

UNIVERSIDAD PÚBLICA DE EL ALTO  
CARRERA INGENIERÍA DE SISTEMAS



PROYECTO DE GRADO

**“SISTEMA DE GESTIÓN DE INFORMACIÓN DE FICHAS  
MÉDICAS E HISTORIAL CLÍNICO EN LA ESPECIALIDAD  
DE ODONTOLOGÍA”**

CASO: CENTRO DE SALUD “VILLA COOPERATIVA”

Para optar al título de Licenciatura en Ingeniería de Sistemas

Mención: GESTIÓN Y PRODUCCIÓN

**Postulante: Univ. Marina Mamani Uluri**

**Tutor Metodológico: Ing. Maricel Yarari Mamani**

**Tutor Especialista: M.Sc. Dulfredo Villca Lázaro**

**Tutor Revisor: Lic. Carmen Vega Flores**

EL ALTO-BOLIVIA

2019

# ÍNDICE

<b>1. MARCO PRELIMINAR.....</b>	<b>1</b>
1.1. INTRODUCCIÓN .....	1
1.2. ANTECEDENTES.....	2
1.2.1. Antecedentes Institucionales .....	2
1.2.2. Antecedentes Académicos:.....	2
1.3. PLANTEAMIENTO DEL PROBLEMA.....	4
1.3.1. Problema Principal.....	4
1.3.2. Problemas Secundarios.....	4
1.4. OBJETIVOS.....	5
1.4.1. Objetivo General.....	5
1.4.2. Objetivos Específicos.....	5
1.5. JUSTIFICACIÓN .....	6
1.5.1. Técnica.....	6
1.5.2. Económica.....	6
1.5.3. Social.....	6
1.6. METODOLOGÍA.....	7
1.6.1. Metodología RUP (Rational Unified Process).....	7
1.6.2. UML (Lenguaje Modelado Unificado).....	7
1.7. HERRAMIENTAS.....	8
1.8. LÍMITES Y ALCANCES.....	11
1.8.1. Límites .....	11
1.8.1. Alcances.....	11
1.9. APORTES:.....	11
<b>2. MARCO TEÓRICO.....</b>	<b>13</b>
2.1. SISTEMA .....	13
2.2. GESTIÓN.....	14
2.3. INFORMACIÓN.....	14
2.4. GESTIÓN DE INFORMACIÓN .....	15

2.5.	SISTEMA DE GESTIÓN DE INFORMACIÓN .....	15
2.6.	HISTORIAL CLÍNICO .....	16
2.7.	ODONTOLOGÍA.....	16
2.7.1.	Términos Odontológicos: .....	16
2.8.	METODOLOGÍA RUP .....	18
2.9.	UML (LENGUAJE MODELADO UNIFICADO) .....	20
2.9.1.	Diagramas de Caso de Uso.....	21
2.9.2.	Diagrama de Estados.....	23
2.9.3.	Diagrama de Secuencias .....	25
2.9.4.	Diagrama de Colaboraciones .....	27
2.9.5.	Diagramas de Actividades .....	29
2.9.6.	Diagramas de Componentes .....	32
2.9.7.	Diagramas de Distribución.....	33
2.10.	MÉTRICAS DE CALIDAD DE SOFTWARE .....	34
2.10.1.	Métricas de Calidad del Modelo ISO/IEC25000 .....	35
2.10.1.1.	Estándar ISO/IEC 25000 .....	35
2.10.1.2.	Funcionalidad.....	37
2.10.1.3.	Confiabilidad .....	42
2.10.1.4.	Usabilidad .....	43
2.10.1.5.	Mantenibilidad.....	45
2.10.1.6.	Portabilidad .....	46
2.10.1.7.	Eficiencia.....	47
2.11.	MÉTODO DE ESTIMACIÓN DE COSTES DEL SOFTWARE.....	48
2.11.1.	Modelo Constructivo de Costos (COCOMO) .....	48
2.11.2.	Modelos de Estimación .....	48
2.11.2.1.	Puntos Objeto .....	48
2.11.2.2.	Puntos de Función No Ajustados.....	49
2.11.2.3.	Líneas de Código Fuente .....	49
2.12.	MÉTRICAS DE ESTIMACIÓN DE COSTOS COCOMO II.....	50
2.13.	SEGURIDAD DE LA INFORMACIÓN .....	56

2.13.1. Estándar ISO/IEC 27000 .....	56
2.13.1.1. ISO 27002 .....	57
<b>3. MARCO APLICATIVO: .....</b>	<b>58</b>
3.1. Obtención de Requisitos .....	58
3.1.1. Situación Actual de la Institución.....	59
3.1.2. Identificación de Actores .....	59
3.2. INGENIERÍA DE REQUERIMIENTOS.....	60
3.2.1.2. Requerimientos No Funcionales .....	63
3.3. APLICACIÓN DE UML .....	64
3.3.1. MODELO DE CASO DE USO .....	64
3.3.2. Diagrama de Caso de Uso General del Sistema .....	65
3.3.2.1. Diagrama de Caso de Uso: ADMINISTRADOR .....	67
3.3.2.2. Diagrama de Caso de Uso: Asistente.....	68
3.3.2.3. Diagrama de Caso de Uso: Enfermera (Enfermería) .....	70
3.3.2.4. Diagrama de Caso de Uso: Doctor .....	72
3.3.2.5. Diagrama de Caso de Uso: DIRECTOR .....	76
3.3.3. DISEÑO CONCEPTUAL.....	78
3.3.3.1. Modelo Conceptual.....	78
3.3.4. IMPLEMENTACIÓN DEL SISTEMA .....	80
3.3.4.1. Interfaz de Inicio de Sesión o Autenticación .....	80
3.3.4.2. Funcionalidad General del Sistema .....	81
3.3.4.3. Módulos que Integran el Sistema.....	81
3.3.4.3.1. Módulo Asistente .....	81
3.3.4.3.2. Módulo enfermera .....	83
3.3.4.3.3. Módulo doctor .....	84
3.3.4.3.4. Módulo Director .....	88
3.4. MÉTRICAS DE CALIDAD DE SOFTWARE .....	91
3.4.2. Confiabilidad.....	96
3.4.3. Usabilidad.....	97
3.4.4. Mantenibilidad .....	99

3.4.5. Portabilidad .....	100
3.5. ESTIMACIÓN DE COSTES DE SOFTWARE.....	101
3.5.1. Método de Estimación COCOMO II .....	101
3.6. SEGURIDAD DE LA INFORMACIÓN DEL SISTEMA .....	105
3.6.1. Sistema de Gestión de Seguridad de la Información ISO-27002.....	105
3.6.1.1. Seguridad Lógica.....	105
3.6.1.2. Seguridad Física.....	106
3.6.1.3. Seguridad Organizativa .....	106
3.6.2. SEGURIDAD DEL SISTEMA.....	106
<b>4. CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>108</b>
4.1. CONCLUSIONES .....	108
4.2. RECOMENDACIONES .....	109
BIBLIOGRAFÍA.....	110

## ÍNDICE DE TABLAS

Tabla Nº 1.1. Herramienta.....	10
Tabla Nº 2.2. Estereotipos del Diagrama de Casos de Uso.....	22
Tabla Nº 2.3. Estereotipos del Diagrama de Estado .....	24
Tabla Nº 2.4. Estereotipos del Diagrama de Secuencias.....	25
Tabla Nº 2.5. Estereotipos del Diagrama de Colaboración .....	28
Tabla Nº 2.6. Estereotipos del Diagrama de Actividad.....	30
Tabla Nº 2.7. Estereotipos del Diagrama de Componentes .....	32
Tabla Nº 2.8. Estereotipos del Diagrama de Distribución.....	33
Tabla Nº 2.9. Ecuación de la Funcionalidad.....	38
Tabla Nº 2.10. Factor de Ponderación .....	39
Tabla Nº 2.11. Variables para el Cálculo de la Funcionalidad.....	39
Tabla Nº 2.12. Factor de Escala .....	40
Tabla Nº 2.13. Ajustes de la Complejidad .....	41
Tabla Nº 2.14. Ecuación de la Confiabilidad .....	43
Tabla Nº 2.15. Ecuación de la Usabilidad .....	44
Tabla Nº 2.16. Valoración de las Preguntas.....	44
Tabla Nº 2.17. Preguntas para Determinar la Usabilidad.....	45
Tabla Nº 2.18. Ecuación de la Mantenibilidad.....	46
Tabla Nº 2.19. Constantes de Costes .....	50
Tabla Nº 2.20. Coeficientes de COCOMO .....	51
Tabla Nº 2.21. Detalle de Coeficiente de COCOMO II .....	52
Tabla Nº 2.22. Ecuaciones del Método COCOMO II.....	53
Tabla Nº 2.23. Atributos de Coste (FAE).....	54
Tabla Nº 3.24. Tareas para la Obtención de Requisitos .....	58
Tabla Nº 3.25. Descripción de Actores.....	59
Tabla Nº 3.26. Categoría de las Funciones.....	61
Tabla Nº 3.27. Requerimientos funcionales .....	62

Tabla N° 3.28. Requerimientos No Funcionales .....	63
Tabla N° 3.29. Descripción de Usuario y Administrador .....	65
Tabla N° 3.30. Caso de Uso: Administración de Usuario.....	67
Tabla N° 3.31. Caso de Uso: Asistente.....	69
Tabla N° 3.32. Caso de Uso: Enfermera.....	70
Tabla N° 3.33. Caso de Uso: Antecedentes Patológicos .....	72
Tabla N° 3.34. Caso de Uso: Revisión Odontológica.....	73
Tabla N° 3.35. Caso de Uso: Revisión Odontológica Dental.....	74
Tabla N° 3.36. Caso de Uso: Diagnostico.....	74
Tabla N° 3.37. Caso de Uso: Reporte de Historial Clínico .....	75
Tabla N° 3.38. Caso de Uso: Administración de Personal y Paciente .....	76
Tabla N° 3.39. Caso de Uso: Administración de Usuarios Director.....	77
Tabla N° 3.40. Caso de Uso: Reportes .....	78
Tabla N° 3. 41. Autenticación para Ingresar al Sistema .....	80
Tabla N° 3.42. Características de la Funcionalidad .....	91
Tabla N° 3.43. Parámetros de Medición.....	92
Tabla N° 3.44. Punto de Función (Factores de Ponderación) .....	93
Tabla N° 3.45. Valores de Ajuste de la complejidad .....	94
Tabla N° 3.46. Escala de Valoración de Preguntas .....	98
Tabla N° 3.47. Preguntas para Determinar la Usabilidad.....	98
Tabla N° 3.48. Valores para Determinar la Mantenibilidad.....	100
Tabla N° 3.49. Coeficiente del Modelo de COCOMO II.....	102
Tabla N° 3.50. Ecuaciones del Modelo COCOMO II.....	102
Tabla N° 3.51. Cálculo de los Atributos FAE.....	103
Tabla N° 3.52. Gestión de Comunicaciones y Operaciones.....	105

## ÍNDICE DE FIGURAS

Figura N° 2.1. Gráfica General de un Sistema .....	13
Figura N° 2.2. Estructura de desarrollo de software.....	20
Figura N° 2.3. Diagrama de Caso de Uso .....	23
Figura N° 2.4. Diagrama de Estados.....	24
Figura N° 2.5. Diagrama de Secuencias .....	27
Figura N° 2.6. Diagrama de Colaboración.....	29
Figura N° 2.7. Ramificación de Diagrama de Actividad.....	31
Figura N° 2.8. Sincronización de un Diagrama de Actividad .....	31
Figura N° 2.9. Diagrama de Distribución.....	34
Figura N° 2.10. Requisitos de la norma ISO 25000 .....	36
Figura N° 2.11. Estructura ISO 27002.....	57
Figura N° 3.12. Descripción de Usuarios .....	64
Figura N° 3.13. Diagrama de Caso de Uso: GENERAL DEL SISTEMA .....	66
Figura N° 3.14. Diagrama de Caso de Uso: ADMINISTRADOR .....	67
Figura N° 3.15. Diagrama de Caso de Uso: ASISTENTE .....	68
Figura N° 3.16. Diagrama de Caso de Uso: ENFERMERA.....	70
Figura N° 3.17. Diagrama de Caso de Uso: DOCTOR .....	72
Figura N° 3.18. Diagrama de Caso de Uso: DIRECTOR .....	76
Figura N° 3.19. Diagrama de Clases.....	79

## **1. MARCO PRELIMINAR**

### **1.1. INTRODUCCIÓN**

En muchos países se da significativa importancia a los Sistemas de Información, como una herramienta estratégica en la sanidad, ya que las tecnologías de la información como un medio, permiten a los centros de salud o hospitales, nuevas posibilidades de actualización, mediante la sistematización de las tareas, recogida de información de calidad y reingeniería de los procesos, que contribuyen a aumentar la calidad del sistema.

El Centro de Salud “Villa Cooperativa”, se inició el año 1999, y desde entonces se producen problemas referentes a la calidad de atención de los pacientes, provocadas por una insuficiente asignación de recursos humanos y equipamiento. Uno de los problemas principales es el manejo manual de la información, ya que con el transcurso del tiempo la población de pacientes fue aumentando provocando la existencia de problemas en el manejo de la Información clínica. De hecho, es muy importante para la institución absolver estos problemas, puesto que uno de los objetivos que persigue el Centro de Salud, es mejorar la calidad de atención al paciente utilizando los medios tecnológicos posibles.

Es por esta razón, que el presente trabajo de investigación, tiene por propósito desarrollar un Sistema de Gestión de Información de fichas médicas e historial clínico en la especialidad de odontología, utilizando la metodología RUP, y las herramientas de desarrollo: Apache, PHP,HTML, Css, JavaScript, Ajax, Bootstrap, CodeIgniter, jQuery y MySQL.

Se pretende con el Sistema de Gestión de Información alcanzar los objetivos planteados, lo cual permite disminuir los problemas frecuentes que se presenta, como duplicaciones, pérdida, deterioro de la información de los pacientes y problemas con el espacio físico y de, esta forma, mejorar la calidad de atención brindada al paciente en el Centro de Salud “Villa Cooperativa”.

## **1.2. ANTECEDENTES**

### **1.2.1. Antecedentes Institucionales**

El “Centro de Salud Villa Cooperativa” fue fundado el año 1999, ubicado en la Ciudad de El Alto, Zona Mariscal Sucre “C”. Las áreas de atención médica que se ofrecen a la población son las siguientes:

- ✓ Pediatría
- ✓ Ginecología
- ✓ Odontología
- ✓ Farmacia

El Centro de Salud actualmente cuenta con 1631 historiales clínicos. Esto implica una alta demanda del servicio de salud, lo cual provoca situaciones como las siguientes: sobrecarga de trabajo del personal médico, prolongada espera de pacientes para ser atendidos y extensión del periodo de recuperación de los pacientes.

### **1.2.2. Antecedentes Académicos:**

- ✓ [Julián Parada Niño; 2008] “Análisis, Diseño e Implementación de Historias Clínicas de Régimen SUBSIADO”, cuyo propósito consiste en Diseñar e Implementar un aplicativo que supla las necesidades informáticas del hospital San José, para así llevar un control de sistema para gestionar los servicios médicos de la institución. Entre las herramientas para el desarrollo de este sistema se utiliza: NetBeans, lenguaje de programación Java y gestor de base de datos MySQL. Bogota – Corporación Universitaria Minuta de Dios.
- ✓ [Pablo Calderón Vilchez; 2012] “Sistema de Atención Medica Odontológica”, tuvo por finalidad elaborar una aplicación software que permita la gestión especializada de la historia clínica Estomatológica de los pacientes de un

centro Médico. Este proyecto se desarrolla mediante la metodología de investigación GRAPPE (Guía para la ingeniería de aplicaciones de aplicaciones rápidas) y para diagramas UML (lenguaje unificado de Modelado). Lima, Perú - Universidad Peruana de Ciencias Aplicadas.

- ✓ [Sara Patricia Huanca; 2015] “Sistema Web de Control de Pagos, Citas e Historiales Clínicos”, su objetivo se refiere a desarrollar un Sistema Web de control de pagos, citas e historiales clínicos para la clínica dental LAVADENT, que permita tener un apropiado manejo de información odontológica en constante actualización, y así garantizar la preferencia de antiguos y nuevos pacientes. La metodología utilizada para el desarrollo de este proyecto fue: SCRUM y UWE, que le permite identificar las necesidades del sistema Web. La Paz-Universidad Mayor de San Andrés.
  
- ✓ [Elizabeth Villa Vargas; 2012] “Sistema Integrado de Seguimiento de Historias Clínicas y tareas Administrativas en una Clínica Odontológica”, tuvo por propósito diseñar e implementar un sistema informático que lleve el control y seguimiento de los pacientes en la clínica dental “ODAS”, control del personal que trabaja en la clínica y el control de stock de los materiales médicos”. Entre las herramientas para el desarrollo de este sistema que se utilizan se tienen: Lenguaje de programación PHP en HTML y gestor de base de datos MySQL. La Paz – Universidad Mayor de San Andrés.

Los proyectos mencionados anteriormente se basan principalmente en historiales clínicos generales y tareas administrativas. También utilizan Lenguajes de Programación Puro. Por lo tanto el presente proyecto se diferencia porque se centra en historiales clínicos que esté orientada principalmente en la especialidad de odontología y la generación de fichas médicas y asignación de turnos, de tal manera que coadyuve a mejorar la calidad de atención a los pacientes. También

se utiliza la herramienta de framework Codeigniter, para desarrollar un software de calidad.

### **1.3. PLANTEAMIENTO DEL PROBLEMA**

#### **1.3.1. Problema Principal**

Debido al manejo manual de grandes volúmenes de información de los historiales clínicos y el procedimiento usado en la asignación de fichas médicas, se origina demoras en la atención a los pacientes y la duplicidad de historiales clínicos, repercutiendo en un inadecuado control y seguimiento de los tratamientos de pacientes del Centro de Salud “Villa Cooperativa”.

#### **1.3.2. Problemas Secundarios**

- ✓ La demora en la búsqueda del historial clínico, ocasiona una pérdida de tiempo significativo al momento de atención al paciente.
- ✓ El hecho de que no exista un control en la asignación de fichas médicas, origina que no se realice un seguimiento adecuado de los tratamientos.
- ✓ Las limitaciones de control del historial clínico ocasiona pérdida de información de los pacientes.
- ✓ Al momento de tomar decisiones se detecta dificultades de acceso oportuno a la información requerida de los pacientes.

¿Cómo mejorar los procedimientos de atención a los pacientes con el Sistema de Control de información para la asignación de fichas médicas, seguimiento del historial clínico y obtención de información oportuna en el centro de salud de “Villa Cooperativa” de la ciudad de El Alto?

## **1.4. OBJETIVOS**

### **1.4.1. Objetivo General**

Desarrollar un Sistema de Gestión de información de Fichas Médicas e Historial Clínico en la Especialidad de Odontología, que coadyuve a un adecuado control y seguimiento de los tratamientos de los pacientes del Centro de Salud de “Villa Cooperativa” de la ciudad de El Alto, coadyuvando a mejorar la calidad de atención.

### **1.4.2. Objetivos Específicos**

- ✓ Recolectar y analizar el flujo de información, para identificar los requerimientos del Centro de Salud en la especialidad de odontología.
- ✓ Reducir la demora en la búsqueda del historial clínico y facilitar el proceso de atención a los pacientes.
- ✓ Actualizar las fichas médicas e historial clínico de manera oportuna y eficiente, mejorando el control y seguimiento del historial clínico.
- ✓ Proporcionar un módulo de reportes para acceder a la información verídica, para una adecuada toma de decisiones.

## **1.5. JUSTIFICACIÓN**

### **1.5.1. Técnica**

El Centro de Salud “Villa Cooperativa”, actualmente no cuenta con el hardware y software necesario. Sin embargo, el Sistema de Gestión de Información a desarrollar necesita un nivel de tecnología totalmente accesible a los recursos de la organización, por lo cual es viable su aplicación.

### **1.5.2. Económica**

El Centro de Salud se beneficiará con el Sistema de Gestión de Información, puesto que se utiliza software de acceso libre. A través de este recurso informático se automatizará los procesos de Centro de Salud, se reducirá el tiempo de búsqueda de historiales clínicos y lo más importante, incrementará la confiabilidad de los pacientes, lo cual significa mayor posibilidad de ganancia para el Centro de Salud.

### **1.5.3. Social**

El sistema de Gestión de Información proporcionará un mejor desempeño en el Centro de Salud, lo cual beneficiará tanto al personal del centro de salud como a la sociedad, el mismo permitirá búsquedas de información rápida y oportuna al momento que se requiera información de los pacientes, de manera que se atenderá a más pacientes de lo que atendía anteriormente, obteniendo mejores resultados y mayores ventajas competitivas al centro de salud.

## **1.6. METODOLOGÍA**

### **1.6.1. Metodología RUP (Rational Unified Process)**

Es una secuencia de pasos necesarios para el desarrollo y mantenimiento de gran cantidad de sistemas, en diferentes áreas de aplicación, distintas organizaciones, diversos medios de competencia y en proyectos de tamaños variables (desde el más básico al más complejo). Actualmente es propiedad de International Business Machines (IBM) y está basado en un enfoque disciplinado de asignación de tareas y responsabilidades dentro de una organización de desarrollo con la finalidad de asegurar la obtención de un software de alta calidad que satisfagan la necesidad de los usuarios finales dentro de un calendario y tiempo predecible.(Pérez, 2011).

### **1.6.2. UML (Lenguaje Modelado Unificado)**

Es una herramienta que permite a los creadores de sistemas generar diseños que capturen sus ideas en una forma fácil de comprender, para comunicar a otras personas. UML trabaja con los siguientes diagramas:

- ✓ Diagrama de casos de uso
- ✓ Diagrama de secuencias
- ✓ Diagrama de estados
- ✓ Diagrama de actividades
- ✓ Diagrama de colaboración
- ✓ Diagrama de clases

Los diagramas mencionadas son los más utilizados para desarrollar un software. (Gesvin Romero, 2004).

## 1.7. HERRAMIENTAS

- ✓ **XAMPP:** Es un servidor independiente multiplataforma, de software libre, que integra en una sola aplicación, un servidor web apache, interpretador de lenguaje de scripts PHP, un servidor de base de datos MySQL, un servidor de FTP. Fácil de instalar y poner en marcha en cualquier sistema operativo, Xampp es una herramienta que trabaja de forma local, que permite probar trabajos en el propio ordenador sin la necesidad de tener que acceder a internet. (Prendice Hall, 2002).
  
- ✓ **APACHE:** Es un servidor web de software libre, estable, eficiente, extensible y multiplataforma. Apache no es un servidor físico, sino es un servidor web que se ejecuta en el servidor. Apache establece una conexión entre un servidor y los clientes a través de los navegadores, es multiplataforma, por lo cual trabaja con: Linux, UNIX, Windows y otros sistemas operativos. Apache es personalizado, porque tiene módulos de seguridad, almacenamiento en cache, reescrituras de URL y autenticación de contraseña. Se ajusta a las propias configuraciones del servidor a través de un archivo llamado .htaccess, que es un archivo de configuración de apache compatible con todos los planes de Hostnger. (Prendice Hall, 2002).
  
- ✓ **MYSQL:** Es un gestor de base de datos que es muy rápido, confiable, robusto y fácil de usar tanto en volúmenes de datos grandes como pequeños. 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. (Ian Gilfillan, 2003).

- ✓ **PHP:** Es un lenguaje de código abierto interpretado, de alto nivel, embebido en páginas HTML y ejecutado en el servidor, este lenguaje de programación es empleado normalmente para la creación de páginas web dinámicas. PHP puede ser utilizado en cualquiera de los principales sistemas operativos del mercado, incluyendo Linux, Microsoft Windows. PHP soporta la mayoría de los servidores Web, entre los que se encuentra Apache. Las características de PHP son velocidad, estabilidad, seguridad y simplicidad. (Juan Pavón, 2012).
  
- ✓ **JQUERY:** Es una biblioteca de JavaScript que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web. (Isaac Ojeda, 2012).
  
- ✓ **JAVASCRIPT:** Es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativos, débilmente tipado y dinámico. Se utiliza principalmente en su forma del lado del cliente, implementando como parte de un navegador web permitiendo mejorar en la interfaz de usuario y pagina web dinámico. (Prendice Hall, 2002).
  
- ✓ **CSS:** Es el lenguaje de estilo que define la presentación de los documentos HTML. Con Css se trabaja con fuentes, colores, tamaño, imágenes de fondo, maquetación, crear animaciones y otros efectos. (Juan Gauchat, 2012).
  
- ✓ **CODEIGNITER:** Es un Framework para el desarrollo de aplicaciones en PHP, que utiliza MVC (Modelo Vista Controlador), CodeIgniter es extremadamente liviano, rápido, genera URLs limpios y es extensible. Proporciona un amplio conjunto de bibliotecas para tareas comunes, así como una interfaz simple y una estructura lógica para acceder a esas bibliotecas, CodeIgniter permite centrarse creativamente en un proyecto al minimizar la cantidad de código necesario. (Barcelona, 2018).

- ✓ **AJAX:** Las siglas de Asynchronous JavaScript And XML (Javascript asíncrono y XML). No es en sí un lenguaje de programación, sino una nueva técnica que combina varios lenguajes de programación. Ajax es asíncrono, es decir consiste en que, la página sigue ejecutando acciones, mientras se hace la petición al servidor sin la necesidad de refrescar la página. (Juan Fuentes, 2009).
- ✓ **BOOTSTRAP:** Permite crear interfaces web con Css y JavaScript, cuya particularidad es que se logra adaptar la interfaz del sitio web al tamaño del dispositivo en que se visualice. Es decir, se adapta automáticamente al tamaño de un PC, una Tablet u otro dispositivo. (Miguel Muñoz, 2018).

Las herramientas que se utilizaran para el desarrollo y la implementación del Sistema se muestran en la Tabla N° 1.1

Donde se describe servidor, el gestor de la base de datos, lenguaje de programación, herramientas de diseño y framework.

**Tabla N° 1.1. Herramienta**

SERVIDOR	GESTOR DE BASE DE DATOS	LENGUAJE DE PROGRAMACIÓN	HERRAMIENTA DE DISEÑO	FRAMEWORK
Apache	MySQL	Php	CSS HTML	Bootstrap CodeIgniter JQuery

**Fuente:** Elaboración Propia

## **1.8. LÍMITES Y ALCANCES**

### **1.8.1. Limites**

El sistema de Control de Información no toma en cuenta las áreas de Pediatría, Ginecología, Farmacia y Activos, ya que estas están fuera del alcance del proyecto.

El sistema está destinado exclusivamente para generar fichas médicas e historiales clínicos en la especialidad odontológica.

### **1.8.1. Alcances**

El proyecto se orienta al análisis, diseño e implementación del sistema de control de información que involucra lo siguiente:

- ✓ Registro de datos del personal a cargo (médicos).
- ✓ Registro de las fichas medicas e historial clínico de los pacientes.
- ✓ Asignación de turnos a las distintas especialidades.
- ✓ Registro de nuevos pacientes (altas, modificaciones).
- ✓ Búsqueda de historiales clínicos de pacientes ya registrados.
- ✓ Realizar consultas. (Permite manejar toda la información relacionadas con el paciente).
- ✓ Desarrollo de medidas para corregir historiales duplicados.
- ✓ Desarrollo de medidas de seguridad de los historiales clínicos.

## **1.9. APORTES:**

El aporte del presente proyecto es un Sistema de Gestión de Información de Fichas Medicas e Historiales Clínicos en la Especialidad Odontológica que brinda

una herramienta para facilitar y agilizar los procesos de seguimiento evaluación y toma de decisiones sobre los pacientes con alguna alergia a inyección u otros tratamientos médicos.

## 2. MARCO TEÓRICO

En este capítulo se presentan todas las bases teóricas para el desarrollo del presente proyecto.

### 2.1. SISTEMA

Un sistema es un módulo ordenado de elementos que se encuentran interrelacionados y que interactúan entre sí. El concepto se utiliza tanto para definir a un conjunto de concepto como a objetos reales dotados de organización, también puede mencionarse la noción de sistema informático, muy común en las sociedades modernas. A este tipo de sistemas, se le denominan como el conjunto de hardware, software y soporte humano que forma parte de una empresa u organización. Incluye ordenadores con los programas necesarios para procesar datos y las personas encargadas de su manejo.

(Julián Pérez Porto, 2008).

En la figura N° 2.1. Se muestra las partes que componen un sistema.

**Figura N° 2.1. Gráfica General de un Sistema**



**Fuente:** (Julián Pérez Porto, 2008)

Los sistemas reciben (entradas) datos, energía o materia del ambiente y proveen (salidas) como información. Disponible en: (Leandro Alegsa, Alegsa, 2017)

Por tanto, se menciona que un Sistema es un conjunto de partes o elementos organizados y relacionados que interactúan entre sí para lograr un objetivo. Esta definición, es aplicable a todos los tipos de sistemas de información. (Kendal, 1997).

## **2.2. GESTIÓN**

Gestión es la acción y el efecto de gestionar y administrar recursos de la organización, para alcanzar los objetivos propuestos por la misma.

Referencia a ordenar por lo cual en general, es la acción que desarrolla actividades productivas con el fin de generar rendimientos de los factores que intervienen con “Efectos de administrar”. (Española, 2001).

## **2.3. INFORMACIÓN**

La información se puede definir como un conjunto de datos procesados y que tienen un significado (relevante, propósito y contexto), y por lo tanto son de utilidad para la toma de decisiones.

Proceso que se encarga de suministrar los recursos necesarios para la toma de decisiones, permitiendo mejorar los procesos, productos y servicios de la organización. Realizar procesos de gestión permite identificar, organizar, representar y recuperar información dispersa en áreas. En este sentido se menciona que “La gestión de la información se ocupa de los resultados finales, no solo citas y localizaciones”.

La gestión de información implica:

- ✓ Determinar la información que se precisa
- ✓ Recoger y analizar información

- ✓ Registrarla y recuperarla cuando sea necesaria
- ✓ Utilizarla y divulgarla.

Disponible en: (Novoa V. D., 2017)

## **2.4. GESTIÓN DE INFORMACIÓN**

La gestión de información es todo lo que tiene que ver con obtener la información correcta, en la forma adecuada, para la persona indicada, al costo correcto, en el momento oportuno, en el lugar indicado para tomar la acción precisa. (Woodman, 1985).

## **2.5. SISTEMA DE GESTIÓN DE INFORMACIÓN**

“Sistema integrado y automatizado para proveer la información que sostenga las funciones de operatividad, gestión y toma de decisiones en una organización”. (Davis G. & Olson, 1985)

Por tanto, permite la gestión de los recursos de información tanto internos como externos. Su finalidad es generar servicios y productos que respondan a las necesidades y sobrepasen las expectativas de los usuarios, posibilitando que el sistema trabaje eficientemente y económicamente a la vez. El Sistema de Gestión de Información aprovecha al máximo sus recursos de información en función de la mejora continua y de la toma de decisiones organizacional a todos los niveles jerárquicos, desde la cúspide estratégica hasta la base operativa.

Una vez definido algunos términos sobre sistemas y gestión de información se procede a definir términos más utilizados en el área de odontología.

## 2.6. HISTORIAL CLÍNICO

La palabra historia se refiere a hechos pasados o de los sucesos que le ocurrieron a un individuo en la vida. Lo clínico, en tanto, es aquello vinculado a las prácticas médicas.

Se conoce como historial clínico, por lo tanto, a la recopilación y el almacenamiento de los datos referidos a una persona que son de importancia para la medicina. La historia clínica incluye información sobre las enfermedades que atravesó y los tratamientos que recibió. (Dr. Hugo Duran de Castro, 2010).

## 2.7. ODONTOLOGÍA

La odontología es una especialidad de la medicina que estudia la anatomía y fisiología de los dientes y sus estructuras circundantes en la cavidad oral (labios, boca, paladar, orofaringe, mandíbula, etc.). La odontología también se refiere a la práctica de la reparación y restauración de los dientes y al tratamientos las alteraciones de los dientes y sus estructuras sus de soporte. Se encarga de las caries, endodoncia, ortodoncia, prótesis dentales, etc.

(Julián Pérez Porto, 2009).

### 2.7.1. Términos Odontológicos:

- ✓ **ATM:** La ATM es la articulación que hace posible abrir y cerrar la boca. Es la articulación dónde la mandíbula se articula con el hueso temporal del cráneo, delante del oído y en cada lado de la cabeza. Se utiliza al masticar, hablar, tragar, bostezar, etc. Por ello, es una de las articulaciones más usadas en el cuerpo.
- ✓ **Ganglios:** Los ganglios linfáticos, nódulos linfáticos, nodos linfáticos o linfonodos son unas estructuras ovaladas o reniformes (con forma de riñón), encapsuladas, que forman parte estructuralmente del sistema linfático y

funcionalmente del sistema inmunitario. Se ubican a lo largo del trayecto de los vasos linfáticos formando cadenas o racimos.

- ✓ **Labios:** Los labios, como parte del rostro de determinados animales, son los pliegues que forman el inicio de la boca, en el conjunto de la cabeza y como inicio del aparato digestivo.
- ✓ **Lengua:** La lengua es un hidrostato muscular, un órgano (contiene glándulas salivales) móvil situado en el interior de la boca, impar, medio y simétrico, que desempeña importantes funciones como la hidratación de boca y alimentos mediante la salivación, la deglución, el lenguaje y el sentido del gusto. Deglutir y hablar son acciones prácticamente imposibles sin la saliva.
- ✓ **Paladar:** Constituye la pared superior o techo de la cavidad oral. Está dividido en dos partes, la bóveda palatina o paladar óseo en sus dos tercios anteriores, y el paladar blando o velo del paladar en su tercio posterior. En el centro y en la parte más posterior inferior del velo del paladar cuelga la úvula. El paladar separa la cavidad bucal de las fosas nasales. Es una zona de roce cuya interacción lengua paladar permite articular sonidos.
- ✓ **Piso de la boca:** La región del piso de la boca está irrigada por la arteria lingual, a nivel de lo que era una de sus ramas que era la arteria sublingual, pero también por lo que son las arterias submentonianas y submandibulares de la arteria facial, y también por lo que es la arteria incisiva, que es parte de la arteria alveolar inferior.
- ✓ **Mucosa Yugal:** La mucosa bucal está formada por la mucosa de revestimiento. Se encuentra en la cara interna del labio, cara interna de las mejillas, piso de la boca, cara inferior de la lengua y paladar blando. Estas zonas no participan, directamente, en el fenómeno masticatorio y no tienen receptores del gusto.
- ✓ **Encías:** Es una fibromucosa formada por tejido conectivo denso con una cubierta de epitelio escamoso queratinizado que cubre los procesos

alveolares y rodea a los dientes. La encía es contigua al ligamento periodontal y, en su exterior, con los tejidos mucosos de la cavidad oral. (Zimbrón & Feingold, 1993).

## 2.8. METODOLOGÍA RUP

La metodología RUP es una disciplina que permite mantener un orden debidamente estricto el cual asigna responsabilidades en una organización. RUP es una herramienta determinada por ciclos y fases para el proceso del modelado. La metodología RUP, está conformado por 4 fases las cuales son: fase de inicio, fase de elaboración, fase de construcción y fase de transición.

- ✓ **Fase de inicio:** Es la fase en la cual se comienzan a identificar los actores en la organización y los requerimientos que se necesiten para resolver los problemas que se presenten en la misma. Se identificaran los problemas que tiene la organización y lo que se necesita, luego se determinara el tiempo del proyecto.
- ✓ **Fase de elaboración:** Es la fase en la cual ya se tiene determinados los puntos importantes para comenzar a elaborar el proyecto lo más importante es tener claramente la visión de lo que se va a realizar, también es importante definir los riesgos que puedan ocasionar algunos elementos en el desarrollo del proyecto, y se dan las observaciones respectivas por el cliente o especialista. La fase de elaboración resuelve los siguientes criterios:
  - ❖ Se desarrollan un modelo de casos de uso en el cual ya se han identificado los casos de uso. El modelo de casos de uso debe estar en un 80% completo.
  - ❖ Una descripción de la arquitectura del software en un proceso del desarrollo del software.
  - ❖ Un plan de desarrollo para el proyecto total.

Si el proyecto no puede pasar esta fase, debe ser reajustado, mediante las iteraciones que sean necesarias.

- ✓ **Fase de construcción:** Es la fase en la que la mayoría de decisiones son aceptadas. Aquí es donde el proyecto está en ejecución en una plataforma definida. Todas las componentes restantes se desarrollan e incorporan al producto. Aquí se crea una guía para el usuario con la que el usuario podrá tener conocimiento del sistema al utilizarlo.
- ✓ **Fase de transición:** En esta fase se tiene el producto listo para ser instalado en la empresa, en todo caso distribuirlo a donde requieran el producto. Luego se sugieren las aprobaciones si es que el usuario se adecua al sistema. Finalmente se espera la aprobación del sistema y su aceptación en la organización.

(Pérez, 2011).

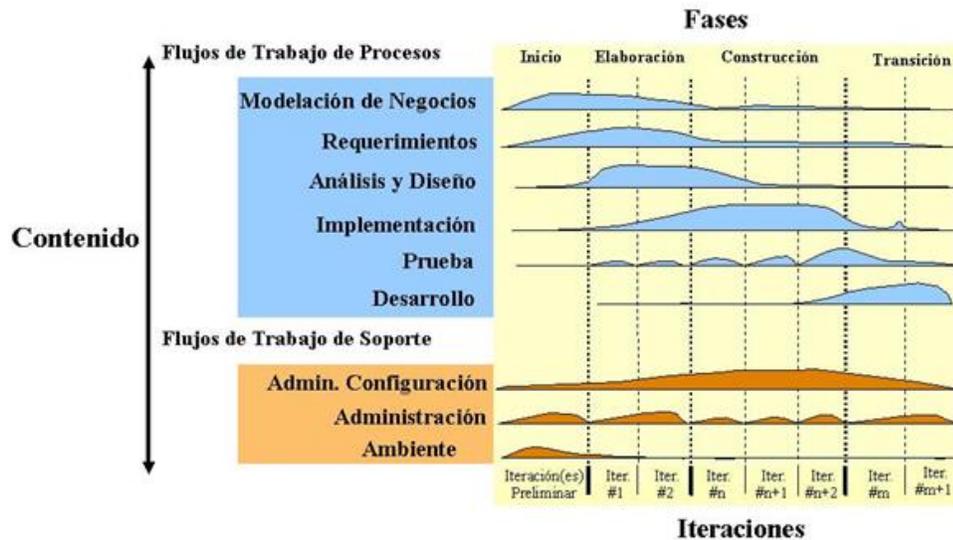
## **ESTRUCTURA DEL PROCESO UNIFICADO DE DESARROLLO**

El proceso unificado de desarrollo de software está compuesto por 4 fases, cada una de estas fases es desarrollada mediante el ciclo de iteraciones, la cual consiste en reproducir el ciclo de vida en cascada a menor escala.

El eje horizontal representa la parte dinámica del proceso, el tiempo, iteraciones y las metas, el eje vertical representa la parte estática del proceso donde se describe los flujos de trabajo, requisitos, análisis, diseño, implementación y prueba. Las curvas son una aproximación hasta donde se llevan a cabo los flujos de trabajo en cada fase, como se puede ver en la Figura 2.2

(Jacobson, 2000).

**Figura Nº 2.2. Estructura de desarrollo de software**



**Fuente:** (Pérez metodología RUP, 2011).

## 2.9. UML (LENGUAJE MODELADO UNIFICADO)

UML, es el lenguaje de modelado de sistemas de software más conocido y utilizado en la actualidad; está respaldado por el OMG (Object Management Group). Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. UML es estándar para describir un "plano" del sistema (modelo), incluyendo aspectos conceptuales tales como procesos de negocio, funciones del sistema, y aspectos concretos como expresiones de lenguajes de programación, esquemas de bases de datos.

Es importante remarcar que UML es un "lenguaje de modelado" para especificar o para describir métodos o procesos. Se utiliza para definir un sistema, para detallar los artefactos en el sistema y para documentar y construir. En otras palabras, es el lenguaje en el que está descrito el modelo.

Se puede aplicar en el desarrollo de software gran variedad de formas para dar soporte a una metodología de desarrollo de software (tal como el Proceso Unificado Racional o RUP), pero no especifica en sí mismo qué metodología o proceso usar.

UML no puede compararse con la programación estructurada, pues UML significa Lenguaje Unificado de Modelado, no es programación, solo se diagrama la realidad de una utilización en un requerimiento. Mientras que, programación estructurada, es una forma de programar como lo es la orientación a objetos, la programación orientada a objetos viene siendo un complemento perfecto de UML, pero no por eso se toma UML sólo para lenguajes orientados a objetos.

UML cuenta con varios tipos de diagramas, los cuales muestran diferentes aspectos de las entidades representadas.

(Larman, 1999).

El UML fue creado por Grady Booch, James Rumbaugh e Ivar Jacobson a mediados de los años noventa, el lenguaje compuesto por diversos elementos gráficos se combinan para formar diagramas, debido a que es un lenguaje, cuenta con reglas para combinar tales elementos.

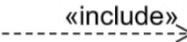
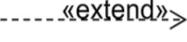
La finalidad de los diagramas es presentar diversas perspectivas de un sistema, a las cuales se les conoce como modelo. El modelo UML de un sistema es similar a un modelo a escala de un edificio junto con la interpretación del artista del edificio. Es importante destacar que un modelo UML describe lo que supuestamente hará un sistema, pero no dice cómo implementar dicho sistema.

### **2.9.1. Diagramas de Caso de Uso**

Un caso de uso es la descripción de las acciones del sistema desde el punto de vista del usuario, forman parte de la fase del análisis, es una herramienta para obtener requerimientos del sistema. (Garcia & Holgado, 2018).

En la tabla N° 2.2 se describe los iconos que se utiliza para desarrollar un diagrama de casos de uso.

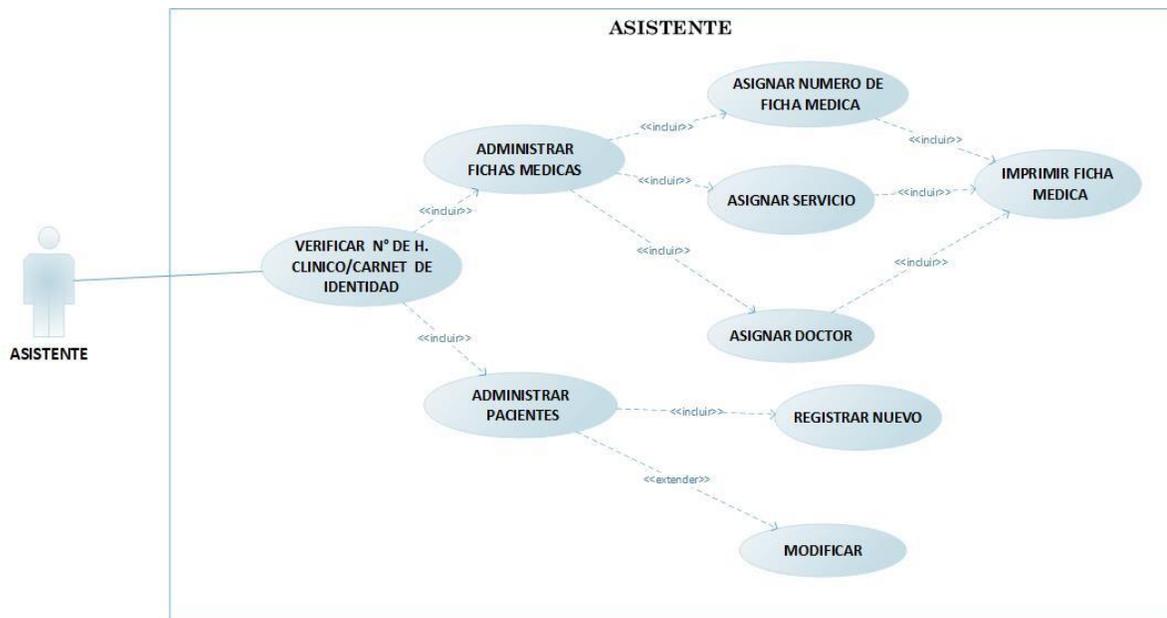
**Tabla N° 2.2. Estereotipos del Diagrama de Casos de Uso**

ICONO	NOMBRE	DESCRIPCIÓN
	Actor	Entidades externas que interactúan con el sistema.
	Caso de uso	Función del sistema.
	Asociación	Línea de comunicación entre un actor y un caso de uso en el que participa
	Generalización	Una relación entre un caso de uso general y un caso de uso más específico, que hereda y añade propiedades al caso de uso base
	Inclusión	Inserción de comportamiento adicional en un caso de uso base, que describe explícitamente la inserción
	Extensión	Inserción de comportamiento adicional en un caso de uso base que no tiene conocimiento sobre él
	Realización	Establece una relación entre el caso de uso y los diagramas que describen la funcionalidad del caso de uso

**Fuente:** (Garcia & Holgado, UML.Unified Modeling Language, 2018)

En la Figura N° 2.3, se muestra un ejemplo del diagrama del caso de uso, se describe como se realiza la generación de fichas médicas.

**Figura N° 2.3. Diagrama de Caso de Uso**



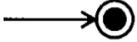
**Fuente:** Elaboración Propia

### 2.9.2. Diagrama de Estados

Permite un cambio en un sistema es decir que los objetos que lo componen modificaron su estado como respuesta a los sucesos y al tiempo. (Joseph, 2000).

En la tabla N° 2.3. Se muestra la descripción de los iconos que se utiliza para diagramar el diagrama de estados.

**Tabla Nº 2.3. Estereotipos del Diagrama de Estado**

ICONO	DESCRIPCIÓN	ICONO	DESCRIPCIÓN
	Estado		Estado Inicial
	Transición		Estado Final

**Fuente:** (Garcia & Holgado, UML.Unified Modeling Language, 2018)

- ✓ **Estado:** Es la representación de la situación durante la vida de un objeto. Se representa con un rectángulo que tiene sus esquinas.
  - ✓ **Transición:** es una flecha que representa el pasaje entre diferentes estados de un objeto. Se etiqueta con el evento que lo provoca y con la acción resultante.
- En la figura Nº 2.4, se observa la diagramación de estado, por ejemplo el estado de reserva de ficha médica.

**Figura Nº 2.4. Diagrama de Estados**



**Fuente:** Elaboración propia

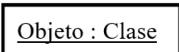
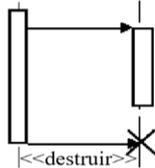
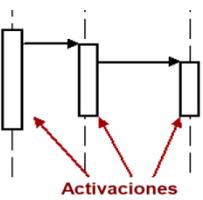
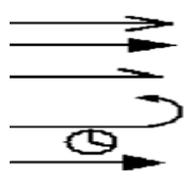
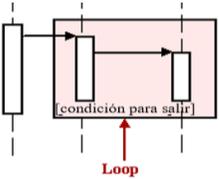
### 2.9.3. Diagrama de Secuencias

El diagrama de secuencias muestra la forma en que los objetos se comunican entre sí al transcurrir el tiempo.

Un diagrama de secuencia muestra una iteración ordenada según la secuencia temporal de eventos y el intercambio de mensajes. Ponen especial énfasis en el orden y el momento en el que se envían los mensajes a los objetos.

En la tabla Nº 2.4. Se describe los iconos que se utilizan para diagramar un diagrama de Secuencias.

**Tabla Nº 2.4. Estereotipos del Diagrama de Secuencias**

ICONO	DESCRIPCIÓN	ICONO	DESCRIPCIÓN
	Rol de clases		<b>Dstrucción de objeto.</b>
	Activaciones		Tipos de mensaje <ul style="list-style-type: none"> <li>✓ Simple</li> <li>✓ Sincrónico</li> <li>✓ Asincrónico</li> <li>✓ Rechazado</li> <li>✓ Time out</li> </ul>
	Líneas de vida		Loops

**Fuente:** (Joseph, aprendiendo UML en 24 horas, 2000)

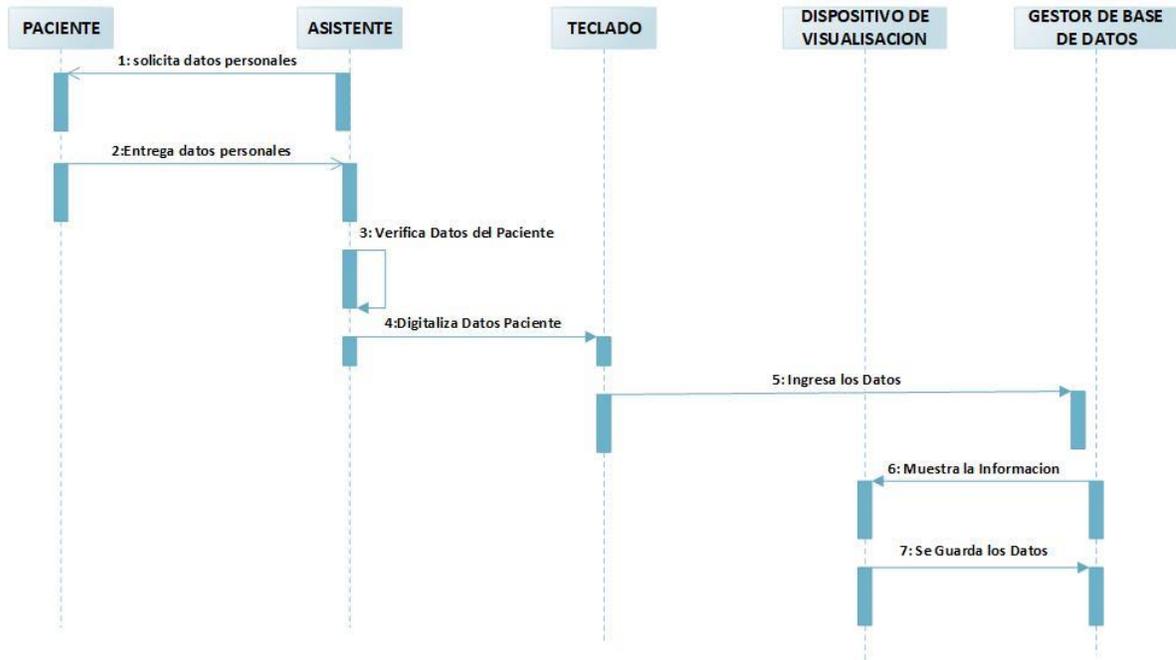
- ✓ **Rol de la clase:** El rol de la clase describe la manera en que un objeto se va a comportar en el contexto.
- ✓ **Activación:** Los cuadros de activación representan el tiempo que un objeto necesita para completar una tarea.
- ✓ **Mensajes:** Los mensajes son flechas que representan comunicaciones entre objetos.
- ✓ **Línea de vida:** Las líneas de vida son verticales y en línea de puntos, ellas indican la presencia del objeto durante el tiempo.
- ✓ **Destrucción de objetos:** Los objetos pueden ser eliminados temporalmente usando una flecha etiqueta “<<destruir>>” que apunta a una x.
- ✓ **Loops:** Es una repetición, en que se debe colocar una condición para abandonar el Loop.

**Fuente:** (Garcia & Holgado, UML.Unified Modeling Language, 2018)

En la figura N° 2.5, se observa la forma como interactúan los objetos en sí. Por ejemplo el registro de un paciente en el centro de salud. Por lo tanto la secuencia seria:

- ✓ Asistente solicita datos personales del paciente
- ✓ Paciente entrega datos personales
- ✓ Asistente verifica los datos
- ✓ Asistente digitaliza los datos del paciente
- ✓ Ingresa los datos a la Base de Datos
- ✓ Se muestra la información en dispositivo de visualización
- ✓ Se guarda los datos.

**Figura N° 2.5. Diagrama de Secuencias**



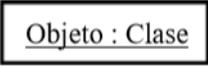
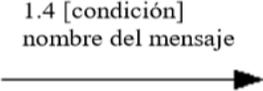
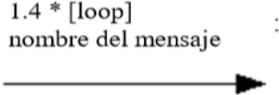
**Fuente:** Elaboración propia

#### **2.9.4. Diagrama de Colaboraciones**

Un diagrama de colaboraciones muestra a los objetos como tales y sus relaciones entre sí. Además de las relaciones entre objetos, el diagrama de colaboración muestra los mensajes que se envían los objetos, evitando la multiplicidad que podría ser fuente de confusión.

En la tabla N° 2.5, se muestra los iconos para diagramar los diagramas de colaboración.

**Tabla N° 2. 5. Estereotipos del Diagrama de Colaboración**

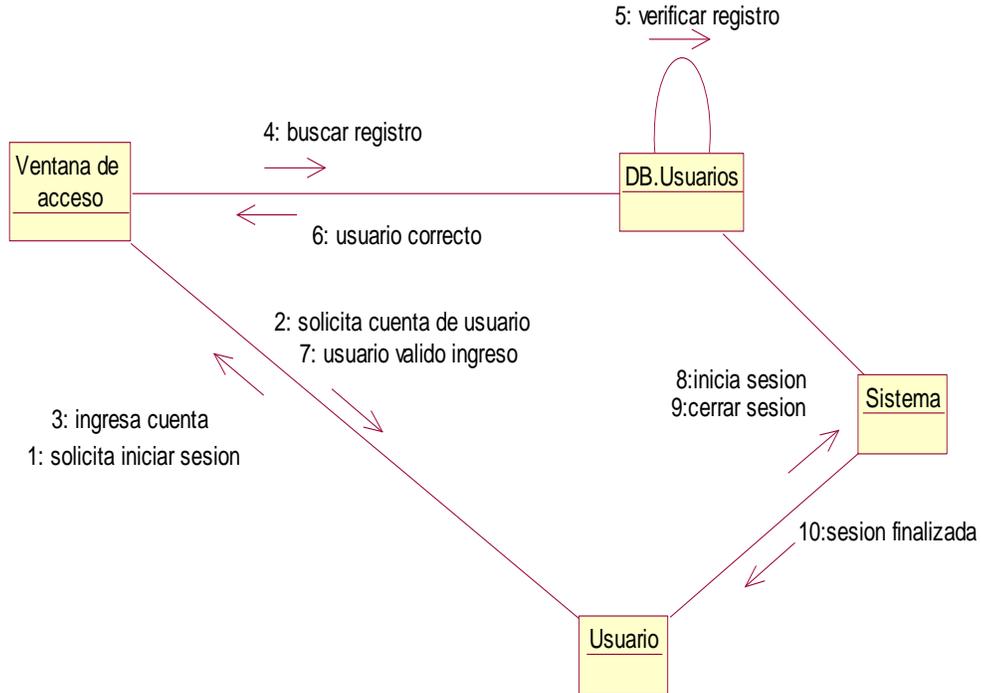
ICONO	DESCRIPCIÓN	ICONO	DESCRIPCIÓN
	Rol de la clase		Mensajes
	Rol de las Asociaciones		

**Fuente:** (Garcia & Holgado, UML.Unified Modeling Language, 2018)

- ✓ **Rol de clases:** Describe cómo se va a comporta un objeto. Los atributos del objeto no se listan.
- ✓ **Rol de las Asociaciones:** Describen como se va a comportar una asociación en una situación particular. Se usa línea simple.
- ✓ **Mensajes:** Contrariamente a los diagramas de secuencias, los diagramas de colaboración no tienen una manera explícita para denotar el tiempo, por lo que entonces numeran a los mensajes en orden de ejecución.  
(Garcia & Holgado, 2018).

En la figura N° 2.6, se muestra un ejemplo del proceso de acceso al sistema, también se observa la forma de diagramación de un Diagrama de Colaboración.

**Figura N° 2.6. Diagrama de Colaboración**



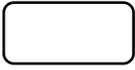
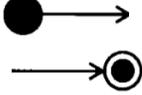
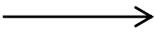
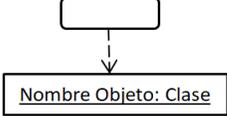
**Fuente:** (Joseph, aprendiendo UML en 24 horas, 2000)

### 2.9.5. Diagramas de Actividades

Un diagrama de actividades ha sido diseñado para mostrar una visión simplificada de lo que ocurre durante una operación o proceso. Es una extensión de un diagrama de estados. El diagrama de actividades resalta, precisamente a las actividades. (Garcia & Holgado, 2018).

En la tabla N° 2.6, se muestra los iconos para diagramar los Diagramas de Actividad.

**Tabla N° 2.6. Estereotipos del Diagrama de Actividad**

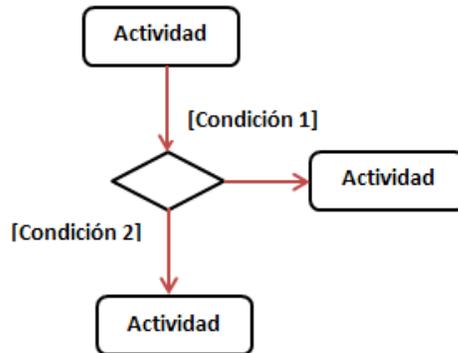
ICONO	DESCRIPCIÓN	ICONO	DESCRIPCIÓN
	Estado de Acción		Estado Inicial Estado Final
	<b>Flujo de acción</b>		<b>Flujo de objeto</b>

**Fuente:** (Garcia & Holgado, UML.Unified Modeling Language, 2018)

- ✓ **Estado de Acción:** Representan las acciones no interrumpidas de los objetos.
- ✓ **Flujo de acción:** Representado por una flecha, ilustra las relaciones entre los estados de acción.
- ✓ **Flujo de objeto:** Es la creación y modificación de objetos por parte de actividades. Una flecha de flujo de objeto, desde una acción está creando o influyendo sobre dicho objeto. Una flecha de flujo de objeto, desde un objeto a una acción, indica que el estado de acción utiliza dicho objeto.

En la figura N° 2.7, se observa la ramificación, en donde un rombo representa una decisión con caminos alternativos. Las salidas alternativas deben estar etiquetadas con una condición.

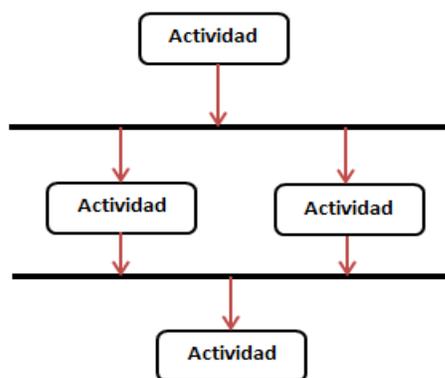
**Figura Nº 2.7. Ramificación de Diagrama de Actividad**



**Fuente:** (Joseph, aprendiendo UML en 24 horas, 2000)

En la figura Nº 2.8, se observa la sincronización, donde una barra de sincronización ayuda a ilustrar la ocurrencia de transiciones paralelas, así quedan representadas las acciones concurrentes.

**Figura Nº 2. 8. Sincronización de un Diagrama de Actividad**



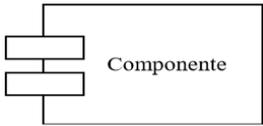
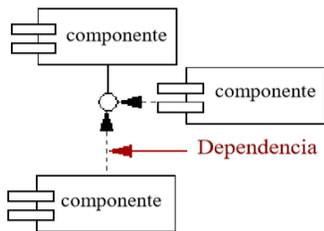
**Fuente:** (Joseph, aprendiendo UML en 24 horas, 2000)

## 2.9.6. Diagramas de Componentes

Un diagrama de componentes describe la organización de los componentes físicos de un sistema.

En la tabla N° 2.7, se muestra los iconos para diagramar los Diagramas de Componentes.

**Tabla N° 2.7. Estereotipos del Diagrama de Componentes**

ICONO	DESCRIPCIÓN	ICONO	DESCRIPCIÓN
	Componente		Dependencias
	Interface		

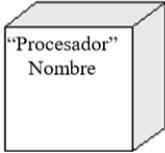
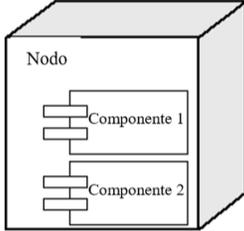
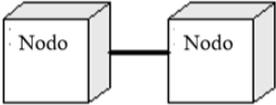
**Fuente:** (Garcia & Holgado, UML.Unified Modeling Language, 2018)

- ✓ **Componente:** Es un bloque de construcción física del sistema
- ✓ **Interface:** Una interface describe a un grupo de operaciones usada o creada por componentes.
- ✓ **Dependencias:** Las dependencias entre componentes se grafican usando flechas de puntos.

### 2.9.7. Diagramas de Distribución

El elemento primordial del hardware es un nodo, que es un nombre genérico para todo tipo de recurso de cómputo. Es posible usar dos tipos de nodos: un procesador, el cual puede ejecutar un componente, y un dispositivo que no lo ejecuta.

**Tabla N° 2.8. Estereotipos del Diagrama de Distribución**

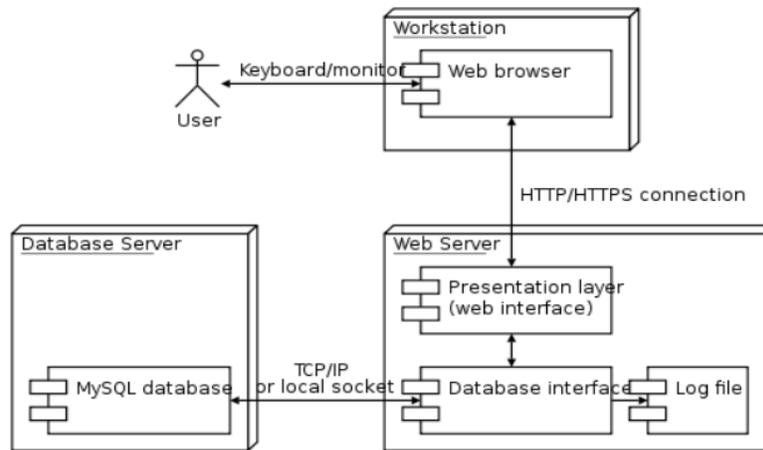
ICONO	DESCRIPCIÓN	ICONO	DESCRIPCIÓN
	Nodo		Componentes y nodos
	Asociación		

**Fuente:** (Joseph, aprendiendo UML en 24 horas, 2000)

- ✓ **Nodo:** Es un recurso físico capaz de ejecutar componentes de código. (Procesador).
- ✓ **Asociación:** Es la conexión física entre los nodos, como un ejemplo el Ethernet.

En la figura N° 2.9. Se muestra un ejemplo de cómo se realiza una diagramación de un Diagrama de Distribución.

**Figura N° 2.9. Diagrama de Distribución**



**Fuente:** (Joseph, aprendiendo UML en 24 horas, 2000)

Después de describir los diagramas que utiliza UML, se pasara a desarrollar las métricas de calidad del software.

## 2.10. MÉTRICAS DE CALIDAD DE SOFTWARE

Las métricas de calidad ayudan a entender tanto el proceso técnico que se utiliza para desarrollar un producto, como el propio producto. El proceso para intentar mejorarlo, el producto se mide para intentar aumentar su calidad.

El principio, podría parecer que la necesidad de la medición es algo evidente después de todo es lo que permite cuantificar y por consiguiente gestionar de forma más efectiva.

Hay varias razones para medir un producto.

- ✓ Para indicar la calidad del producto

- ✓ Para evaluar la productividad de la gente que desarrolla el producto.
- ✓ Para evaluar los beneficios en términos de productividad y de calidad.
- ✓ Para establecer una línea de base para la estimación.
- ✓ Para ayudar a justificar el uso de nuevas herramientas o de formación adicional.

Las mediciones del mundo físico pueden englobarse en dos categorías:

**Medidas directas.** En el proceso se encuentran el costo, y el esfuerzo aplicado. Las líneas de código producidas, velocidad de ejecución, el tamaño de memoria y los defectos observados en un determinado periodo de tiempo.

**Medidas Indirectas.** Se encuentran la funcionalidad, calidad, complejidad, eficiencia, fiabilidad, facilidad de mantenimiento, entre otras.

## **2.10.1. Métricas de Calidad del Modelo ISO/IEC25000**

### **2.10.1.1. Estándar ISO/IEC 25000**

La ISO/IEC 25000 también llamada SQuare (Requisitos y Evaluación de Calidad de Productos de Software), están conformados por las normas ISO/IEC 9126 e ISO/IEC 14598, surge para crear modelos, métricas, procesos y herramientas de evaluación de calidad del software como producto, por medio de la especificación de los requerimientos.

#### **Estructura de la familia ISO/IEC 25000**

- ✓ **ISO/IEC 2500n:** División de gestión de calidad.
- ✓ **ISO/IEC 2501n:** División del modelo de calidad.
- ✓ **ISO/IEC 2502n:** División de mediciones de calidad.
- ✓ **ISO/IEC 2503n:** División de requisitos de calidad.
- ✓ **ISO/IEC 2504n:** División de evaluación de la calidad.
- ✓ **ISO/IEC 25050–25099:** Estándares de extensión SQuaRE.

Al igual que la norma ISO/IEC 9126, este estándar define tres vistas diferenciadas en el estudio de la calidad.

- ✓ **Vista interna:** Esta vista se ocupa de las propiedades del software como: el tamaño, la complejidad o la conformidad con las normas de orientación a objetos.
- ✓ **Vista externa:** Vista que analiza el comportamiento del software en producción y estudia sus atributos, por ejemplo: el rendimiento de un software en una máquina determinada, el uso de memoria de un programa o el tiempo de funcionamiento entre fallos.
- ✓ **Vista en uso:** Mide la productividad y efectividad del usuario final al utilizar el software.

En la figura N° 2.10. Se observa los cuadros puntos más importantes de Requisitos de la norma ISO 25000.

**Figura N° 2.10.Requisitos de la norma ISO 25000**



**Fuente (Andrade Oliveira, 2014).**

A continuación se procede a desarrollar las características de calidad del Modelo ISO/IEC 25000.

### **2.10.1.2. Funcionalidad**

Es la capacidad del producto de software de satisfacer los requisitos funcionales prescriptos y las necesidades implícitas de los usuarios y tiene las siguientes sub características:

- ✓ **Adecuación:** La capacidad del producto de software para proveer un adecuado conjunto de funciones para las tareas y objetivos especificados por el usuario.
- ✓ **Exactitud:** Atributos del software relacionados con la disposición de resultados o efectos correctos o acordados.
- ✓ **Interoperabilidad:** La capacidad del producto de software de interactuar con uno o más sistemas especificados.
- ✓ **Seguridad:** La capacidad del producto de software para proteger la información y los datos de modo que las personas o los sistemas no autorizados no puedan leerlos o modificarlos.
- ✓ **Conformidad:** La capacidad del producto software para adaptarse a los estándares, convecciones o regulaciones en leyes y prescripciones relativas a la funcionalidad.

Para la medición del sistema, se determina las siguientes ecuaciones, en la tabla N° 2.9, donde se observa el punto de función, punto de función máximo, y la funcionalidad

**Tabla Nº 2.9. Ecuación de la Funcionalidad**

VARIABLE	ECUACIÓN
Punto de función	$PF = Cuenta\ Total \times (0.65 + 0.01 \times \sum Fi)$ <p>Dónde:</p> <p>Cuenta Total: Es la sumatoria del producto del factor de ponderación y valores de los parámetros.</p> <p><math>\sum F</math>: Es la sumatoria de los valores de ajuste de la complejidad.</p>
Punto de función Máximo	$PF_{max} = Cuenta\ Total \times (0.65 + 0.01 \times \sum Fi)$ <p>Dónde:</p> <p>Considerando el máximo valor de ajuste de complejidad</p> <p><math>\sum Fi = 70</math></p>
Funcionalidad	$Funcionalidad = \frac{PF}{PF_{max}}$

**Fuente:** (Pressman Roger S., 2010)

### Métricas Basadas en la Función

La métrica punto función (PF) se usa de manera efectiva como medio para medir la funcionalidad que entrega un sistema. PF se deriva empleando una relación empírica basada en medidas contables del dominio de la información del software y las evaluaciones de la complejidad de este.

En la tabla Nº 2.10. Se observa los factores de ponderación y los parámetros de medida.

**Tabla Nº 2.10. Factor de Ponderación**

PARÁMETROS DE MEDIDA		FACTORES DE PONDERACIÓN		
		SIMPLE	MEDIO	COMPLEJO
1	Número de Entradas de Usuario	3	4	6
2	Número de Salidas de Usuario	4	5	7
3	Número de Peticiones de Usuario	3	4	6
4	Número de Archivos	7	10	15
5	Número de Interfaces Externas	5	7	10

**Fuente:** (Pressman Roger S., 2010)

En la tabla Nº 2.11. Se muestra las características para calcula la funcionalidad del software.

**Tabla Nº 2.11. Variables para el Cálculo de la Funcionalidad**

CARACTERÍSTICAS	DESCRIPCIÓN
Número de entradas de usuario	Se cuenta cada entrada del usuario que proporcione al software diferentes datos aplicados a la aplicación. Las entradas deben ser distinguidas de las peticiones
Número de Salidas del usuario	Se encuentra cada salida que proporciona al usuario información orientada a la aplicación. En este contexto se refieren a informes, pantallas, mensajes de error
Número de peticiones al usuario	Una petición está definida como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida
Número de archivos	Se cuenta cada archivo maestro lógico es decir una agrupación lógica de datos que puede ser una parte

	en gran base de datos o un archivo independiente
Número de interfaces externas	Se cuentan todas las interfaces legibles por la máquina.

**Fuente:** (Pressman Roger S., 2010)

En la tabla N° 2.12. Se muestra los factores de Escala que hay que tomar en cuenta para la medición de complejidad.

**Tabla N° 2.12. Factor de Escala**

<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>
No Influencia	Incidental	Moderado	Medio	Significativo	Esencial

**Fuente:** (Pressman R., 2002)

Posteriormente determinar los valores de ajuste de complejidad, para ello se debe analizar las preguntas que se muestra en la Tabla N° 2.13, de acuerdo de Factor de Escala de la Tabla N° 2.12

**Tabla Nº 2.13. Ajustes de la Complejidad**

	Importancia	0%	20%	40%	60%	80%	100%	Fi
	Escala	No influencia	Incidencia	Moderado	Medio	Significativo	Esencial	
No	Factor/Preguntas	0	1	2	3	4	5	
1	¿Requiere el sistema copia de seguridad y de recuperación fiable?							
2	¿Requiere comunicación de datos?							
3	¿Existen funciones de procesamiento distribuido?							
4	¿El rendimiento es crítico?							
5	¿Sera ejecutado el sistema en entorno existente y fuertemente utilizado?							
6	¿Requiere el sistema Entrada de datos interactiva?							
7	¿Requiere la entrada de datos interactiva que las transiciones de entrada se llevan a cabo sobre múltiples pantallas o variadas opciones?							
8	¿Se actualizan los archivos maestros de forma interactiva?							
9	¿Son complejas de las entradas, las salidas de archivos y las peticiones?							
10	¿Es complejo el proceso Interno?							
11	¿Se diseña el código para ser reutilizable?							
12	¿Están incluidas en el diseño conversiones de instalación?							
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?							
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?							
15	Total							X

**Fuente:** (Pressman R., 2010)

### 2.10.1.3. Confiabilidad

La capacidad del producto de software para mantener un nivel de desempeño bajo condiciones establecidas, por un periodo tiempo y contempla las siguientes sub características.

- ✓ **Madurez:** Es la capacidad del producto de software para evitar fallas como resultado de errores en el software.
  
- ✓ **Tolerancia a Errores:** Es la capacidad del producto de software para mantener un nivel especificado de funcionamiento en caso de errores del software o de incumplimiento de su interfaz especificada.
  
- ✓ **Recuperabilidad:** Es la capacidad del producto de software para restablecer un nivel especificado de funcionamiento y recuperar los datos afectados directamente en el caso de una falla.
  
- ✓ **Conformidad:** Es la capacidad del producto software para adaptarse a estándares, convenciones y regulaciones referidas a la fiabilidad.

Para calcular la fiabilidad del sistema se toma en cuenta el periodo de tiempo en el que se ejecuta y se obtiene muestras con la siguiente ecuación de la tabla N° 2.14, y también las probabilidades de operación.

**Tabla N° 2.14. Ecuación de la Confiabilidad**

VARIABLE	ECUACIÓN
<b>Confiabilidad</b>	$F(t) = f * e(-\mu*t)$ <p>Dónde:</p> <p><i>f</i>: Funcionalidad del sistema.</p> <p><math>\mu</math>: Es la probabilidad de error que puede tener el sistema.</p> <p><i>t</i>: Tiempo que dura la gestión en el sistema.</p>
<b>Probabilidad de operación</b>	<p><math>P(T \leq t) = F(t)</math> Probabilidad de fallas (el termino en el cual el sistema trabaja sin fallas)</p> <p><math>P(T \leq t) = 1 - F(t)</math> Probabilidad de trabajo sin fallas (Tiempo en el cual no falla el sistema)</p>

**Fuente:** Elaboración Propia

#### 2.10.1.4. Usabilidad

Es la capacidad del producto de software de ser entendido, aprendido, usado y atractivo al usuario, cuando es utilizado bajo las condiciones especificadas y se divide en las siguientes subcaracterísticas.

- ✓ **Entendibilidad:** Es la capacidad del producto de software para permitir al usuario entender si el software es adecuado.
- ✓ **Aprendizaje:** Es la capacidad del producto de software para permitir al usuario aprender su aplicación. Un aspecto a considerar es la documentación del software.
- ✓ **Operabilidad:** Es la capacidad del producto de software para permitir al usuario operarlo y controlarlo.
- ✓ **Atracción:** Es la capacidad del producto para atraer al usuario.

- ✓ **Conformidad:** Es la capacidad del producto software para adaptarse a estándares, convenciones y regulaciones referidas a la usabilidad.

Para determinar la usabilidad del sistema se utiliza la siguiente ecuación que se encuentra en la tabla N° 2.15.

**Tabla N° 2.15. Ecuación de la Usabilidad**

VARIABLE	ECUACIÓN
Usabilidad	$FU = [(\sum Xi n /) * 100]$ <p>Dónde:  <i>Xi</i>: Es la sumatoria de valores  <i>n</i>: Es el número de preguntas</p>

**Fuente:** (Pressman Roger S., 2010)

Para responder a las preguntas se debe considerar la tabla N° 2.16 donde están las valoraciones de preguntas

**Tabla N° 2.16. Valoración de las Preguntas**

ESCALA	VALOR
Muy Bueno	25
Bueno	4
Regular	3
Malo	2
Pésimo	1

**Fuente:** (Pressman Roger S., 2010)

En la tabla N° 2.17, se muestra algunas de las preguntas para determinar la Usabilidad del sistema.

**Tabla N° 2.17. Preguntas para Determinar la Usabilidad**

NRO	PREGUNTAS	SI	NO	EVALUACIÓN
1	¿Puede utilizar con facilidad el sistema?			X
2	¿Puede controlar operaciones que el sistema solicite?			X
3	¿El sistema permitió la retroalimentación de información?			X
4	¿El sistema cuenta con interfaz agradable a la vista			X
5	¿La respuesta del sistema es satisfactoria?			X
6	¿Le parecen complicadas las funciones del sistema?			X
7	¿Los resultados que proporciona el sistema facilitan el trabajo?			X
8	¿Durante el uso del sistema se produjo errores?			X
<b>Total</b>				X

**Fuente:** (Pressman Roger S., 2010)

#### 2.10.1.5. Mantenibilidad

Capacidad del producto de software para ser modificado. Las modificaciones pueden incluir correcciones, mejoras o adaptación del software a cambios en el entorno, y especificaciones de requerimientos funcionales.

- ✓ **Analizabilidad:** Es la capacidad del producto de software para atenerse a diagnósticos de deficiencias o causas de fallas en el software o la identificación de las partes a ser modificadas.
- ✓ **Cambiabilidad:** Es la capacidad del software para permitir que una determinada modificación sea implementada.

- ✓ **Estabilidad:** Es la capacidad del producto de software para evitar efectos inesperados debido a modificaciones del software.
- ✓ **Capacidad de prueba:** Es la capacidad del software para permitir que las modificaciones sean validadas.
- ✓ **Conformidad:** Es la capacidad del producto software de cumplir los estándares o convenciones relativas a la mantenibilidad.

Para la medición de Mantenibilidad se debe calcular la ecuación que se encuentra en la tabla N° 2.18.

**Tabla N° 2.18. Ecuación de la Mantenibilidad**

VARIABLE	ECUACIÓN
<b>Mantenibilidad</b>	$IMS = \frac{[Mt - (Fa + Fc + Fd)]}{Mt}$ <p>Dónde:</p> <p><b>Mt:</b> Número de módulos de la versión actual.</p> <p><b>Fc:</b> Número de módulos en la versión actual que se han modificado.</p> <p><b>Fa:</b> Número de módulos en la versión actual que se han añadido.</p> <p><b>Fd:</b> Número de módulos de la anterior versión que se han borrado en la versión actual.</p>

**Fuente:** (Pressman Roger S., 2010)

#### **2.10.1.6. Portabilidad**

La capacidad del software para ser trasladado de un entorno a otro. El entorno puede incluir entornos organizacionales, de hardware o de software.

- ✓ **Adaptabilidad:** Es la capacidad del producto de software para ser adaptado a diferentes entornos especificados sin aplicar acciones o medios diferentes de los previstos para el propósito del software considerado.
- ✓ **Facilidad de Instalación:** Es la capacidad del producto de software para ser instalado en un ambiente especificado.
- ✓ **Reemplazabilidad:** Es la capacidad del producto de software para ser utilizado en lugar de otro producto de software, para el mismo propósito y en el mismo entorno.
- ✓ **Coexistencia:** Es la capacidad del producto para coexistir con otro software independiente en un ambiente común compartiendo recursos.
- ✓ **Conformidad:** Es la capacidad del producto software para adaptarse a estándares referidas a la portabilidad.

#### 2.10.1.7. Eficiencia

La capacidad del producto de software para proveer un desempeño adecuado, de acuerdo a la cantidad de recursos utilizados y bajo las condiciones planteadas.

- ✓ **Comportamiento temporal:** Es la capacidad del producto de software para proveer tiempos adecuados de respuesta y procesamiento, y ratios de rendimiento cuando realiza su función bajo las condiciones establecidas.
- ✓ **Utilización de recursos:** Es la capacidad del producto de software para utilizar cantidades y tipos adecuados de recursos cuando este funciona bajo las condiciones establecidas.
- ✓ **Conformidad:** Es la capacidad del producto software de cumplir los estándares o convenciones relacionadas a la eficiencia.

## 2.11. MÉTODO DE ESTIMACIÓN DE COSTES DEL SOFTWARE

La estimación de costo de un proyecto consiste en estimar los costos de los recursos necesarios (humanos y materiales) para completar las actividades del proyecto.

### 2.11.1. Modelo Constructivo de Costos (COCOMO)

El Modelo Constructivo de Costos COCOMO, por su acrónimo del inglés (COnstructive COst MOdel) es un modelo matemático de base empírica utilizado para estimación de costos de software. Incluye tres submodelos, cada uno ofrece un nivel de detalle y aproximación. (COCOMO, 2017)

### 2.11.2. Modelos de Estimación

En la estimación del tamaño de software COCOMO II utiliza tres técnicas

#### 2.11.2.1. Puntos Objeto

El procedimiento para determinar Puntos de Objeto de un proyecto de Software se resume en los siguientes:

**Determinar Cantidad de Objetos:** Estimar la cantidad de pantallas, reportes, componentes que contendrá aplicación.

**Clasificar cada instancia de un objeto:** Según sus niveles de complejidad (simple, medio o difícil).

**Dar el peso a cada objeto según el nivel de complejidad,** los pesos reflejan el esfuerzo relativo requerido para implementar una instancia de ese nivel de complejidad.

**Determinar la cantidad de Puntos de Objeto**, sumando todos los pesos de las instancias de los tipos de objetos especificados

### **2.11.2.2. Puntos de Función No Ajustados**

El modelo COCOMO II usa Puntos de Función y/o Líneas de Código Fuente (SLOC) como base para medir tamaño en los modelos de estimación de Diseño Temprano y Post-Arquitectura.

Los puntos de función están basados en información disponible en las etapas tempranas del ciclo de vida del desarrollo de software. COCOMO II considera solamente UTF (Puntos de Función no Ajustable).

$$FT = UFP \times TCF$$

Dónde:

*UFP: Puntos de Funcion no Ajustados*

*TCF: Factor de Complejidad Tecnica*

Para calcular los UFP, se deben identificar lo siguiente:

- ✓ Entradas Externas (Input)
- ✓ Salidas Externas (Outputs)
- ✓ Archivo Lógicos Internos (Archivos)
- ✓ Archivos Externos de Interface (Interfaces)
- ✓ Solicitudes Externas (Queries)

### **2.11.2.3. Líneas de Código Fuente**

El objetivo es medir la cantidad de trabajo intelectual puesto en el desarrollo de un programa. Definir una línea de código es difícil debido a que existen diferencias

conceptuales cuando se cuentan sentencias ejecutables y de declaraciones de datos en lenguajes diferentes.

A los efectos de COCOMO II, se eliminan las categorías de software que consumen poco esfuerzo. Así no están incluidas librerías de soporte, sistemas operativos, librerías comerciales y más, ni tampoco el código generado con generadores de código fuente. (Garcia L. M., 2012).

## 2.12. MÉTRICAS DE ESTIMACIÓN DE COSTOS COCOMO II

COCOMO II permite realizar estimaciones en función del tamaño de software, y de un conjunto de factores de coste y de escala.

- ✓ En los factores de coste se incluyen aspectos relacionados con la naturaleza del sistema, equipo, y de características propias del proyecto.
- ✓ Los factores de escala incluyen, la parte de escala producida a medida que un proyecto de software incrementa su tamaño. (Garcia L. M., 2012).

### ✓ **Modelo Básico**

Se utiliza para obtener una primera aproximación rápida del esfuerzo, y hace uso de la tabla N° 2.19. Donde están los constantes para calcular distintos aspectos de costes.

**Tabla N° 2.19. Constantes de Costes**

<b>MODO</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
<b>Orgánico</b>	2.40	1.05	2.50	0.38
<b>Semi-Libre</b>	3.00	1.12	2.50	0.35
<b>Rígido/Empotrado</b>	3.60	1.20	2.50	0.32

**Fuente: (COCOMO, 2017)**

Se puede observar que a medida que aumenta la complejidad del proyecto (modo), las constantes aumentan de 2.4 a 3.6, que corresponde a un incremento del esfuerzo del personal. Hay que utilizar con mucho cuidado el modelo básico puesto que se obvian muchas características del entorno.

#### ✓ **Modelo Intermedio**

En este modelo se introducen 15 atributos de coste para tener en cuenta en el entorno de trabajo, incrementando así la precisión de la estimación.

Para este ajuste, al resultado de la fórmula general se lo multiplica por el coeficiente surgido de aplicar los atributos que se decidan utilizar.

Los valores de coeficiente se muestran en la siguiente Tabla N° 2.20.

**Tabla N° 2.20. Coeficientes de COCOMO**

<b>PROYECTO SOFTWARE</b>	<b>A</b>	<b>B</b>
<b>Orgánico</b>	3.2	1.05
<b>Semi-Libre</b>	3.0	1.12
<b>Rígido</b>	2.8	1.20

**Fuente: (COCOMO, 2017)**

Se puede observar que los exponentes son los mismos que los del modelo básico, confirmando el papel que representa el tamaño; mientras que los coeficientes de los modos orgánico y rígido han cambiado, para mantener el equilibrio al rededor del Semi-Libre con respecto al efecto multiplicador de los atributos de coste.

✓ **Modelo Detallado**

Incluye todo lo del modelo intermedio además del impacto de cada conductor de coste en las distintas fases de desarrollo. Presenta principalmente dos mejoras respecto al modelo básico e Intermedio.

Por un lado, COCOMO define tres modos de desarrollo o tipos de proyecto:

- ✓ **Modo Orgánico:** Un pequeño grupo de programadores experimentados desarrollan software o proyectos relativamente sencillos en un entorno estable. Las líneas de código son menores de 50 KDLC.
- ✓ **Modo Semilibre o Semi-acoplado:** Corresponde a proyecto de intermedio entre el orgánico y el rígido (donde las líneas de código son menores de 300 KDLC); el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- ✓ **Empotrado o Rígido:** Se aplica este modelo en proyectos bastante complejos, en los que apenas se tiene experiencia y se engloban en un entorno de gran innovación técnica. Además, se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

En la tabla N° 2.21. Se detalla los coeficientes de COCOMO II, lo cual se utiliza para el cálculo del esfuerzo.

**Tabla N° 2.21. Detalle de Coeficiente de COCOMO II**

<b>PROYECTO DE SOFTWARE</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>
<b>Orgánico</b>	2.40	1.05	2.50	0.38
<b>Semiacoplado</b>	3.00	1.12	2.50	0.35
<b>Empotrado</b>	3.60	1.20	2.50	0.32

**Fuente:** (Pressman Roger S., 2010)

Se usará la técnica COCOMO para realizar una estimación del esfuerzo, necesario, para la elaboración del proyecto.

Para la realización del COCOMO previamente es necesario conocer el número de líneas de código, para realizar los cálculos del método de estimación se basará en la siguiente tabla.

En la tabla N° 2.22, se muestra las variables, para posteriormente calcular la ecuación del Método de COCOMO II.

**Tabla N° 2. 22. Ecuaciones del Método COCOMO II**

<b>Variable</b>	<b>Ecuación</b>	<b>Tipo/Unidad</b>
Esfuerzo requerido por el proyecto	$E = a \times (KLDC)^b \times FAE$	Persona/Mes
Tiempo requerido por el proyecto	$T = c \times (E)^d$	Meses
Número de personas requeridas para el proyecto	$NP = E / T$	Persona
Costo Total	$CT = Sueldo\ Mes \times NP \times T$	\$us.

**Fuente:** (Prentice Hall, 1981)

Posteriormente se procede a hallar la variable FAE, lo cual se obtiene mediante la multiplicación de los valores, para ello se mostrará la siguiente tabla de atributos de coste, en la tabla N° 2.23.

**Tabla N° 2.23. Atributos de Coste (FAE)**

Atributos que afectan al coste	Valor					
	Muy Bajo	Bajo	Nomina	Alto	Muy Alto	Extra Alto
<b>Atributos de Software</b>						
Fiabilidad del software	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de Datos		0,94	1,00	1,08	1,16	
Complejidad del producto	0,70	0,85	1,00	1,15	1,30	1,65
<b>Atributos de hardware</b>						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de Memoria Virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
<b>Atributos de Personal</b>						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia de la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los Programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en S.O. usando	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje de programación	1,14	1,07	1,00	0,95		
<b>Atributos del proyecto</b>						
Uso de Técnicas actuales de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	

**Fuente:** (Pressman Roger S., 2010).

## DESCRIPCIÓN DE VALORES

### Atributos

Cada atributo se cuantifica para un entorno de proyecto. La escala es **muy bajo - bajo - nominal - alto - muy alto - extremadamente alto**. Dependiendo de la calificación de cada atributo, se asigna un valor para usar de multiplicador en la fórmula.

El significado de los atributos es el siguiente, según su tipo:

### ATRIBUTOS DE SOFTWARE

- ✓ **RELY:** Garantía de funcionamiento requerida al software. Indica las posibles consecuencias para el usuario en el caso que existan defectos en el producto.
- ✓ **DATA:** Tamaño de la Base de Datos en relación con el tamaño del programa. Datos en bytes y K es el tamaño del programa en cantidad de líneas de código.
- ✓ **CPLX:** Representa la complejidad del producto.

### ATRIBUTOS DE HARDWARE

- ✓ **TIME:** Limitaciones en el porcentaje del uso de la CPU.
- ✓ **STOR:** Limitaciones en el porcentaje del uso de la memoria.
- ✓ **VIRT:** Volatilidad en la máquina virtual.
- ✓ **TURN:** Tiempo de respuesta requerido.

### ATRIBUTOS DEL PERSONA

- ✓ **ACAP:** Calificación de los analistas.
- ✓ **AEXP:** Experiencia del personal en aplicaciones similares.
- ✓ **PCAP:** Calificación de los programadores.
- ✓ **VEXP:** Experiencia del personal en la máquina virtual.

- ✓ **LEXP:** Experiencia en el lenguaje de programación a usar.

## **ATRIBUTOS DEL PROYECTO**

- ✓ **MODP:** Uso de prácticas modernas de programación.
- ✓ **TOOL:** Uso de herramientas de desarrollo de software.
- ✓ **SCED:** Limitaciones en el cumplimiento de la planificación. (COCOMO, 2017).

Una vez definido la métrica de calidad y estimación de costos (COCOMO II) se procede a definir la seguridad del software, aplicando la ISO/IEC 27000 que establece la seguridad de la información y las diferentes políticas y procedimientos que normaliza la gestión de la seguridad.

## **2.13. SEGURIDAD DE LA INFORMACIÓN**

### **2.13.1. Estándar ISO/IEC 27000**

Estándar para la seguridad de información, aprobada y publicada en octubre de 2005 por ISO. Especifica los requisitos necesarios para establecer, mantener y mejorar un Sistema de Gestión de la Seguridad de la Información (SGSI). SGSI (Sistema de Gestión de la Seguridad de la Información), conjunto de políticas y procedimiento que normaliza la gestión de la seguridad de la información de toda una organización, debe garantizar la disponibilidad, confidencialidad e integridad de la información.

### 2.13.1.1. ISO 27002

Es una guía de buenas prácticas que permite a las organizaciones mejorar la seguridad de su información mediante control y gestión. (Maidana S., Seguridad de la Información, 2014).

Figura N° 2.11. Estructura ISO 27002



Fuente: (Maidana S., Seguridad Informática, 2014)

### 3. MARCO APLICATIVO:

En esta etapa se desarrolló el Sistema de Gestión de Información de Fichas Médicas e Historial Clínico en la Especialidad de Odontología, para el Centro de Salud de Villa Cooperativa, y se muestra la metodología aplicada descrita en los siguientes puntos.

#### 3.1. Obtención de Requisitos

La tarea de la ingeniería de requisitos es fundamental para que un sistema sea eficiente, en este sentido, para el presente proyecto se realizaron las actividades que se indica en la Tabla N° 3.24

**Tabla N° 3.24. Tareas para la Obtención de Requisitos**

TAREAS	DESCRIPCIÓN
Entrevistas	Se realizó la entrevista al personal del centro de salud: <ul style="list-style-type: none"><li>✓ Director, responsable del centro de salud.</li><li>✓ Asistente, encargado de la dar fichas de atención.</li><li>✓ Enfermería, encargada de peso y talla de los pacientes.</li><li>✓ Doctor, encargado de elaborar el historial clínico.</li></ul>
Observación	Los problemas observados en el centro de salud son de que existe demora en la atención y duplicidad de los historiales clínicos, puesto que esto se realiza de forma manual razón por la cual existe perdidas de los historiales clínicos.
Documentación	Se tuvo acceso a la documentación física del centro de salud sobre los historiales clínicos.

**Fuente:** Elaboración Propia

### 3.1.1. Situación Actual de la Institución

### 3.1.2. Identificación de Actores

La identificación de actores permite conocer a las personas involucradas en el proceso de gestión y administración del Centro de Salud.

En la siguiente tabla N° 3.25, se muestra los actores, junto con la descripción de las funciones que desempeña en el Centro de Salud.

**Tabla N° 3. 25. Descripción de Actores**

<b>ACTOR</b>	<b>DESCRIPCIÓN</b>
<b>Asistente (ventanilla)</b>	Encargada de realizar las siguientes funciones: <ul style="list-style-type: none"><li>✓ Recepción de documentos de cada Paciente.</li><li>✓ Registrar los datos de los pacientes y el tutor o responsable del paciente.</li><li>✓ Verificar los requisitos presentados por parte de los pacientes.</li><li>✓ Otorga el número de ficha médica e historial clínico.</li><li>✓ Remitir la documentación recepcionada a enfermería.</li><li>✓ Comunica a los pacientes y personal diferentes actividades (reuniones, talleres, campañas y otros).</li></ul>
<b>Enfermera</b>	Encargado de colaborar en la parte de peso y estatura. Realiza las siguientes funciones: <ul style="list-style-type: none"><li>✓ Revisión y registró de PA., FC., FR., TEM., peso, y talla.</li><li>✓ Emite paso al doctor encargado.</li><li>✓ Elaborar informe de los pacientes atendidos.</li></ul>

<b>Doctor</b>	Encargado de realizar las siguientes funciones: <ul style="list-style-type: none"> <li>✓ Coordina con la enfermera, para la realización del historial clínico.</li> <li>✓ Realiza la revisión y registro del examen extra oral, examen intra oral y odontograma de los pacientes.</li> <li>✓ Realiza el diagnóstico del paciente.</li> <li>✓ Otorga recetas al paciente.</li> </ul>
<b>Director</b>	Encargado de realizar las siguientes funciones: <ul style="list-style-type: none"> <li>✓ Coordina con el doctor, para realizar el paso del paciente a otro centro de salud especializada siempre cuando el paciente lo requiera.</li> </ul>

**Fuente:** Elaboración Propia

A continuación procedemos a capturar los requerimientos requeridos para desarrollar el sistema, de acuerdo a las necesidades del usuario.

### **3.2. INGENIERÍA DE REQUERIMIENTOS**

Agrupar un conjunto de tareas que permite al analista de desarrollo, plasmar con claridad los requerimientos realizados por el cliente de una manera clara la cual permite realizar; Análisis de facilidad de lo que se pretende desarrollar; establecer la fecha de entrega, validar los requerimientos y administrarlo durante el desarrollo. En base a: (Pressman Roger S., 2010)

Una función es descrita como un conjunto de entradas, comportamientos y salidas. Los requerimientos funcionales pueden ser: Cálculos, detalles técnicos, manipulación de datos y otras funcionalidades específicas que se supone que un

sistema debe cumplir. Los requerimientos de comportamiento para cada requerimiento funcional se muestran en los casos de uso. Son complementados por los requerimientos no funcionales, que se enfocan en el cambio de diseño o la implementación.

En la tabla N° 3.26, se muestra las categorías de la función y el significado de cada uno de las categorías.

**Tabla N° 3.26. Categoría de las Funciones**

<b>CATEGORÍA DE LA FUNCIÓN</b>	<b>SIGNIFICADO</b>
Evidente	Debe realizarse, y el usuario debería saber que se ha realizado.
Oculto	Debe realizarse, aunque no es visible para los usuarios. Esto se aplica a muchos servicios técnicos, como guardar información en un mecanismo persistente de almacenamiento, las funciones ocultas muchas veces se omiten durante el proceso de obtención de los requerimientos.
Superflua	Opcionales, su solución no repercute significativamente en el costo ni en las funciones.

**En base a: (Arias, 2006)**

### 3.2.1. Requerimientos Funcionales y No Funcionales

#### 3.2.1.1. Requerimientos Funcionales

Los requisitos funcionales describen las interacciones entre el sistema, su entorno (usuario u otros sistemas) y las características que necesitan el sistema a partir de la información obtenida como parte de las tareas de obtención de requerimientos de la Tabla N° 3.24, sin tener en cuenta cuestiones de implementación. En la Tabla N° 3.27, se muestra los requerimientos funcionales.

**Tabla N° 3. 27. Requerimientos funcionales**

REF.	FUNCIÓN	CATEGORÍA
R.1.	Acceso al Sistema por tipo de Usuario (director, asistente y enfermera).	Evidente
R.2.	Gestión y administración de cargo de usuarios del sistema.	Oculto
R.3.	Desplegar menú de acuerdo al privilegio de cada usuario	Oculto
R.4.	Registro de fichas médicas.	Evidente
R.5.	Desplegar del servicio y especialidad	Evidente
R.6.	Desplegar el estado (Activo/Inactivo).	Oculto
R.7.	Registro de datos del paciente	Evidente
R.8.	Determina fecha y hora de la atención	Evidente
R.9.	Registro de peso y estatura	Evidente
R.10.	Registro de examen extra oral e Intra oral	Evidente
R.11.	Registro de odontograma	Evidente
R.12.	Registro de diagnostico	Evidente
R.13.	Registro de receta	Evidente
R.14.	Generación de reportes de ficha medica	Evidente
R.15.	Reporte de historial clínico	Evidente

<b>R.16.</b>	Generación de reportes de recetas medicas	Evidente
<b>R.17.</b>	Calculo de CEO	Oculto
<b>R.18.</b>	Calculo de CPO	Oculto
<b>R.19.</b>	Inicio y cierre de gestión del sistema.	Oculto
<b>R.20.</b>	Búsqueda de cada paciente e historial clínico.	Evidente

**Fuente :** Elaboracion propia

### 3.2.1.2. Requerimientos No Funcionales

Describen aspectos del sistema visibles por el usuario que no se relacionan en forma directa con el comportamiento funcional del sistema.

**Tabla Nº 3. 28. Requerimientos No Funcionales**

<b>REF.</b>	<b>FUNCIÓN</b>	<b>CATEGORÍA</b>
<b>R.1.</b>	El sistema debe funcionar correctamente en cualquier ordenador que disponga de un navegador Navegador, Mozilla y Chrome.	Evidente
<b>R.2.</b>	El acceso al sistema deberá ser sencillo, ya que los usuarios no disponen de grandes conocimientos técnicos en sistemas. La navegación por la aplicación tiene que ser clara para facilitar el manejo al usuario.	Evidente
<b>R.3.</b>	Acceso restringido a la aplicación mediante autenticación por usuarios y contraseñas. El sistema tendrá que evitar que el personal no autorizado, o sin permisos, pueda acceder a los datos privados de los usuarios, así como el acceso al sistema.	Evidente

**Fuente:** Elaboración Propia

A continuación procedemos a aplicar UML, para graficar, visualizar, especificar, construir y documentar el sistema.

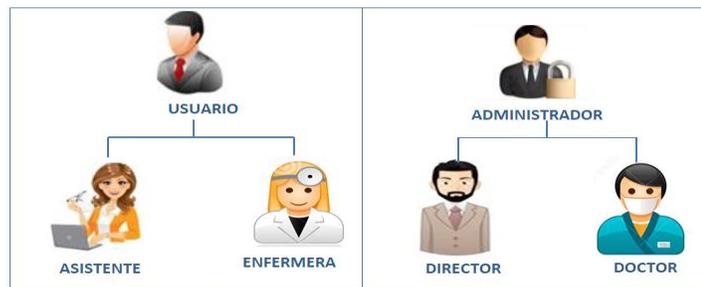
### 3.3. APLICACIÓN DE UML

En este punto se plasma, del análisis de requerimientos del sistema para realizar el diseño de Diagrama de casos de uso y presentación que describe el comportamiento del sistema frente a las acciones de los actores del mismo y funcionamiento del sistema.

#### 3.3.1. MODELO DE CASO DE USO

Primeramente, se muestra la descripción de tipos de usuarios que interactúan con los casos de uso del sistema.

**Figura N° 3.12. Descripción de Usuarios**



**Fuente:** Elaboración Propia

**Tabla N° 3.29. Descripción de Usuario y Administrador**

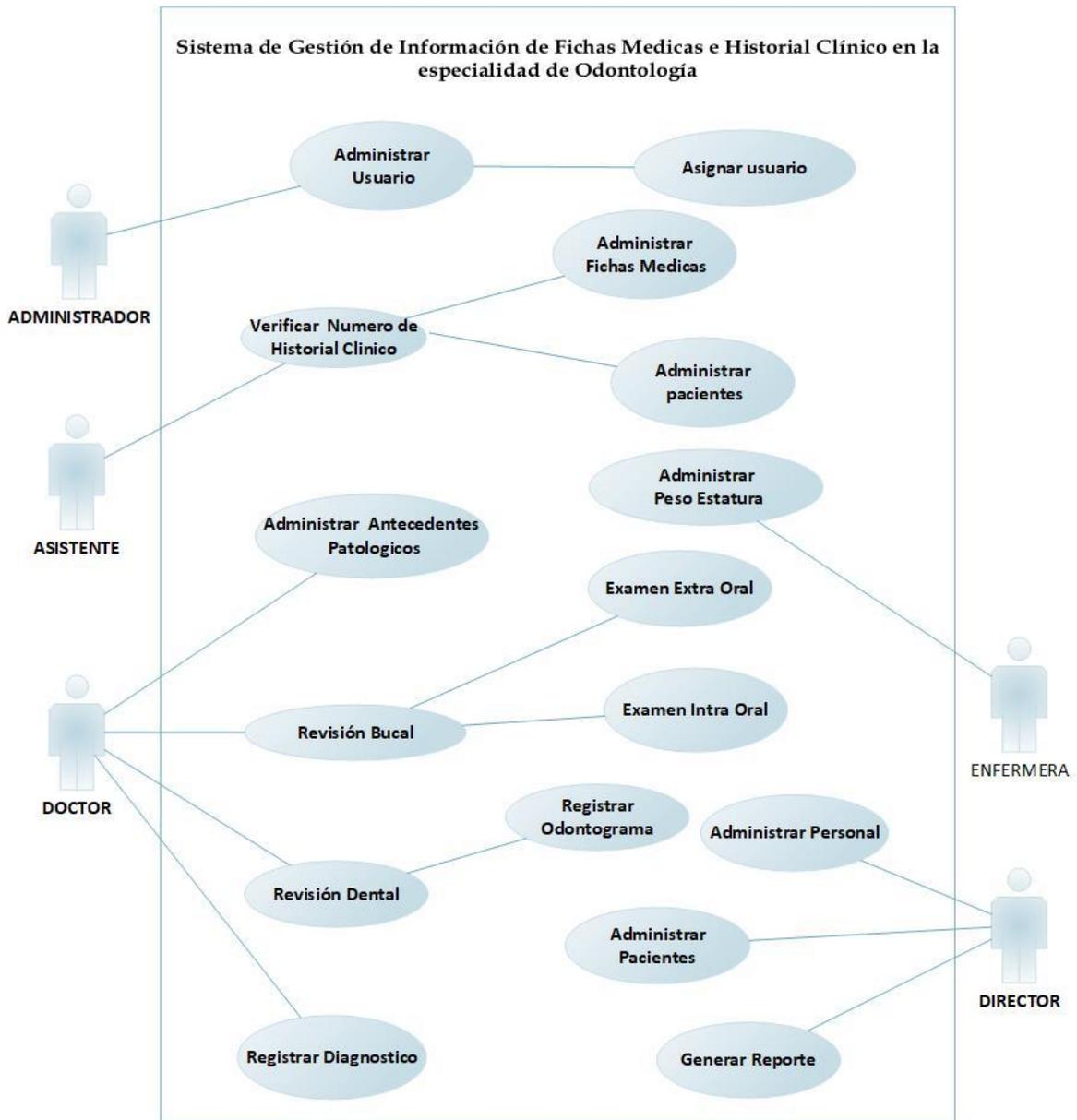
<b>DESCRIPCIÓN DE USUARIO Y ADMINISTRADOR</b>	
El usuario y administrador es aquel que tiene permiso para acceder al sistema, debe ingresar de acuerdo al rol asignado por el administrador.	
❖ Usuario	- Asistente - Enfermera
❖ Administrador	- Director - Doctor

**Fuente:** Elaboración Propia

### **3.3.2. Diagrama de Caso de Uso General del Sistema**

El caso de uso general del sistema se asemeja al concepto orientado a objeto, en la siguiente Figura N° 3.13, se muestra el modelado donde se puede apreciar cómo interactúan los actores sobre los casos de uso del sistema. Es un modelo de uso representado para la realización del sistema y los casos de uso que maneja el sistema.

**Figura Nº 3.13. Diagrama de Caso de Uso: GENERAL DEL SISTEMA**



**Fuente:** Elaboración Propia

### 3.3.2.1. Diagrama de Caso de Uso: ADMINISTRADOR

Figura N° 3.14. Diagrama de Caso de Uso: ADMINISTRADOR



Fuente: Elaboración Propia

Tabla N° 3.30. Caso de Uso: Administración de Usuario

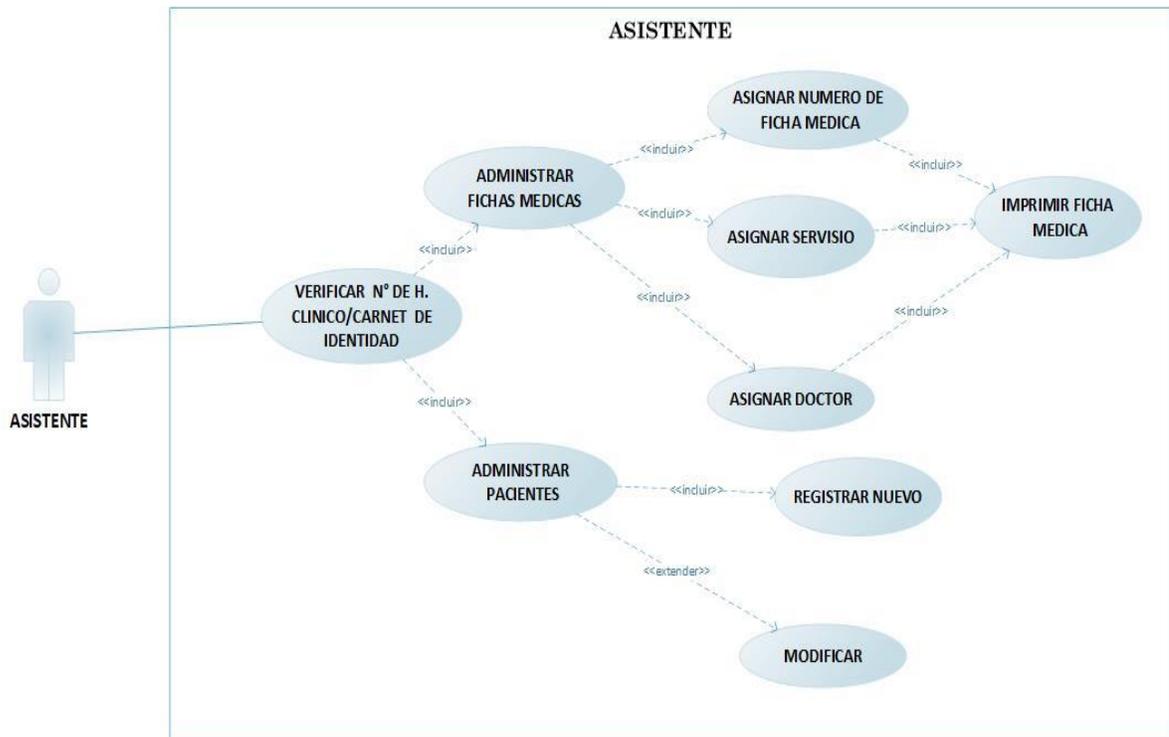
Caso de Uso: Administración de Usuario	
<b>Objetivo</b>	Describe el proceso de administrar el registro de los usuarios.
<b>Precondiciones</b>	El administrador debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Administrador.
<b>Descripción</b>	El administrador selecciona el menú de usuarios. El sistema despliega la lista de usuarios y muestra las acciones como: Adicionar, modificar y dar de baja. El administrador adiciona y designa el privilegio de cada

	<p>usuario de acuerdo a la función que desempeña.</p> <p>Dependiendo de la opción, realiza una acción al desplegar el formulario para Modificar, adicionar. Valida el formulario y guarda los datos con la opción realizada y retorna al listado con la acción realiza.</p>
--	---

Fuente: Elaboración Propia

### 3.3.2.2. Diagrama de Caso de Uso: Asistente

Figura N° 3.15. Diagrama de Caso de Uso: ASISTENTE



Fuente: Elaboración Propia

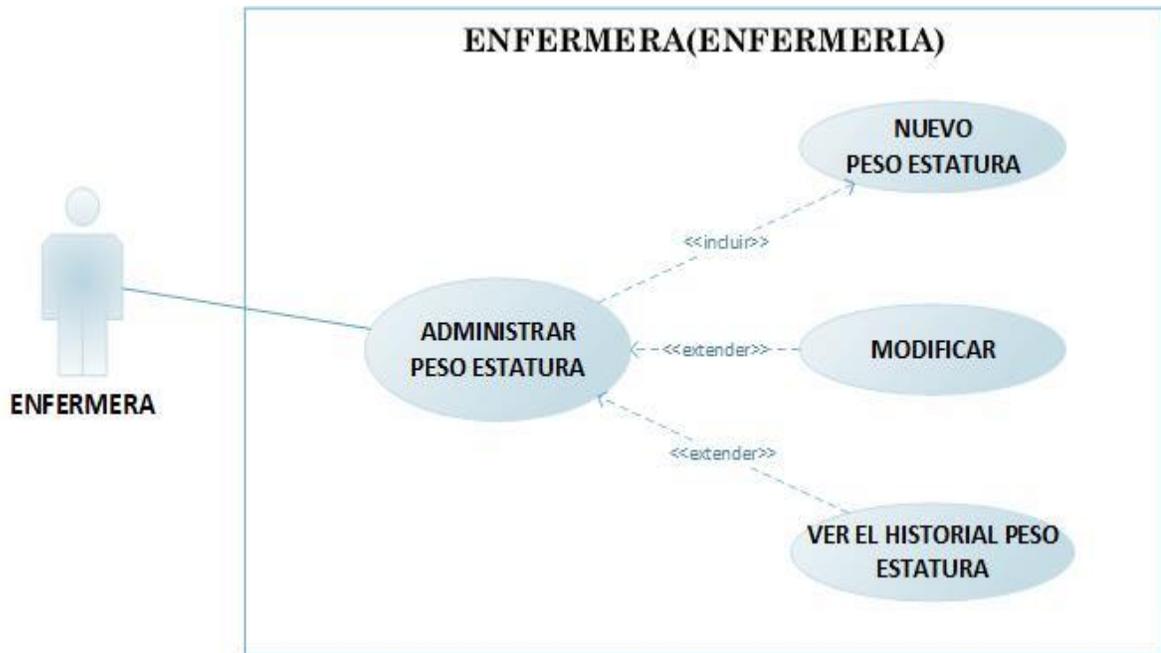
**Tabla Nº 3. 31. Caso de Uso: Asistente**

<b>Caso de Uso: Asistente</b>	
<b>Objetivo</b>	Describe el proceso de administrar de Fichas Medicas la administración de los pacientes.
<b>Precondiciones</b>	El Asistente debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Asistente.
<b>Descripción</b>	<p>El asistente realiza la verificación del historial clínico o número de carnet de identidad, si encuentra el número de historial clínico se le asigna la ficha médica al paciente, en caso contrario se registra nuevo paciente o se realiza la modificación de algún datos erróneo.</p> <p>Para la asignación de ficha médica se despliega el formulario (generar ficha medica) y se registra el tipo de atención, se asigna el doctor especialista y turno, posteriormente se elige la opción guardar. Y se imprime la ficha Médica de atención.</p> <p>Para el registro de nuevo paciente se despliega el formulario (Nuevo paciente), y el asistente pide datos del paciente o responsable tutor y procede a guardar registro.</p>

**Fuente:** Elaboración Propia

### 3.3.2.3. Diagrama de Caso de Uso: Enfermera (Enfermería)

Figura N° 3.16. Diagrama de Caso de Uso: ENFERMERA



Fuente: Elaboración Propia

Tabla N° 3.32. Caso de Uso: Enfermera

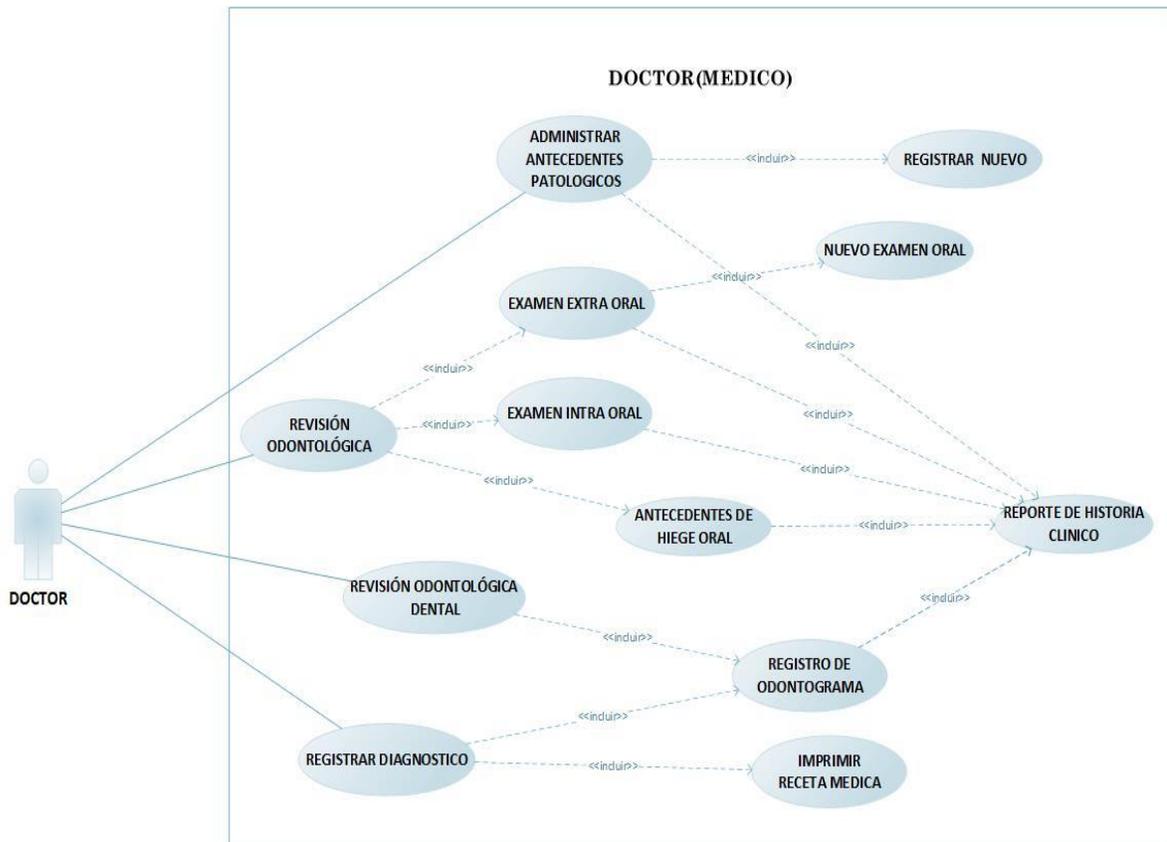
<b>Caso de Uso: Enfermera</b>	
<b>Objetivo</b>	Proceso de administrar de peso estatura.
<b>Precondiciones</b>	La enfermera debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Enfermera
<b>Descripción</b>	La enfermera selecciona el menú de peso estatura. El sistema despliega la lista de los pacientes que sacaron ficha médica para su atención, y muestra las opciones de:

	<ul style="list-style-type: none"><li>✓ Agregar nuevo peso estatura Se despliega el formulario (nuevo peso estatura) donde se registra: peso, talla, temperatura, presión arterial, frecuencia cardiaca y frecuencia respiratoria. Posteriormente se procede a guardar el registro con el botón guardar.</li><li>✓ Modificar registra Se despliega el formulario para modificar datos erróneos de peso estatura del paciente y se presiona el botón actualizar.</li><li>✓ Ver Historial de peso estatura Se despliega un listado de registros de pesos y estaturas anteriores del paciente.</li></ul>
--	---

**Fuente:** Elaboración Propia

### 3.3.2.4. Diagrama de Caso de Uso: Doctor

Figura Nº 3.17. Diagrama de Caso de Uso: DOCTOR



Fuente: Elaboración Propia

Tabla Nº 3.33. Caso de Uso: Antecedentes Patológicos

Caso de Uso: Antecedentes Patológicos	
<b>Objetivo</b>	Describe el proceso de antecedentes patológicos.
<b>Precondiciones</b>	El doctor debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Doctor

<b>Descripción</b>	<p>El doctor especialista elige la opción de registrar nuevo antecedente patológico.</p> <p>El sistema despliega el formulario para realizar el registro del antecedente patológico, y posteriormente se presiona el botón guardar.</p>
--------------------	---

**Fuente:** Elaboración Propia

**Tabla Nº 3.34. Caso de Uso: Revisión Odontológica**

<b>Caso de Uso: Revisión Odontológica</b>	
<b>Objetivo</b>	Describe el proceso de revisión odontológica
<b>Precondiciones</b>	El doctor debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Doctor
<b>Descripción</b>	<p>El doctor especialista elige la opción de revisión odontológica.</p> <p>El sistema muestra el formulario de registro de revisión odontológica, donde realiza el registro de: examen extra e intra oral, y también realiza el registro de antecedentes de higiene oral. Y posteriormente se selecciona el botón guardar.</p>

**Fuente:** Elaboración Propia

**Tabla N° 3.35. Caso de Uso: Revisión Odontológica Dental**

<b>Caso de Uso: Revisión Odontológica Dental</b>	
<b>Objetivo</b>	Describe el proceso de registro de revisión odontológica dental.
<b>Precondiciones</b>	El doctor debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Doctor
<b>Descripción</b>	El doctor especialista elige la opción de revisión odontológica dental. El sistema despliega el formulario (registrar revisión dental) donde se realiza el registro del estado de los 20 dientes temporales o en caso contrario los 32 dientes permanentes, Conjuntamente con los dientes importantes. Y se procede a guardar los registros con el botón guardar.

**Fuente:** Elaboración Propia

**Tabla N° 3.36. Caso de Uso: Diagnostico**

<b>Caso de Uso: Diagnostico</b>	
<b>Objetivo</b>	Describe el proceso de diagnóstico.
<b>Precondiciones</b>	El Doctor debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Doctor
<b>Descripción</b>	El doctor selecciona la opción nuevo diagnóstico. El sistema despliega el formulario (nuevo diagnostico), el

	<p>cual se registra subjetivo, objetivo, análisis y plan de acción, posterior mente se procede a guardar el registro.</p> <p>El sistema imprimiera receta para el paciente.</p>
--	---

**Fuente:** Elaboración Propia

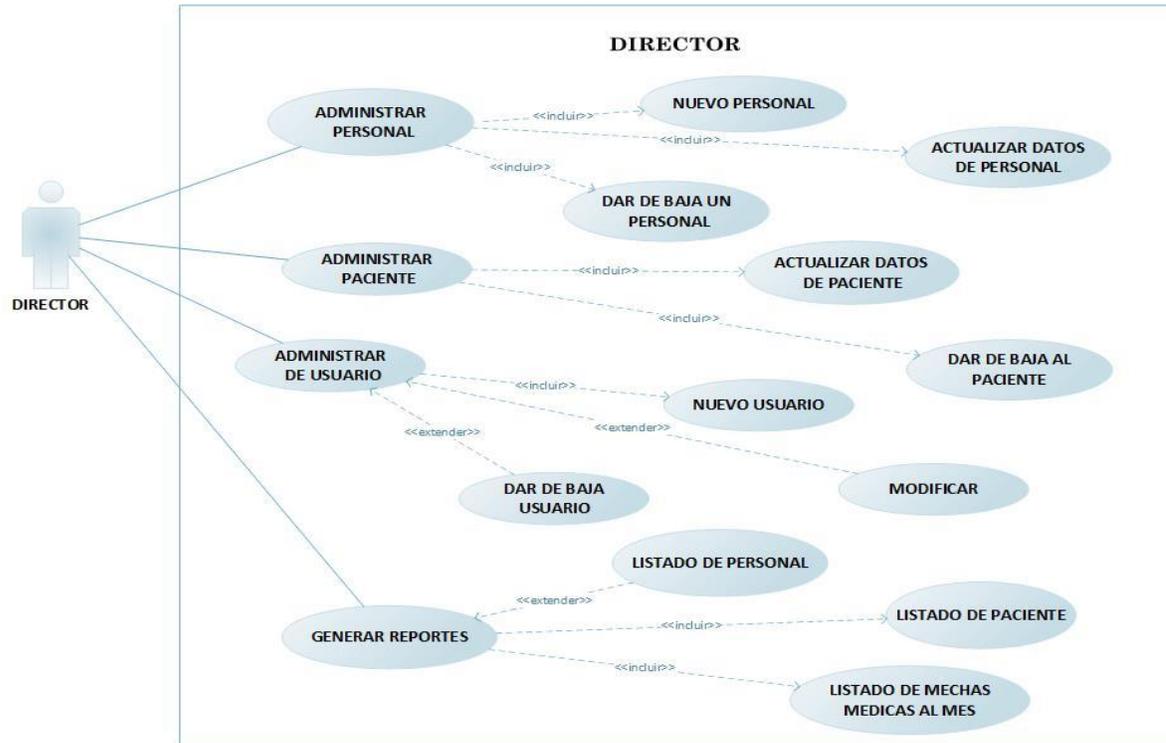
**Tabla N° 3.37. Caso de Uso: Reporte de Historial Clínico**

<b>Caso de Uso: Reportes Historial Clínico</b>	
<b>Objetivo</b>	Realización de historial clínico del paciente.
<b>Precondiciones</b>	El doctor debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Doctor
<b>Descripción</b>	El sistema nuestro el listado de los pacientes, con la opción de imprimir su historial clínico.

**Fuente:** Elaboración Propia

### 3.3.2.5. Diagrama de Caso de Uso: DIRECTOR

Figura N° 3.18. Diagrama de Caso de Uso: DIRECTOR



Fuente: Elaboración Propia

Tabla N° 3.38. Caso de Uso: Administración de Personal y Paciente

Caso de Uso: Administración de Personal y Paciente	
<b>Objetivo</b>	Describir el proceso de administración de personal y paciente.
<b>Precondiciones</b>	El director debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Director
<b>Descripción</b>	El sistema muestra listados del personal y paciente con las opciones de :

	<ul style="list-style-type: none"> <li>✓ Agregar nuevo</li> </ul> <p>El sistema despliega el formulario nueva persona, donde se registran los datos requeridos, indicando el tipo de persona (paciente o personal).y se procede a guardar.</p> <ul style="list-style-type: none"> <li>✓ Modificar datos de la persona</li> </ul> <p>El sistema despliega el formulario de modificar registro, y se procede a actualizar la información. Y se actualiza el registro.</p> <ul style="list-style-type: none"> <li>✓ Dar de baja persona</li> </ul> <p>El sistema muestra la opción de dar de baja de personal/paciente.</p>
--	--

**Fuente:** Elaboración Propia

**Tabla Nº 3.39. Caso de Uso: Administración de Usuarios Director**

<b>Caso de Uso: Administración de Usuarios</b>	
<b>Objetivo</b>	Describir el proceso de administración de usuario.
<b>Precondiciones</b>	El director debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Director
<b>Descripción</b>	<p>El sistema muestra listados de usuarios activos con las opciones de:</p> <ul style="list-style-type: none"> <li>✓ Asignar nuevo usuario</li> </ul> <p>El sistema despliega el formulario (nueva usuario), donde se registra el nombre de usuario, rol de usuario y su respectiva contraseña.</p> <ul style="list-style-type: none"> <li>✓ Modificar usuario</li> </ul>

	<p>El sistema despliega el formulario (modificar usuario), para modificar algún dato erróneo. Y se procede a actualizar los datos.</p> <p>✓ Dar de baja usuario</p> <p>El sistema muestra la opción de dar de baja usuario para inhabilitar un usuario.</p>
--	---

**Fuente:** Elaboración Propia

**Tabla N° 3.40. Caso de Uso: Reportes**

<b>Caso de Uso: Reportes</b>	
<b>Objetivo</b>	Describir el proceso de generar reportes.
<b>Precondiciones</b>	El director debe ingresar al sistema con usuario y contraseña.
<b>Actores</b>	Director
<b>Descripción</b>	<p>El sistema despliega las opciones de reporte del personal y paciente.</p> <p>Posteriormente se elige una de las opciones para imprimir el reporte.</p>

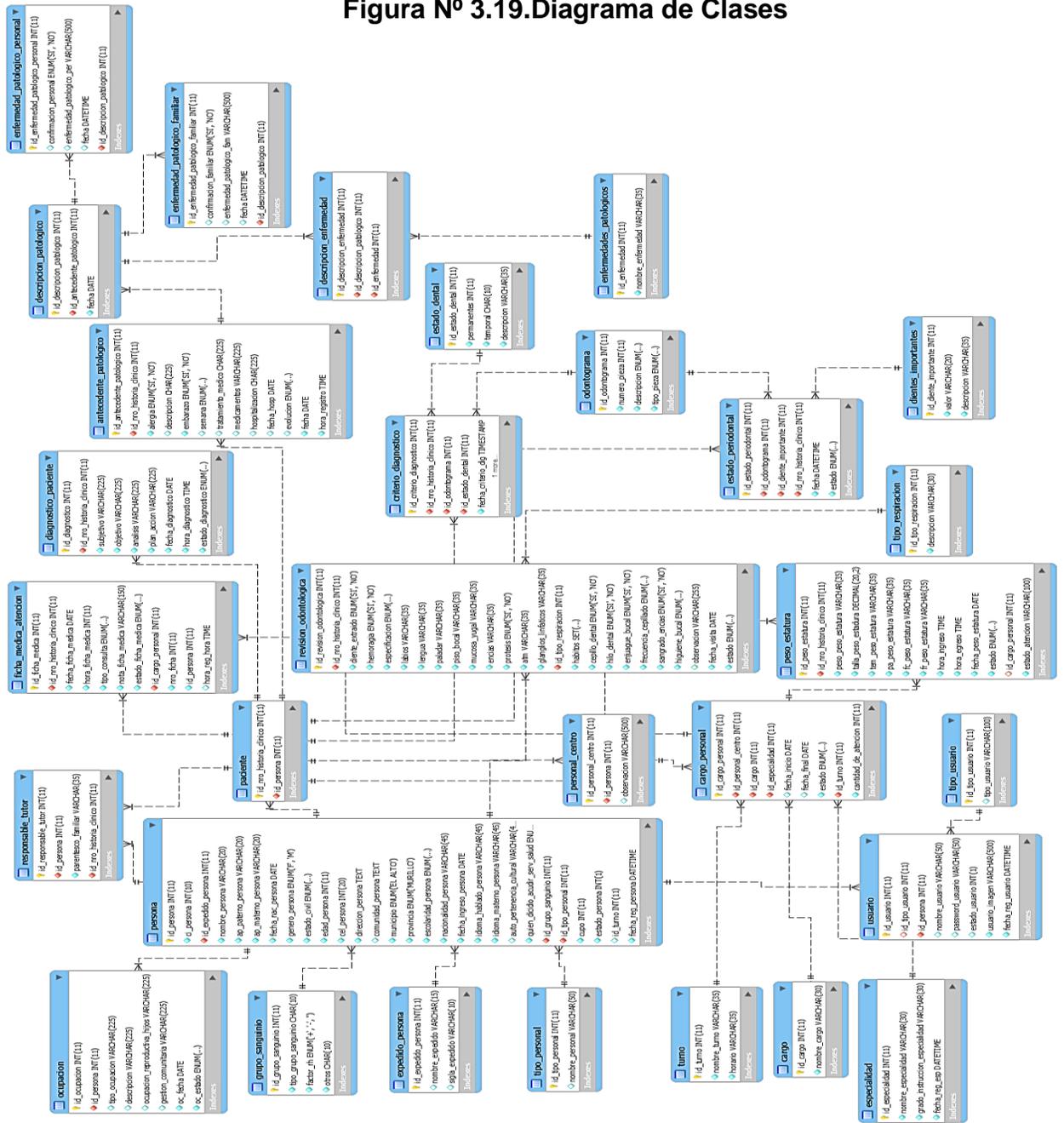
**Fuente:** Elaboración Propia

### 3.3.3. DISEÑO CONCEPTUAL

#### 3.3.3.1. Modelo Conceptual

En la Figura N° 3.19, se muestra el diagrama de clases del sistema SIFHO (Sistema de Gestión de Información de Fichas Medicas e Historial Clínico en la especialidad de Odontología)

Figura N° 3.19. Diagrama de Clases



Fuente: Elaboración propia

### 3.3.4. IMPLEMENTACIÓN DEL SISTEMA

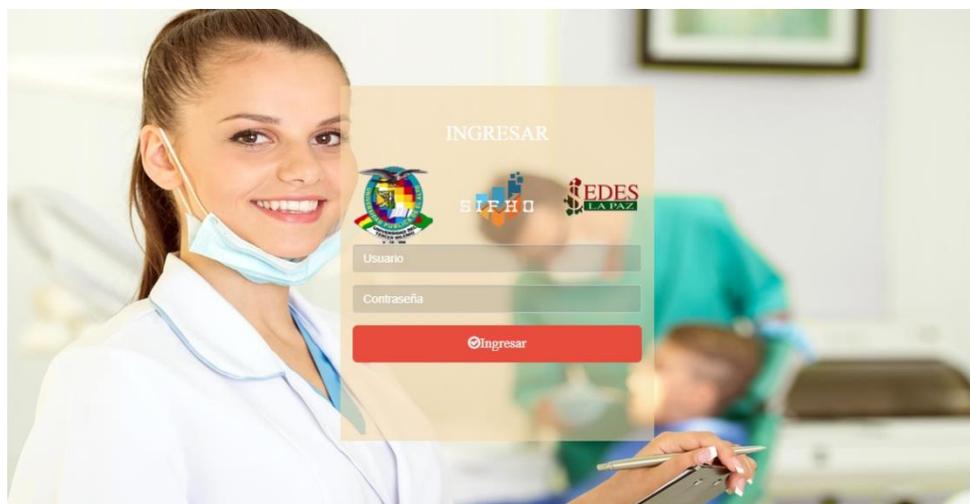
La fase de implementación consiste en mostrar el desarrollo de la presentación de las interfaces del sistema y sus elementos.

#### 3.3.4.1. Interfaz de Inicio de Sesión o Autenticación

**Tabla Nº 3. 41. Autenticación para Ingresar al Sistema**

Objetivo	Se debe autenticar para ingresar al sistema y posteriormente se muestra la interfaz correspondiente.
Descripción	Se debe acceder con nombre de usuario y contraseña de acuerdo al privilegio de Usuario asignado por el Administrador del Sistema.

#### ❖ Interfaz de Inicio de Sesión



**Fuente: (Elaboración Propia)**

### 3.3.4.2. Funcionalidad General del Sistema

#### ❖ Página General



Fuente: (Elaboración Propia)

### 3.3.4.3. Módulos que Integran el Sistema

#### 3.3.4.3.1. Módulo Asistente



Fuente: Elaboración Propia

## ❖ Registro de paciente

Introduzca el Número de Cédula de Identidad: 10067471

Introduzca el Nro. del Historial Clínico: [Empty]

Introduzca el Historial Clínico: [Empty]

**Detalles del Paciente**

Cédula de Identidad	Nombre Persona	Apellido Paterno	Apellido Materno	Género	Edad
10067471	MARINA	MAMANI	ULURI	F	28

**Detalles de Cita**

Servicio*	Especialista*	Fecha Asignación de Ficha
Odontología	JUAN EDUARDO ARECHIGA-(Odontologia)( Cupo: 5)	26/11/2019

Turno\*: MAÑANA 10:30 a 12:30 am-(0 de 5)

Estado\*: CONFIRMADO

Nota: URGENTE

\* Campos Obligatorios

Cancelar Limpiar Guardar

Fuente: Elaboración Propia

## ❖ Asignación de ficha

Introduzca el Número de Cédula de Identidad del Paciente: 10067471

Introduzca el Nro. del Historial Clínico: [Empty]

Introduzca el Historial Clínico: [Empty]

**Detalles de Cita**

Servicio*	Especialista*	Fecha Asignación de Ficha
Odontología	JUAN EDUARDO ARECHIGA-(Odontologia)( Cupo: 5)	26/11/2019

Turno\*: MAÑANA 10:30 a 12:30 am-(0 de 5)

Estado\*: CONFIRMADO

Nota: URGENTE

\* Campos Obligatorios

Cancelar Limpiar Guardar

Fuente: Elaboración Propia

❖ Reporte de ficha medica



**Centro de Salud**  
**“Villa Cooperativa”**



**Generacion de Ficha Medica**

NRO. DE FICHA:	1						
<b>1. DATOS DEL PACIENTE</b>							
Nombre:	MARINA	Ap.Paterno:	MAMANI	Ap.Materno:	ULURI	Genero:	F
CI:	10067471	Expedido:	LP	Observacion:	Ninguna observacion		
<b>2. DATOS DE FICHA MEDICA</b>							
Servicio:	ODONTOLOGIA	Especialista:	Dr(a).JUAN EDUARDO ARECHIGA GOMEZ				
Fecha:	2019-11-26	Hora:	MAÑANA - 10:30 a 12:30 am	Estado:	CONFIRMADO		
Nota:	URGENTE						

FECHA: 26-11-2019 14:19:54

Fuente: Elaboración Propia

3.3.4.3.2. Módulo enfermera

❖ Agregar nuevo peso estatura

Fuente: Elaboración Propia

## ❖ Historial de Peso Estatura

The screenshot shows a web browser window with the URL localhost/clinica/pesoestatura. The main content is a form titled 'Historial de Peso Estatura' for patient C.I. 10067471, MARINA AP. PATERNO MAMANI AP. MATERNO ULURI, female, 28 years old. The form includes a date field set to 2019-11-26 and a table with the following data:

Peso	Talla	Temperatura °C	Presión Arterial	Frecuencia Cardiaca	Frecuencia Respiratoria
60	60.00	60	60	60	60

A 'Cerrar' button is located below the table. The left sidebar shows a menu with 'Inicio', 'Persona', 'Ficha Médica', 'Peso Estatura', and 'Revisión Odontológica'.

Fuente: Elaboración Propia

### 3.3.4.3.3. Módulo doctor

## ❖ Revisión de Antecedentes Patológicas

The screenshot shows a web browser window with the URL localhost/clinica/patologico/vista\_antecedente\_patologico. The main content is a form titled 'Registrar Antecedente Patológico' for patient C.I. 10067471, MARINA MAMANI ULURI, female, 28 years old. The form includes the following fields:

- Datos Personales del Paciente:** C.I. 10067471, Nombre Persona MARINA, Apellido Paterno MAMANI, Apellido Materno ULURI, Genero F, Edad 28.
- Antecedentes Patológicos:**
  - Alergia:** Descripción: no tiene alergia; Embarazo: SI/NO; EnSemanas: 0; Tratamiento Médico: no tratamiento.
  - Medicamentos:** no medicamentos; Hospitalización por: no hospitalizado; Fecha de Hospitalización: dd/mm/aaaa; Evolución: Seleccione\*.
  - Descripción Patológica:** Familiar: SI/NO; Personal: SI/NO.

Buttons for 'Cancelar', 'Limpiar', and 'Guardar' are at the bottom. The left sidebar shows a menu with 'Inicio', 'Persona', 'Ficha Médica', 'Peso Estatura', 'Revisión Odontológica', 'Diagnóstico', 'Usuario', and 'Especialidad'.

Fuente: Elaboración Propia

## ❖ Revisión odontológica

**Registrar Revisión Odontológica**

C.I.: 10087471 NOMBRE: MARINA AP. PATERNO, MAMANI AP. MATERNO, ULURI GÉNERO: F EDAD: 28

**DATOS DE REVISIÓN ODONTOLÓGICA**

Dientes Extraídos?  SI  NO Hemorragia  SI  NO Especificación: NINGUNO Fecha Ult. Visita: 26/11/2019 Hábitos:  FUMAR  BEBER  OTRO  NINGUNO

**EXAMEN EXTRA ORAL**

Atm: Atm Gargajos Linfáticos: Gargajos Respirador: NASAL Observación: NINGUNA OBSERVACION

**EXAMEN INTRA ORAL**

Labios: Labios Lengua: Lengua Paladar: Paladar

Piso de la Boca: Piso de la Boca Mucosa Yugal: Mucosa Yugal Encías: Encías Utiliza Prótesis Dental:  SI  NO

**ANTECEDENTES DE HIGIENE ORAL**

Utiliza Cepillo Dental?  SI  NO Utiliza Hilo Dental?  SI  NO Utiliza Enjuague Bucal?  SI  NO Frecuencia del Cepillado Dental? NINGUNO

Durante el Cepillado Dental le Sangran las encías?  SI  NO Higiene Bucal:  BUENA  REGULAR  MALA

Cancelar Limpieza Guardar

Fuente: Elaboración Propia

## ❖ Revisión de dientes permanentes

**Registrar Dientes de Personas**

**DIENTES PERMANENTES**

Nombre: JUAN  
N° Historial: 1047  
C.I.: 10087400  
Edad: 28  
sesion\_numero historial: 1047

**Dientes Normales**

18	17	16	15	14	13	12	11	21	22	23	24	25	26	27	28
-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-	-0-

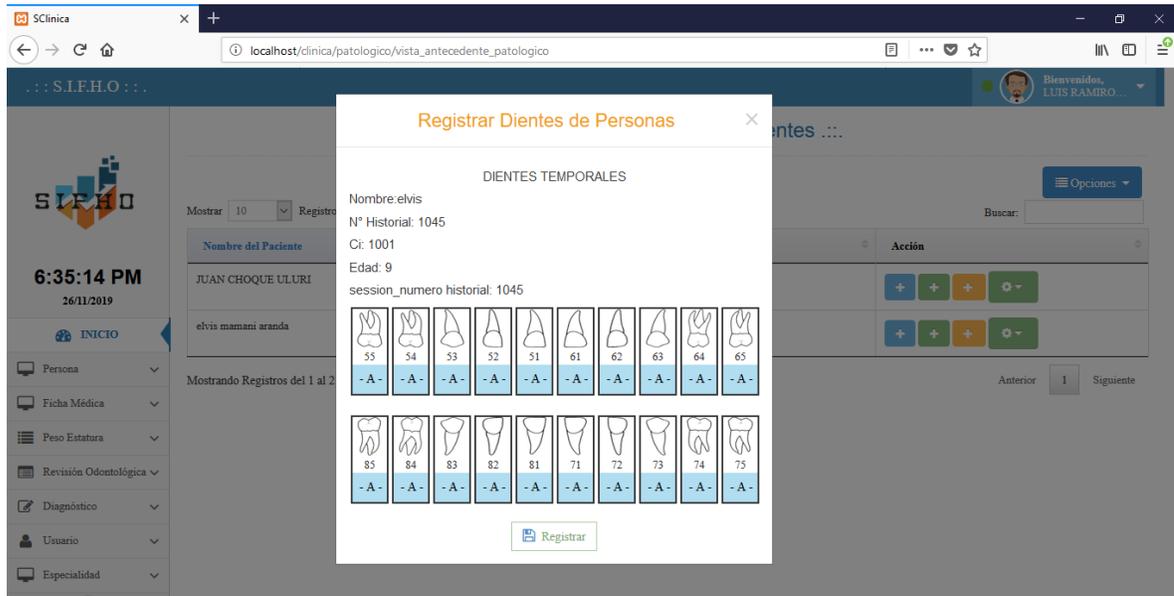
**Dientes Importantes**

37	36	11	21	26	27
-0-	-0-	-0-	-0-	-0-	-0-
-0-	-0-	-0-	-0-	-0-	-0-
46	47	41	31	37	36

Registrar

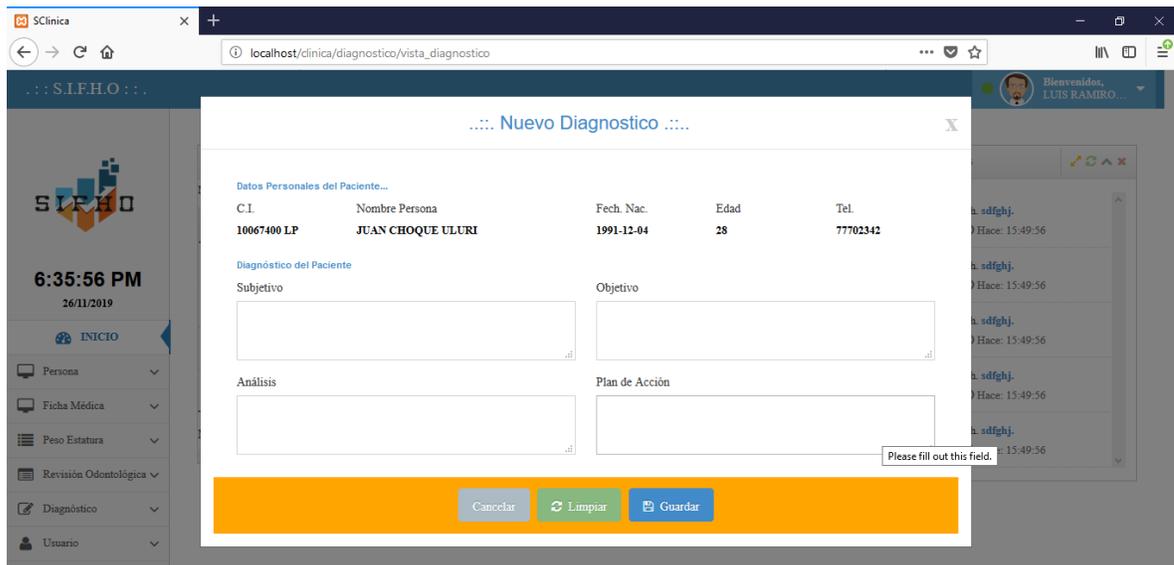
Fuente: Elaboración Propia

## ❖ Revisión de dientes temporales



Fuente: Elaboración Propia

## ❖ Registro de diagnostico



Fuente: Elaboración Propia



### 3.3.4.3.4. Módulo Director

#### ❖ Administración del personal

7	69321450 CB	JUAN EDUARDO ARECHIGA GOMEZ	30	77772501	O	Personal	Odontologia	2019-07-01 12:54:51	 
8	78954111 LP	PILAR ALEJANDRA LOZA ALIAGA	24	789652344	AB +	Personal	Odontologia	2019-07-01 12:55:04	 
9	6325411 LP	LUIS RAMIRO QUINO MARCA	26	63254895	A -	Personal	Ingenieria de Sistemas	2019-07-01 12:55:47	 
10	3698524 TJ	MANUEL ALARCON RAMOS	21	6985472	AB +	Paciente		2019-07-01 13:30:49	 

Fuente: Elaboración Propia

#### ❖ Adicionar nuevo personal

Cedula de Identidad	Expedido	Nombre Persona	Apellido Paterno
<input type="text" value="Introduzca CI"/>	<input type="text" value="...:Seleccione:..."/>	<input type="text" value="Introduzca Nombre de Person..."/>	<input type="text" value="Introduzca Apellido Paterno"/>
Apellido Materno	Fecha de Nacimiento	Genero	Estado Civil
<input type="text" value="Introduzca Apellido Materno"/>	<input type="text" value="mm / dd / yyyy"/>	<input type="text" value="...:Seleccione:..."/>	<input type="text" value="...:Seleccione..."/>
Número de Celular	Dirección	Comunidad de Persona	Escolaridad
<input type="text" value="Introduzca Celula..."/>	<input type="text" value="Introduzca Direcc..."/>	<input type="text" value="Introduzca la Comunidad del Personal"/>	<input type="text" value="...:Seleccione:..."/>
Nacionalidad	Idioma Hablado	Idioma Materno Hablado	
<input type="text" value="Boliviano(a)"/>	<input type="text" value="Introduzca IDIOMA Hablado"/>	<input type="text" value="Introduzca Idioma MATERNO Hablado"/>	
Autopertenencia Cultural	Quien decido acudir a la Clínica	Grupo Sanguineo	
<input type="text" value="Introduzca IDIOMA"/>	<input type="text" value="...:Seleccione:..."/>	<input type="text" value="*seleccione Sanguineo"/>	
Seleccione tipo de Paciente: *			
<input type="text" value="Personal"/>			
Cargo para el Personal Funcionario			
Cargo	Especialidad	Turno	
<input type="text" value="...:Seleccione Cargo:..."/>	<input type="text" value="...:Seleccione la Especialidad:..."/>	<input type="text" value="...:Seleccione el Turno:..."/>	
Fecha de Inicio	Fecha Final	Cupo	
<input type="text" value="mm / dd / yyyy"/>	<input type="text" value="mm / dd / yyyy"/>	<input type="text" value="Introduzca el cupo del Doctor"/>	
Observación			
<input type="text" value="Introduzca Alguna Observacion"/>			

Fuente: Elaboración Propia

## ❖ Reporte personal



### Centro de Salud "Villa Cooperativa"



NÓMINA GENERAL DE: Personal

NRO	NOMBRE COMPLETO	CEDULA DE IDENTIDAD	EDAD	F. DE NACIMIENTO	TIPOS DE PACIENTE/PERSONAL	ESPECIALIDAD	GRUPO SANGUÍNEO	DIRECCIÓN	CELULAR	CUPO	TURNO Y HORARIO	FECHA DE REGISTRO
1	JUAN EDUARDO ARECHIGA GOMEZ	69321450 CB	30	1989-02-02	Personal	Odontología	O	calle los pinos	7772501	5	MAÑANA 8:00 a 12:30 am	2019-07-01 12:54:51
2	PILAR ALBAJANDRA LOZA ALIAGA	78954111 LP	24	1995-06-05	Personal	Odontología	AB +	ave juan pablo II	789852344	10	MAÑANA 8:00 a 10:30 am	2019-07-01 12:55:04
3	LUIS RAMIRO QUINO MARCA	632141 LP	26	1993-06-15	Personal	Ingeniería de Sistemas	A -	AVE 24 DE SEPTIEMBRE	83284895	0	MAÑANA 8:00 a 12:30 am	2019-07-01 12:55:47
4	LUCIANA VALENTINA ALBAJANDRIA LOPEZ	8698324 CB	29	1990-05-05	Personal	Médico General	B +	AVE SANTA PE	73886974	10	TARDE 2:30 a 6:30 pm	2019-07-02 12:37:43

FECHA DE EMISIÓN: 2019-12-05 13:26:59

Fuente: Elaboración Propia

## ❖ Administración del paciente

Nro	CI	Nombre y Apellidos	Edad	Celular	G. Sanguíneo	Tipo Personal	Especialidad	Fecha Reg.	Acción
1	10067471 LP	marina mamani uluri	0	77777	A +	Paciente		2019-11-23 14:51:02	[Edit] [Delete]
2	1001 LP	elvis mamani aranda	9	898989	A +	Paciente		2019-11-26 17:45:09	[Edit] [Delete]
3	10067400 LP	JUAN CHOQUE ULURI	28	77702342	A +	Paciente		2019-11-26 17:42:43	[Edit] [Delete]
4	10067 LP	asda xdsadf gfhfjhg	0	12	B +	Paciente		2019-11-02 18:49:14	[Edit] [Delete]
5	10067489 LP	edvin alderson uluri	21	78789787	A +	Paciente		2019-11-02 18:32:52	[Edit] [Delete]

Fuente: Elaboración Propia

## ❖ Adicionar nuevo Paciente

Fuente: Elaboración Propia

## ❖ Reporte paciente



### Centro de Salud "Villa Cooperativa"



#### NÓMINA GENERAL DE: Paciente

NRO	NOMBRE COMPLETO	CEDULA DE IDENTIDAD	EDAD	F. DE NACIMIENTO	TIPOS DE PACIENTE/PERSONAL	ESPECIALIDAD	GRUPO SANGUÍNEO	DIRECCIÓN	CELULAR	CUPO	TURNO Y HORARIO	FECHA DE REGISTRO
1	MANUEL ALARCON RAMOS	3898524 TJ	21	1998-05-05	Paciente		AB +	ave 10 de octubre	8985472	0		2019-07-01 13:30:49
2	JONAS RAMOZ QUIROGA	9688701 CB	13	2006-06-30	Paciente		AB -	ave 12 de septiembre	38520147	0		2019-07-01 13:30:33
3	LUCAS ALEJANDRO RAMIREZ GOMEAL	89652147 LP	7	2012-05-05	Paciente		B +	AVE 3 DE SANTIAGO	96328874	0		2019-07-02 18:27:32
4	asda xdsadf ghfjhg	10067 LP	0	2019-07-26	Paciente		B +	sdsad	12	0		2019-11-02 18:49:14
5	edwin alderson uluri	10067489 LP	21	1998-02-03	Paciente		A +	sonojkdsjfsdfjsalñ kjfdksjafkd	78789787	0		2019-11-02 18:32:52
6	elvis mamani aranda	100 LP	9	2010-11-02	Paciente		A +	callefjdsajfkdsj	898989	0		2019-11-02 19:27:52
7	lucifer belial abadon	888 LP	13	2006-06-06	Paciente		A +	el averno	6888	0		2019-11-12 15:56:38

FECHA DE EMISIÓN: 2019-12-05 13:27:02

Fuente: Elaboración Propia

### 3.4. MÉTRICAS DE CALIDAD DE SOFTWARE

Para realizar la medición de calidad de software se aplicará la norma ISO-25000, define un modelo general de calidad que ayudará a demostrar la confiabilidad del sistema.

#### 3.4.1. Funcionalidad

La funcionalidad de un software, se mide según la complejidad del mismo, por esta razón corresponde derivar medidas directas como es el punto función que cuantifica el tamaño del sistema en términos de las funciones del usuario. Para la funcionalidad se debe determinar las cinco características de información, descritas a continuación en la Tabla N° 3.42.

**Tabla N° 3.42. Características de la Funcionalidad**

<b>CARACTERÍSTICAS</b>	<b>DESCRIPCIÓN</b>
<b>Número de entradas de usuario</b>	Número de Entradas de usuario, se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación.
<b>Número de Salidas del usuario</b>	Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a: Informes, reportes, pantallas, mensajes de error y más.
<b>Número de peticiones al usuario</b>	Una petición está definida como una entrada interactiva que resulta de la generación de algún tipo de respuesta en forma de salida interactiva.

<b>Número de archivos</b>	Se toma en cuenta cada archivo maestro lógico, estos pueden ser: grupos lógicos de datos (tablas en base de datos), o un archivo independiente (archivo de texto u hoja de cálculo).
<b>Número de interfaces externas</b>	Se cuentan todas las interfaces legibles por la máquina.

**Fuente:** Elaboración propia

Contado por separado de acuerdo a la Tabla N° 3.43, se tiene los siguientes datos:

**Tabla 3.43. Parámetros de Medición**

<b>PARÁMETROS DE MEDIDA</b>	<b>CUENTA</b>
Número de entradas de usuario	55
Número de salidas de usuario	35
Número de peticiones de usuario	60
Número de archivos	65
Número de interfaces externas	2

**Fuente:** Elaboración Propia

Para calcular el punto de función, se debe realizar el cálculo de la Cuenta Total con los factores de ponderación especificados en la siguiente tabla:

**Tabla N° 3.44. Punto de Función (Factores de Ponderación)**

Parámetros de Medida		Cuenta	Factores de Ponderación			Total
			Simple	Medio	Complejo	
1	Número de entradas de usuario	55	3	4	6	220
2	Número de salidas de usuario	35	4	5	7	175
3	Número de peticiones de usuario	60	3	4	6	240
4	Número de archivos	65	7	10	15	650
5	Número de interfaces externas	2	5	7	10	14
<b>Cuenta Total</b>						<b>1299</b>

**Fuente:** Elaboración Propia

En la Tabla N° 3.44, se muestra la Cuenta Total que se obtiene de la sumatoria de los factores de ponderación y las características de parámetros de medida.

Para determinar los valores de complejidad, se basa de acuerdo a las respuestas obtenidas de las preguntas que se muestra en la siguiente Tabla N° 3.45

**Tabla N° 3.45. Valores de Ajuste de la complejidad**

	Importancia	0%	20%	40%	60%	80%	100%	Fi
	Escala	No influencia	Incidencia	Moderado	Medio	Significativo	Esencial	
N	Factor/Preguntas	0	1	2	3	4	5	
1	¿Requiere el sistema copia de seguridad y de recuperación fiable?						X	5
2	¿Requiere comunicación de datos?						X	5
3	¿Existen funciones de procesamiento distribuido?				X			3
4	¿El rendimiento es crítico?				X			3
5	¿Sera ejecutado el sistema en entorno existente y fuertemente utilizado?						X	5
6	¿Requiere Entrada de datos interactiva, en LÍNEA?						X	5
7	¿Requiere la entrada de datos interactiva que las transiciones de entrada se llevan a cabo sobre múltiples pantallas o variadas opciones?						X	5
8	¿Se actualizan los archivos maestros de forma interactiva?					X		4
9	¿Son complejas de las entradas, las salidas de archivos y las peticiones?				X			3
10	¿Es complejo el proceso Interno?				X			3
11	¿Se diseña el código para ser reutilizable?					X		4
12	¿Están incluidas en el diseño conversiones de instalación?				X			3
13	¿Se ha diseñado el sistema para soportar múltiples instalaciones en diferentes organizaciones?			X				2
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?						X	5
<b>Factor de Complejidad Total(FCT)</b>								<b>55</b>

**Fuente:** Elaboración propia

Para calcular los puntos de función, se utiliza la siguiente ecuación:

$$PF = Cuenta\ Total \times (0.65 + 0.01 \times \sum Fi)$$

Dónde:

**Cuenta Total:** Es la sumatoria del producto del factor de ponderación.

$\sum Fi$ : Es la sumatoria de los valores de ajuste de la complejidad.

0.65: Valor mínimo de ajuste.

0.01: Factor de conversión, es decir un error de 1%

**Calculando:** El punto función de según la ecuación:

$$PF = 1299 \times (0.65 + 0.01 \times 55)$$

$$PF = 1299 \times 1.2 \rightarrow PF = 1558.8$$

Si calcula al 100% el nivel de confianza consideramos la sumatoria de

$\sum Fi = 70$  como el máximo valor de ajuste de complejidad se tiene:

$$PFmax = Cuenta\ Total \times (0.65 + 0.01 \times \sum Fi)$$

$$PF = 1299 \times (0.65 + 0.01 \times 70)$$

$$PFmax = 1299 \times 1.35$$

$$PFmax = 1753.65$$

Obtenido ambos valores, se calcula la funcionalidad:

$$Funcionalidad = PF / PFmax$$

$$Funcionalidad = 1558.8 / 1753.65 = 0.89 = f$$

$$\text{Funcionalidad} = 0.89 * 100\% = 89\%$$

Entonces la funcionalidad del sistema es un 89%, esto quiere decir, que el sistema tiene un 89% que funcione sin riesgos de fallo y operatividad constante y 11% aproximadamente de colapso de sistema.

### 3.4.2. Confiabilidad

La confiabilidad del sistema se define como la probabilidad de operación libre de fallos de un programa de computadora en un entorno determinado y durante un tiempo específico.

Donde se encuentra:

$P(T \leq t) = F(t)$  Probabilidad de fallas (el termino en el cual el sistema trabaja sin fallas)

$P(T \leq t) = 1 - F(t)$  Probabilidad de trabajo sin fallas (tiempo en el cual no falla el sistema)

Para calcular la confiabilidad del sistema se toma en cuenta el periodo de tiempo en el que se ejecuta y se obtiene muestras.

$$F(t) = f * e(-\mu * t)$$

Dónde:

$f$ : Funcionalidad del sistema

$\mu$ : Probabilidad de error que puede tener el sistema

$t$ : Tiempo que dura una gestion en el sistema

Considerado, un periodo de 20 días como tiempo de prueba, donde se define que de cada 10 ejecuciones se presenta una falla.

Calculando:

$$F(t) = f * e^{(-1/10 * 20)}$$

$$F(t) = 0.89 * e^{(-1/10 * 20)}$$

$$F(t) = 0.120 * 100$$

$$F(t) = 12\%$$

Reemplazando en las fórmulas de probabilidades:

$$P(T \leq t) = F(t) \rightarrow P(T \leq t) = 0.120 = 12 \%$$

$$P(T \leq t) = 1 - F(t) \rightarrow P(T \leq t) = 1 - 0.120$$

$$P(T \leq t) = 0.88 = 88\%$$

Por tanto, la confiabilidad del sistema es del 88% en un periodo de 20 días como tiempo de prueba.

### 3.4.3. Usabilidad

Es la facilidad de uso, muestra la manera de aprender a manejar el producto, para determinar la usabilidad del sistema se utiliza la siguiente ecuación:

$$FU = [(\sum Xi / n) * 100]$$

Dónde:

$Xi$ : Es la sumatoria de valores

$n$ : Es el número de preguntas

Para responder a las preguntas se debe considerar la siguiente tabla:

**Tabla Nº 3.46. Escala de Valoración de Preguntas**

ESCALA	VALOR
Muy Bueno	5
Bueno	4
Regular	3
Malo	2
Pésimo	1

**Fuente:** Elaboración Propia

**Tabla Nº 3. 47. Preguntas para Determinar la Usabilidad**

NRO	PREGUNTA	SI	NO	EVALUACIÓN
1	¿Se puede utilizar con facilidad el sistema?	5	0	1
2	¿Se puede utilizar con facilidad el sistema?	5	0	1
3	¿El sistema permitió la retroalimentación de información?	4	1	0.8
4	¿El sistema cuenta con interfaz agradable a la vista?	5	0	1
5	¿La respuesta del sistema es satisfactoria?	5	0	1
6	¿Le parecen complicadas las funciones del sistema?	3	2	0.6
7	¿Se hace difícil o dificultoso aprender a manejar el sistema?	3	2	0.6
8	¿Los resultados que proporciona el sistema facilitan el trabajo?	5	0	1
9	¿Durante el uso del sistema se produjo errores?	1	4	0.2
<b>TOTAL</b>				7.2

**Fuente:** Elaboración Propia

Se procede a calcular la usabilidad con la ecuación anterior:

$$FU = [(7.2 / 9) * 100]$$

$$FU = [0.8 * 100]$$

$$FU = 80 = 80\%$$

Por tanto, existe un 80% de comprensión o entendimiento de los usuarios con respecto a la capacidad del sistema.

#### **3.4.4. Mantenibilidad**

El mantenimiento se desarrolla para mejorar el sistema en respuesta a los nuevos requerimientos que la institución tenga.

El estándar IEE94 sugiere un índice de madurez del software (IMS) que proporciona un indicador en la estabilidad de un producto, se lo determina con la siguiente ecuación:

$$IMS : \frac{[Mt - (Fa + Fc + Fd)]}{Mt}$$

Dónde:

**Tabla N° 3.48. Valores para Determinar la Mantenibilidad**

DESCRIPCIÓN	VALOR
<i>Mt</i> = Número de módulos de la versión actual.	5
<i>Fc</i> = Número de módulos en la versión actual que se han modificado.	1
<i>Fa</i> = Número de módulos en la versión actual que se han añadido.	0
<i>Fd</i> = Número de módulos de la versión anterior que se han borrado en la versión actual.	0

**Fuente:** Elaboración Propia

Calculando el IMS:

$$IMS = [5 - (0 + 1 + 0)] / 5$$

$$IMS = 0.8 * 100 = 80\%$$

Por tanto, se puede decir que el sistema tiene un índice de estabilidad de 80%, que es la facilidad de mantenimiento, el 20% restante es el margen de error correspondiente a los cambios y modificaciones.

### 3.4.5. Portabilidad

De acuerdo a los factores de calidad, es la facilidad con que se lleva el sistema de un entorno a otro.

El sistema SIFHO por estar diseñado en un entorno de acceso distintos sistemas operativos, se mide la portabilidad en dos secciones: Portabilidad del lado del cliente y portabilidad del lado del servidor.

A nivel sistema de software, El sistema de información: Sistema de Gestión de Información de Fichas Medicas e Historial Clínico en la Especialidad de Odontología, es portable bajo los sistemas operativos de la familia Microsoft Windows y todos los sistemas operativos libres de Linux.

A nivel de base de datos se utiliza base de datos creada en MySql, la portabilidad se muestra que la base de datos puede ser migrada SQL server, Postgres y otros.

Por lo mencionado anteriormente el sistema SIFHO es portable en sus diferentes entornos tanto en Hardware y Software.

### **3.5. ESTIMACIÓN DE COSTES DE SOFTWARE**

El método de la estimación de costes de desarrollo de software, establece una relación matemática entre el esfuerzo y el tiempo de desarrollo.

#### **3.5.1. Método de Estimación COCOMO II**

La estimación de costos del sistema ha sido desarrollada bajo las KLDC (kilo-Líneas de Código). El proyecto se implementó con 10898 Líneas de Código en el lenguaje PHP.

Como:

$$**KLDC = LDC / 1000**$$

$$**KLDC = 10898 / 1000**$$

$$**KLDC = 10.898 KLDC**$$

Por lo que la evaluación del sistema ha sido considerada bajo las 5.686 KLDC. Los coeficientes que se usarán serán los valores que se detallan en la siguiente Tabla N° 3.49.

**Tabla N° 3. 49. Coeficiente del Modelo de COCOMO II**

PROYECTO DE SOFTWARE	A	B	c	d
<b>Orgánico</b>	2.40	1.05	2.50	0.38
<b>Semiacoplado</b>	3.00	1.12	2.50	0.35
<b>Empotrado</b>	3.60	1.20	2.50	0.32

**Fuente:** (S. Pressman, 2010)

A continuación, se describe las ecuaciones que permiten calcular el costo total del software:

**Tabla N° 3.50. Ecuaciones del Modelo COCOMO II**

VARIABLE	ECUACIÓN	TIPO/UNIDAD
Esfuerzo requerido por el proyecto	$E = a \times (KLDC)^b \times FAE$	Personas/Mes
Tiempo requerido por el proyecto	$T = c \times (E)d$	Meses
Número de personas requeridas para el proyecto	$NP = E / T$	Personas
Costo Total	$CT = Sueldo\ Mes \times NP \times T$	\$us

**Fuente:** (Prentice Hall, 1981)

Para hallar los valores de FAE se utiliza la Tabla N° 3.51

**Tabla N°3. 51. Cálculo de los Atributos FAE**

Atributos que afectan al coste	Valor					
	Muy Bajo	Bajo	Nominal	Alto	Muy alto	Extra Alto
<b>Atributos de Software</b>						
Fiabilidad del software	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de Datos		0,94	1,00	1,08	1,16	
Complejidad del producto	0,70	0,85	1,00	1,15	1,30	1,65
<b>Atributos de hardware</b>						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de Memoria Virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
<b>Atributos de Personal</b>						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia de la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los Programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en S.O. usando	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje de programación	1,14	1,07	1,00	0,95		
<b>Atributos del proyecto</b>						
Uso de Técnicas actuales de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	
<b>TOTAL (FAE) = 0.72</b>						

**Fuente:** Elaboración Propia

Aplicando las ecuaciones (descritas en la tabla N° 3.49.) así como los coeficientes a y c y los exponentes b y d que en nuestro caso es de tipo orgánico, será el más apropiado y a que el número de líneas de código no supera los 50 KLDC, (descritos en la Tabla N° 3.50.) y el cálculo de los atributos FAE (descrito en la Tabla N° 3.51) se tiene:

Calculando el esfuerzo:

$$E = a \times (KLDC)^b \times FAE$$

$$E = 2.4 \times (10.898)^{1.05} \times 0.72$$

$$E = 21.22 \text{ Personas /Mes}$$

Calculando el tiempo:

$$T = c \times (E)^d$$

$$T = 2.5 \times (21.22)^{0.38}$$

$$T = 7.98 \text{ Meses, Equivale a 8 meses}$$

Calculando el Personal Promedio:

$$NP = E / T$$

$$NP = 21.22 / 7.98$$

$$NP = 2.66 \text{ Equivale a 3 personas}$$

Calculando el Costo Total:

$$CT = \text{Sueldo Mes}(\text{Programadores y analistas}) \times NP \times t$$

$$CT = 500 \times 3 \times 8$$

$$CT = 12000 \text{ \$us}$$

Por tanto, en resumen se requiere 3 personas estimando un trabajo de 8 meses, por lo que el costo total del sistema es de 12000 \$us.

### 3.6. SEGURIDAD DE LA INFORMACIÓN DEL SISTEMA

#### 3.6.1. Sistema de Gestión de Seguridad de la Información ISO-27002.

La ISO-27002 evalúa y rectifica la implementación mediante el cumplimiento de normas, para que exista una continuidad de mejoramiento de un conjunto de controles que permitan reducir el riesgo de sufrir incidentes de seguridad en el funcionamiento de la institución en cuanto a la seguridad de la información, a continuación, se describe los puntos que se tomara en cuenta:

##### 3.6.1.1. Seguridad Lógica

- ✓ Gestión de Comunicaciones y Operaciones.

Los respaldos (Back-up) de la Base de Datos del sistema, se deberá realizar las precauciones de acuerdo a la siguiente Tabla N° 3.52.

**Tabla N° 3. 52. Gestión de Comunicaciones y Operaciones**

DESCRIPCIÓN	DURACIÓN
En periodos de registro de historial clínico	1 vez por semana
En periodo de registro de fichas medicas	1 vez por semana
En periodo de registro de peso estatura	1 vez por semana
En periodos de revisión odontológica	1 vez por semana
El periodo de registros de datos nuevos	1 vez al mes
Registro de diagnóstico de los pacientes	1 vez al mes

**Fuente:** Elaboración Propia

El personal que interviene en el proceso académico deberá de cambiar el password del sistema periódicamente 1 vez al mes, dando a conocer al plantel estudiantil.

En caso de ser usuario (Asistente, enfermería) se recomienda cambiar periódicamente el password.

### **3.6.1.2. Seguridad Física**

- ✓ Seguridad Física y del Entorno

Se recomienda realizar, Back-up más de 3 copias, estos sean almacenados en distintos lugares.

Los back-up de la base de datos, deberán estar protegidas y tener acceso al personal autorizado en la institución.

### **3.6.1.3. Seguridad Organizativa**

La información del sistema debe tener un nivel de protección, como ser: El manejo de los back-up de acuerdo a las fechas en el que se realizan las copias.

## **3.6.2. SEGURIDAD DEL SISTEMA**

Para la seguridad del sistema se consideran las siguientes precauciones, los cuales son:

- ✓ Autenticación de usuarios (Login)
- ✓ Manejo de privilegios y tipos de usuario en el sistema
- ✓ Manejo de Vistas

- ✓ Integridad y control de datos por url's.
- ✓ Manejo de sesiones y validación de campos
- ✓ Encriptación con el algoritmo SHA1 en las contraseñas o passwords.

El SHA (Secure Hash Algorithm, Algoritmo de Hash Seguro) es una familia de funciones hash de cifrado publicadas por el Instituto Nacional de Estándares y Tecnologías (NIST). La primera versión del algoritmo fue creada en 1993 con el nombre de SHA. Producen una salida resumen de 160 bits (20 bytes) de un mensaje que puede tener un tamaño máximo de  $2^{64}$  bits. (Wikipedia, Secure Hash Algorithm, 2017).

## **4. CONCLUSIONES Y RECOMENDACIONES**

El desarrollo del Sistema de Gestión de Información de Fichas Medicas e Historial Clínico en la Especialidad de Odontología, para el Centro de Salud de Villa Cooperativa ha sido concluido de manera satisfactoria.

### **4.1. CONCLUSIONES**

Con la presentación de este proyecto se logra centralizar la información y efectuar un seguimiento académico confiable, además de que los usuarios tendrán acceso al sistema desde cualquier computadora conectada, todo lo mencionado anteriormente coadyuva a una mejor gestión y administración de la información en el Centro de Salud de Villa Cooperativa de El Alto, y de esta forma se cumplen los objetivos planteados.

- ✓ El sistema de información fue desarrollado, haciendo el uso de la metodología RUP con la cual se realizó el análisis, diseño y desarrollo para su posterior implementación.
- ✓ Se tiene una información más clara sobre los historiales clínicos de los pacientes.
- ✓ El sistema implementa políticas de seguridad utilizando roles de usuario, activando sesión a cada ingreso al sistema.
- ✓ Cuenta con una base de datos con información concreta y precisa para la gestión de fichas médicas e historial clínico.
- ✓ Se desarrolló un módulo de revisión odontológica dental que permite acceder de forma rápida y confiable a la información.
- ✓ Cuenta con reportes necesarios para la toma de decisiones.
- ✓ Se concluye exitosamente con los siguientes módulos:
  - Módulo de Administración de Usuarios
  - Módulo Asistente

- Módulo Enfermera
- Módulo Director.
- Modulo Doctor.
- Módulo Reportes

Finalmente se concluye que en el desarrollo del “SISTEMA DE GESTIÓN DE INFORMACIÓN DE FICHAS MEDICAS E HISTORIAL CLÍNICO EN LA ESPECIALIDAD DE ODONTOLOGÍA”, se cumplieron todos los objetivos; por lo que es un aporte tecnológico oportuno, se reduce el tiempo de registro, consulta y búsqueda de la información perteneciente al Centro de Salud. Cabe recalcar que el manejo de esta información se realiza de forma segura, rápida, precisa y confiable, aplicando con éxito las normas de Calidad y las Herramientas de Programación establecidas.

#### **4.2. RECOMENDACIONES**

Con la implementación del sistema, se recomienda lo siguiente

- ❖ Se sugiere considerar, con respecto al software del sistema, la implementación de módulos para la atención de pediatría, ginecología y farmacia.
- ❖ Capacitar a los nuevos usuarios, administradores para poder realizar operaciones en el sistema y así poder administrarlo correctamente.
- ❖ Resguardar la información, realizando copias de seguridad periódicamente de la base de datos.
- ❖ Para posteriores versiones del sistema se recomienda que se desarrollen Sistemas de Información Integrada, así de esta manera se centralice y mejore la administración de la información

## **BIBLIOGRAFÍA**

- Gesvin Romero Romero, (2004). "UML con Rational Rose" Primera Edición.
- Oiver Andres Pérez, (2011). "cuatro enfoques metodología para el desarrollo de software" facultad de ingeniería UNIMINUTO.
- IanGilfillan, (2003). "La Biblia de MySql" Primera Edición.
- Juan Pavón, (2012). "Aplicaciones web y Sistemas web" facultad de informática
- Prendice Hall, (2002). "servidor apache"
- Barcelona, (2018). "Cdelgniter Rocks"
- Isaac Ojeda, (2012). "Manual jQuery" Instituto Tecnológico de Chihuahua II
- Juan Gauchat, (2012). "El gran libro de HTML5, CSS y Javascript"
- Juan Fuentes, (2009). "AJAX, Fundamentos y Aplicaciones".
- Miguel Muñoz, (2018). "Bootstrap".
- Antonio Zimbrón & Mirella Feingold, (1993). "Odontología Preventiva"
- Moreno Garcia & Holgado Garcia,(2018), "UML.Unified Modeling Language"
- Andrade de Oliveira, (2014). "ABNT NBR ISO/IEC 25000"
- PRESSMAN R. (2002). Ingeniería de Software Un Enfoque Práctico. Quinta Edición (QUITA EDICIÓN ed.). MÉXICO: MC GRAW HILL.
- Pressman R. (2002). Ingeniería de Software, Un Enfoque Práctico. Quinta Edición. Mexico: Mc Graw Hill.
- Pressman Roger S. (2010). Ingeniería del software un Enfoque Práctico (7ma. ed.). México: Mc Graw Hill.

## **DISPONIBLE EN**

- COCOMO. (2017). Obtenido de <https://es.m.wikipedia.org/wiki/COCOMO>
- Consortium, W. W. (Agosto de 2017). Guía Breve de CSS. Obtenido de <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>.
- EcuRed. (12 de Agosto de 2017). Definición de Bootstrap. Obtenido de <https://www.ecured.cu/Bootstrap>

- EcuRed. (12 de Agosto de 2017). Definición de FrameWork. Obtenido de <https://www.ecured.cu/Framework>
- Garcia L. M. (2012). Modelos de Estimación: Un pococ sobre COCOMO II. Obtenido de <http://unpocodejava.wordpress.com/2012/02/07/modelos-de-estimaciónun-poco-sobre-cocomo-ii/>
- Julián Pérez P. & Ana G. (2008). Sistema de Información. Obtenido de <http://definicion.de/sistema-de-informacion>.
- Evaluación de Software. (2017). *Norma de Evaluacion ISO/IEC 9126*. Obtenido de <http://actividadreconocimiento-301569-8.blogspot.com/2013/03/norma-de-evaluación-isoiec-9126.html>
- Garcia L. M. (2012). *Modelos de Estimación: Un pococ sobre COCOMO II*. Obtenido de <http://unpocodejava.wordpress.com/2012/02/07/modelos-de-estimación-un-poco-sobre-cocomo-ii/>
- Julián Pérez P. & Ana G. (2008). *Sistema de Información*. Obtenido de <http://definicion.de/sistema-de-informacion>
- Julián Pérez Porto. (20 de Marzo de 2008). *Definición de Sistema*. Obtenido de <http://definicion.de/sistema>.
- Leandro Alegsa. (20 de Julio de 2017). *Alegsa*. Obtenido de Desarrollo de Sistema : <http://www.alegsa.com.ar/Dic/sistema.php>
- Mater Magazine. (Agosto de 2016). *Definición de Analisis*. Obtenido de <https://mastermagazine.info/termino/3840.php>
- Miguel Angel A. (19 de septiembre de 2012). *Manual de JQuery*. Obtenido de <http://www.desarrolloweb.com/manuales/manual-jquery.html>
- Model(Español), T. P. (Agosto de 2017). *UWE-UML-Based Web Enrineering*. Obtenido de <http://uwe.pst.ifi.lmu.de/teachingTutorialPresentationSpanish.html>
- Novoa V. D. (Agosto de 2017). *Gestión de Información*. Obtenido de <http://www.monografias.com>
- Quiroga. (23 de Marzo de 2015). *Metodologia UWE UML (UML Based Engineering)*. Obtenido de <http://www.scribd.com>