subir un material.	
4. El JDpto. elige la categoría del material y sube el archivo al servidor	4.1 El sistema guarda los datos y publica el material educativo.	
<b>Prototipo</b>		





	
<b>Sección "Eliminar Materiales"</b>	
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>
	2.1 El sistema muestra una interfaz con el listado de los materiales educativos.
3. El JDpto. elimina el material educativo que desee	3.1 El sistema elimina el material educativo seleccionado del sistema con lo que deja de estar publicado, mostrando un mensaje indicando que el material ha sido eliminado del sistema.
<b>Curso Alternativo</b>	
<b>Prototipo</b>	
	
<b>Prioridad:</b>	Secundario

Tabla 28. Descripción detallada del CUS "Gestionar Materiales"

## Anexo 9. Descripción textual del caso de Uso del sistema "Visualizar Materiales"

<b>Nombre del Caso de Uso</b>	<b>Visualizar Materiales.</b>
<b>Actores</b>	Usuario (inicia).







<b>Propósito</b>	Permitir al usuario visualizar cualquier material publicado en el sistema.	
<b>Resumen</b>	Inicia cuando el usuario selecciona la opción para visualizar un material.	
<b>Referencias</b>	R-12	
<b>Precondiciones</b>	El usuario debe haberse autenticado en el sistema.	
<b>Pos condiciones</b>	Para la siguiente acción: Visualizar Material: Se muestra el material educativo.	
<b>Curso Normal de los Eventos</b>		
<b>Acciones del Actor</b>	<b>Respuesta del Sistema</b>	
1. El usuario selecciona Visualizar Materiales educativos.	1.1 El sistema muestra una interfaz con el listado de categorías de materiales educativos.	
2. El usuario selecciona la categoría de materiales que desee ver.	2.1 El sistema muestra una interfaz con el listado de materiales de la categoría seleccionada.	
3. El usuario selecciona el material que desea ver.	3.1 El sistema muestra una interfaz para ver el material.	
<b>Prototipo</b>		
 <p>The screenshot shows a light blue background with a white box containing the text: "Caries Dental:  La caries dental se define como un proceso o enfermedad dinámica crónica, que ocurre en la estructura dentaria en contacto con los depósitos microbianos." The text is in a light blue font.</p>		
<b>Prioridad:</b>	Secundario	

Tabla 29. Descripción detallada del CUS “Visualizar Materiales”



## Anexo 10. Opiniones y Avals



Figura 33. Diploma por haber obtenido la condición de Relevante en la 4ta Jornada Científica, Policlínico "Ernesto Che Guevara"

