

UNIVERSIDAD TÉCNICA DEL NORTE



Facultad de Ingeniería en Ciencias Aplicadas
Carrera de Ingeniería en Sistemas Computacionales

“BENCHMARKING DE METODOLOGÍAS WEB”
“SISTEMA DE CONTROL ODONTOLÓGICO”

Trabajo de grado previo a la obtención del Título de Ingeniero en Sistemas
Computacionales

Autor:

Edison Andrés Fures Quilumbango

Director:

Ing. MacArthur Ortega

Ibarra – Ecuador

2017



UNIVERSIDAD TÉCNICA DEL NORTE

BIBLIOTECA UNIVERSITARIA

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

1. IDENTIFICACIÓN DE LA OBRA

La UNIVERSIDAD TÉCNICA DEL NORTE dentro del proyecto Repositorio Digital institucional, determina la necesidad de disponer los textos completos de forma digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la universidad.

Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente investigación:

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD	100314973-7
APELLIDOS Y NOMBRES	FUERES QUILUMBANGO EDISON ANDRÉS
DIRECCIÓN	ARAHUALPA Y VALLE DEL AMANECER – OTAVALO
EMAIL	eafueres@utn.edu.ec
TELÉFONO MÓVIL	0988212966
DATOS DE LA OBRA	
TÍTULO	“BENCHMARKING DE METODOLOGÍAS WEB. SISTEMA DE CONTROL ODONTOLÓGICO.”
AUTOR	FUERES QUILUMBANGO EDISON ANDRÉS
FECHA	DICIEMBRE DEL 2017
PROGRAMA	<input checked="" type="checkbox"/> PREGRADO <input type="checkbox"/> POSTGRADO
TÍTULO POR EL QUE OPTA	INGENIERÍA EN SISTEMAS COMPUTACIONALES
DIRECTORA	ING. MACARTHUR ORTEGA

2. AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Fueres Quilumbango Edison Andrés con cédula de identidad Nro. 100314973-7, en calidad de autor y titular de los derechos patrimoniales del proyecto de grado descrito anteriormente, hago entrega del ejemplar respectivo en forma digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y el uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión, en concordancia con la Ley de Educación Superior Artículo 144.

3. CONSTANCIAS

El autor (es) manifiesta (n) que la obra objeto de la presente autorización es original y se la desarrolló, sin violar derechos de autor de terceros, por lo tanto, la obra es original y que es (son) el (los) titular (es) de los derechos patrimoniales, por lo que asume (n) la responsabilidad sobre el contenido de la misma y saldrá (n) en defensa de la Universidad en caso de reclamación por parte de terceros.



Firma

Nombre: Fueres Quilumbango Edison Andrés

Cédula: 100314973-7

Ibarra, diciembre del 2017



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE INVESTIGACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE

Yo, Fueses Quilumbango Edison Andrés, con cédula de identidad Nro. 100314973-7, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la ley de propiedad intelectual del Ecuador, artículo 4, 5 y 6, en calidad de autor del proyecto de grado denominado: **“BENCHMARKING DE METODOLOGÍAS WEB.”** con el aplicativo **“SISTEMA DE CONTROL ODONTOLÓGICO”** que ha sido desarrollado para optar por el título de Ingeniero en Sistemas Computacionales, en la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Técnica del Norte.

.....
Firma

Nombre: Fueses Quilumbango Edison Andrés

Cédula: 100314973-7

Ibarra, diciembre del 2017



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN DIRECTOR DE TESIS

Certifico que la tesis **“BENCHMARKING DE METODOLOGÍAS WEB.”** con el aplicativo **“SISTEMA DE CONTROL ODONTOLÓGICO.”**, ha sido realizada con interés profesional y responsabilidad por el señor: **Fueres Quilumbango Edison Andrés**, portador de la cédula de identidad Nro. 100314973-7; previo a la obtención del Título de Ingeniero en Sistemas Computacionales.

A handwritten signature in blue ink, appearing to read "MacArthur Ortega", is written over a horizontal dotted line. The signature is stylized and somewhat abstract.

Ing. MacArthur Ortega
DIRECTOR DE GRADO



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DECLARACIÓN

Yo, Fures Quilumbango Edison Andrés, declaro bajo juramento que el trabajo aquí descrito es de mi autoría, que no ha sido previamente presentado para ningún grado, ni calificación profesional; y que he consultado las referencias bibliográficas que se incluyen en este documento.

Firma

Nombre: Fures Quilumbango Edison Andrés

Cédula: 100314973-7

Ibarra, diciembre del 2017



CONSULTORIO ODONTOLÓGICO

DR. LUIS ANDRANGO NARVAEZ

Ortodoncia Fija y Cirugía Dentofacial

CONSULTORIO. Otavalo Av. 31 de Octubre y Abdón Calderón

2do piso (Junto a la farmacia Cruz Azul) Telf. 2923-708 Cel. 0985431832

CERTIFICACIÓN

Otavalo, 16 de diciembre del 2017

Señores
UNIVERSIDAD TÉCNICA DEL NORTE
Presente,

De mi consideración,

Siendo auspiciante del proyecto de Tesis del Sr. EDISON ANDRÉS FUERES QUILUMBANGO, con C.I. 100314973 quien desarrolló su trabajo de grado con el tema "BENCHMARKING DE METODOLOGÍAS WEB" aplicativo "SISTEMA DE CONTROL ODONTOLÓGICO" me es grato informarle que el proceso de implementación del software ha sido superado satisfactoriamente, por lo que extendiendo el presente documento como constancia de aceptación y finalización del proyecto.

Autorizo al Sr. FUERES QUILUMBANGO EDISON ANDRÉS, hacer uso del documento para los fines pertinentes en la UNIVERSIDAD TÉCNICA DEL NORTE.

Atentamente,

Dr. Luis Andrango Narváez
C.I. 100152911-2

Dr. Luis Andrango N.
Ortodoncia Fija y Cirugía Dentofacial



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

Esta tesis dedico primeramente a **Dios** quien es mi inspiración divina y la fuente de todos mis conocimientos, a Él la gloria y honra.

Mi dedicatoria especial a mis padres **Ángel Fures** y **Juana Quilumbango**, quienes han sido un apoyo durante mi formación profesional y mi ayuda incondicional y en todo el transcurso de la vida. A **Heidy** y **Alexander** mis hermanos queridos quienes también me motivaron a alcanzar esta anhelada victoria, **familia**, este triunfo es para ustedes.

Deseo dedicar este proyecto de titulación a mis amigos y hermanos de la fe en Otavalo, La Compañía, San Pablo del Lago y Mayorista Ambato, quienes, con sus consejos y oraciones han hecho que mi persona alcance este deseado éxito, a ustedes pastores, hermanos y amigos de la IEAN JESÚS, Dios les bendiga.

Dedico también este trabajo de grado a mis amigos, compañeros y docentes de la carrera de Ingeniería en Sistemas Computacionales de la prestigiosa Universidad Técnica del Norte.

Edison Fures



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

En primer lugar, quiero agradecer a **Dios**, quien me ha dado la vida, me apoyó en todo instante y su ayuda incondicional ha sido, es y seguirá siendo visible en todas las etapas de mi vida, un sincero y especial agradecimiento a ti Jesús.

Agradezco infinitamente a mis **Padres** quienes se esforzaron apoyándome en todo el transcurso de mi formación profesional, a mis **hermanos** y demás **familiares** que siempre estuvieron dispuestos a ayudarme de una u otra manera.

Un agradecimiento especial a mi asesor de tesis **Ing. MacArthur Ortega**, por toda la ayuda y su excelente asesoría en el desarrollo del proyecto de titulación.

Agradezco también a **Lorena Chandi**, compañera, amiga, quien forma parte de mi vida y ha sido esencial para lograr este éxito, que Dios nos ayude a seguir obteniendo aún mayores triunfos.

Extiendo mi agradecimiento a todos quienes conforman la **IEAN JESÚS**, los cuales con sus plegarias han estado apoyándome incondicionalmente.

Quiero agradecer a todos quienes conforman la carrera de Ingeniería en Sistemas Computacionales de la Universidad Técnica del Norte, gracias por abrirme las puertas y brindarme la oportunidad de superarme académicamente, a los docentes que sembraron los mejores conocimientos en mi etapa estudiantil, al personal administrativo que permitieron que pueda poner en práctica el conocimiento alcanzado, a mis compañeros de clases, gracias por toda su ayuda brindada. Finalmente agradezco a todos mis **amigos** y demás conocidos, mi sincero agradecimiento a todos ustedes, Dios les bendiga.

RESUMEN

La investigación se basa en el estudio comparativo de determinadas metodologías web, las cuales mediante un análisis exhaustivo se seleccionará la metodología mejor adaptable en el desarrollo de aplicaciones web.

El desarrollo del presente análisis comparativo ha sido clasificado de la siguiente manera:

Capítulo 1

El capítulo uno presenta la introducción, antecedentes, situación actual, prospectiva, planteamiento del problema y los objetivos por los cuales es necesario desarrollar esta investigación, en base a un alcance y una justificación bien estructurada.

Capítulo 2

El capítulo dos establece el marco teórico de las principales metodologías web que permitirán realizar este estudio, a su vez la conceptualización básica de las herramientas y tecnologías que se utilizarán en el desarrollo de la aplicación de control odontológica.

Capítulo 3

El capítulo tres contiene el análisis comparativo de las metodologías web estudiadas y su evaluación mediante el modelo de requisitos y la norma ISO/IEC 9126, seleccionando de esta manera la metodología más adaptable en el desarrollo de aplicaciones web.

Capítulo 4

El capítulo cuatro utiliza la metodología destacada en la investigación, para el desarrollo de la aplicación de control odontológica.

Capítulo 5

El capítulo cinco presenta el software odontológico según los lineamientos establecidos en la metodología seleccionada.

Capítulo 6

El capítulo seis presenta los resultados que generó el uso de la metodología ganadora en la construcción de la aplicación web odontológica.

Palabras Clave: metodologías web, ingeniería web, estudio, investigación, análisis comparativo.

ABSTRACT

The research is based on the comparative study of certain web methodologies, which through a thorough analysis will select the best adaptable methodology in the development of web applications.

The development of the present comparative analysis has been classified as follows:

Chapter 1

Chapter one presents the introduction, background, current situation, prospective, approach to the problem and the objectives for which it is necessary to develop this research, based on a well-structured scope and justification.

Chapter 2

Chapter two establishes the theoretical framework of the main web methodologies that will allow this study, in turn, the basic conceptualization of the tools and technologies that will be used in the development of the dental control application.

Chapter 3

Chapter three contains the comparative analysis of the web methodologies studied and their evaluation through the requirements model and the ISO / IEC 9126 standard, thus selecting the most adaptable methodology in the development of web applications.

Chapter 4

Chapter four uses the methodology highlighted in the research for the development of the dental web application.

Chapter 5

Chapter five presents the dental software according to the guidelines established in the selected methodology.

Chapter 6

Chapter six presents the results generated using the winning methodology in the construction of the dental control application.

Keywords: web methodologies, web engineering, study, research, comparative analysis.

ÍNDICE DE CONTENIDO

AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	II
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE INVESTIGACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	IV
CERTIFICACIÓN DIRECTOR DE TESIS.....	V
DECLARACIÓN	VI
DEDICATORIA.....	VIII
AGRADECIMIENTO.....	IX
RESUMEN.....	X
ABSTRACT	XII
CAPÍTULO 1	1
Introducción.....	1
1.1 Antecedentes.	1
1.2 Situación actual.....	2
1.3 Prospectiva.....	3
1.4 Planteamiento del problema.	3
1.5 Objetivos.....	4
1.5.1 Objetivo general.	4
1.5.2 Objetivos Específicos.....	4
1.6 Alcance.....	4
1.7 Funcionalidad y módulos por desarrollarse.....	5
1.8 Justificación.....	6
1.9 Impactos.....	7
1.9.1 Impacto Tecnológico.....	7
1.9.2 Impacto Económico.....	7
1.9.3 Impacto Social.....	7
CAPÍTULO 2	8
Marco teórico.....	8

2.1	Introducción	8
2.2	Definiciones	8
2.2.1	Ingeniería Web.....	8
2.2.2	Metodologías web.....	8
2.2.3	HDM – Hipertext design model.....	10
2.2.4	RMM –Relationship Management Methodology.....	11
2.2.5	OOHDM – Object Oriented Hypermedia Design Methodology	13
2.2.6	WSDM - Web site design method.....	15
2.2.7	HDM-Lite.....	17
2.2.8	UWE – UML-Based Web Engineering.....	17
2.2.9	WEBML – Web Modelling Language.....	19
2.2.10	IFML – Interaction flow modeling language (IFML).....	20
2.3	Conceptos tecnológicos.....	21
2.3.1	Aplicación web.....	21
2.3.2	Servidor de aplicaciones.....	22
2.4	Herramientas.....	22
2.4.1	PostgreSQL.....	22
2.4.2	Java EE (Java Enterprise Edition).....	23
2.4.3	JavaServer Faces (JSF).....	23
2.4.4	PrimeFaces.....	23
2.4.5	OpenShift.....	23
CAPÍTULO 3	25
Análisis comparativo de metodologías		25
3.1	Introducción.....	25
3.2	Aplicación de la encuesta.....	25
3.2.1	Encuesta.....	25
3.2.2	Población.....	25
3.2.3	Encuesta aplicada a la población.....	26
3.2.4	Resultado de la encuesta aplicada a la población.....	34

3.2.5 Desarrollo de prototipos.....	34
3.2.6 Prototipo de la metodología OOHDM.....	34
3.2.7 Prototipo de la metodología WSDM.....	39
3.2.8 Prototipo de la metodología UWE.....	42
3.2.9 Resultados del desarrollo de prototipos.....	45
3.3 Análisis Comparativo.....	46
3.3.1 Ponderación y escalamiento de Likert.....	46
3.3.2 Modelo de evaluación en base a requisitos.....	47
3.3.3 ISO/IEC 9126.....	49
3.3.4 Criterios de valoración.....	52
3.3.5 Comparativa de metodologías web.....	52
3.3.6 Evaluación mediante los parámetros del modelo de requisitos.....	54
3.3.7 Interpretación de resultados del modelo de requisitos.....	59
3.3.8 Valoración porcentual del modelo de requisitos.....	61
3.3.9 Resultados del análisis comparativo del modelo de requisitos.....	62
3.3.10 Evaluación del parámetro Eficiencia.....	62
3.3.11 Interpretación de resultados del parámetro Eficiencia.....	63
3.3.12 Valoración porcentual del parámetro Eficiencia.....	64
3.3.13 Resultados del análisis comparativo del parámetro Eficiencia.....	65
3.3.14 Evaluación del parámetro Usabilidad.....	65
3.3.15 Interpretación de resultados del parámetro Usabilidad.....	66
3.3.16 Valoración porcentual del parámetro Usabilidad.....	67
3.3.17 Resultados del análisis comparativo del parámetro Usabilidad.....	68
3.4 Interpretación de resultados finales.....	68
3.5 Análisis de resultados finales.....	70
CAPÍTULO 4	71
Desarrollo de la metodología UWE (UML-Based Web Engineering).....	71
4.1 Introducción.....	71
4.2 Modelo de Casos de Uso.....	72

4.3 Modelo de Contenido.....	75
4.4 Modelo de Navegación.....	76
4.5 Modelo de Presentación.....	76
4.6 Modelo de Proceso.....	80
4.6.1 Modelo de Estructura de proceso.....	80
4.6.2 Modelo de Flujo de proceso.....	81
CAPÍTULO 5	82
Construcción de la aplicación web odontológica.....	82
5.1 Introducción.....	82
5.2 Arquitectura de software.....	82
5.3 Ambiente de implementación.....	83
5.4 Diseño de la base de datos.....	84
5.5 Software Odontológico.....	84
CAPÍTULO 6	91
Resultados.....	91
6.3 Control de calidad.....	91
6.3.1 Aspecto Funcional.....	91
6.3.2 Aspecto Técnico.....	91
6.4 Análisis de impactos.....	92
CONCLUSIONES Y RECOMENDACIONES	95
Conclusiones.....	95
Recomendaciones.....	95
GLOSARIO DE TÉRMINOS	96
REFERENCIAS BIBLIOGRÁFICAS	97

ÍNDICE DE FIGURAS

Fig. 1: Esquema de funcionamiento de la aplicación.....	5
Fig. 2: Módulos del sistema.....	6
Fig. 3: Principales metodologías de modelado web.....	9
Fig. 4: Notación HDM.....	11
Fig. 5: Notación de RMM.....	13
Fig. 6: Diagrama de fases de WSDM.....	16
Fig. 7: Componentes de un sitio Web con WebML.....	19
Fig. 8: OpenShift y sus servicios en la nube.....	24
Fig. 9: Resultados de la pregunta 1	26
Fig. 10: Resultados de la pregunta 2	28
Fig. 11: Resultados de la pregunta 3	29
Fig. 12: Resultados de la pregunta 4	30
Fig. 13: Resultados de la pregunta 5	31
Fig. 14: Resultados de la pregunta 6	33
Fig. 15: Diagrama de casos de uso.....	36
Fig. 16: UID caso de uso Autenticar en el sistema.....	36
Fig. 17: UID caso de uso Administrar Usuarios.....	36
Fig. 18: UID caso de uso Administrar Pacientes.....	36
Fig. 19: UID caso de uso Administrar Médicos.....	37
Fig. 20: UID caso de uso Administrar Pagos o procedimientos.....	37
Fig. 21: UID caso de uso Generar reportes.....	37
Fig. 22: Diagrama de clases.....	37
Fig. 23: Diagrama de secuencias	38
Fig. 24: Diagrama navegacional	38
Fig. 25: ADV presentación de la aplicación web.....	39
Fig. 26: Modelo conceptual	40
Fig. 27: Diagrama navegacional	41
Fig. 28: Diagrama de Implementación	41
Fig. 29: Diagrama de casos de uso.....	43
Fig. 30: Diagrama de clases.....	43
Fig. 31: Diagrama navegacional	44
Fig. 32: Diagrama de presentación	44
Fig. 33: Diagrama de estructura de proceso.....	45
Fig. 34: Diagrama de flujo de proceso	45
Fig. 35: Atributos de la norma ISO/IEC 9126.....	49

Fig. 36: Valoración porcentual del modelo de requisitos.....	62
Fig. 37: Valoración porcentual del parámetro eficiencia.....	65
Fig. 38: Valoración porcentual del parámetro Usabilidad.....	68
Fig. 39: Resultados del análisis comparativo de metodologías web	70
Fig. 40: Diagrama de casos de uso.....	73
Fig. 41: Casos de uso Usuario.....	73
Fig. 42: Casos de uso Paciente.....	74
Fig. 43: Casos de uso Medico.....	74
Fig. 44: Casos de uso Procedimientos	75
Fig. 45: Diagrama de clases.....	75
Fig. 46: Diagrama de navegación.....	76
Fig. 47: Diagrama de presentación Principal	77
Fig. 48: Diagrama de presentación Login.....	77
Fig. 49: Diagrama de presentación Estándar.....	78
Fig. 50: Diagrama de presentación Usuarios.....	78
Fig. 51: Diagrama de presentación Pacientes.....	79
Fig. 52: Diagrama de presentación Médicos.....	79
Fig. 53: Diagrama de presentación Procedimientos	80
Fig. 54: Diagrama de Estructura de Proceso	81
Fig. 55: Diagrama de Flujo de Proceso	81
Fig. 56: Arquitectura de software.....	83
Fig. 57: Diseño físico de la base de datos.....	84
Fig. 58: Pantalla principal de la aplicación odontológica	85
Fig. 59: Pantalla de control de acceso al sistema	85
Fig. 60: Menús de acceso.....	86
Fig. 61: Pantalla de pacientes.....	86
Fig. 62: Pantalla de control odontológico	87
Fig. 63: Tratamientos en el control odontológico.....	87
Fig. 64: Pantalla de especialidades.....	88
Fig. 65: Pantalla de médicos	88
Fig. 66: Pantalla de usuarios.....	89
Fig. 67: Pantalla de usuarios.....	89
Fig. 68: Reporte de pacientes	90
Fig. 69: Representación gráfica del análisis de impactos	93
Fig. 70: Representación gráfica del uso de la metodología web.....	93

ÍNDICE DE TABLAS

Tabla 1: Etapas y sub etapas de OOHDM.....	14
Tabla 2: Resultados de la pregunta 1	26
Tabla 3: Resultados resumen de la pregunta 1.....	27
Tabla 4: Resultados de la pregunta 2	27
Tabla 5: Resultados resumen de la pregunta 2.....	28
Tabla 6: Resultados de la pregunta 3	29
Tabla 7: Resultados resumen de la pregunta 3.....	29
Tabla 8: Resultados de la pregunta 4	30
Tabla 9: Resultados resumen de la pregunta 4.....	30
Tabla 10: Resultados de la pregunta 5.....	31
Tabla 11: Resultados resumen de la pregunta 5.....	32
Tabla 12: Resultados de la pregunta 6.....	32
Tabla 13: Resultados resumen de la pregunta 6.....	33
Tabla 14: Especificación de escenario de acceso al sistema.....	35
Tabla 15: Administración del sistema	35
Tabla 16: Reportes del sistema	35
Tabla 17: Clasificación de usuario.....	40
Tabla 18: Representación de la escala de Likert	47
Tabla 19: Requisitos deseables para la evaluación de metodologías.....	47
Tabla 20: Comparativa de las características de la norma ISO/IEC 9126.....	50
Tabla 21: Parámetros de evaluación	51
Tabla 22: Indicadores de Eficiencia	51
Tabla 23: Indicadores de Usabilidad.....	51
Tabla 24: Escala de valoración para la evaluación de metodologías.....	52
Tabla 25: Características generales de las metodologías	53
Tabla 26: Ventajas y desventajas de las metodologías web propuestas.....	54
Tabla 27: Matriz de análisis de metodologías.....	55
Tabla 28: Valoración cualitativa de metodologías	57
Tabla 29: Valoración cuantitativa de metodologías	58
Tabla 30: Valoración de metodologías	61
Tabla 31: Matriz de análisis del parámetro Eficiencia.....	62
Tabla 32: Valoración cualitativa de Eficiencia.....	63
Tabla 33: Valoración cuantitativa de Eficiencia	63
Tabla 34: Valoración de metodologías	64
Tabla 35: Matriz de análisis del parámetro Usabilidad	65

Tabla 36: Valoración cualitativa de Usabilidad	66
Tabla 37: Valoración cuantitativa de Usabilidad.....	66
Tabla 38: Valoración de metodologías	68
Tabla 39: Resultados del análisis comparativo de metodologías web.....	69
Tabla 40: Valoración de resultados	69
Tabla 41: Resultados generales del análisis de metodologías web	69
Tabla 42: Requerimientos de la Software Odontológico	71
Tabla 43: Herramientas para el desarrollo de la aplicación web.....	82
Tabla 44: Ambiente de implementación	84
Tabla 45: Resultados de los aspectos funcionales	91
Tabla 46: Resultados de los aspectos técnicos.....	92
Tabla 47: Resultados del análisis de impactos.....	92

CAPÍTULO 1

Introducción

1.1 Antecedentes.

La navegación en la web se ha convertido en una acción cotidiana del ser humano, todo tipo de usuarios están inmersos en el uso de sitios que se encuentran alojados en el ciberespacio, apoyado por el gran avance de la tecnología informática, herramientas adaptables para múltiples dispositivos electrónicos que permitan una conexión inmediata al internet y el constante desarrollo de las redes de comunicación que conceden un acceso global e instantáneo.

Quien accede al mundo de la web encuentra un espacio infinito de sitios que presentan información, servicios y aplicaciones buscando satisfacer las necesidades de los usuarios, pero se presentan distintas inquietudes como: ¿Cuáles son las garantías de utilizar estas webs? ¿Son espacios webs que han llegado a alcanzar efectividad, impacto y aceptación en los consumidores? o ¿Son aplicaciones web que han pasado por un proceso de estudio y diseño estructural? Llegando a un resultado desfavorable que en la mayoría de las situaciones no se establece un estudio previo, un diseño estructural o planificación para el desarrollo de aplicaciones web (Rodríguez, 2010).

Por tal motivo es necesario hacer el uso de herramientas y metodologías específicas que permitan el desarrollo de aplicaciones informáticas, utilizando métodos formales de diseño e implementación, estableciendo de esta manera que los sistemas informáticos sean totalmente estructurados y funcionales. Aunque actualmente se han definido diversos estándares metodológicos sean estos tradicionales, ágiles o híbridos, no se han destacado otro tipo de metodologías que estén centradas directamente en la creación de aplicaciones web, esto debido al desconocimiento de los métodos y la mínima utilización de los mismos, han hecho que estas técnicas metodológicas pasen desapercibidas al momento de crear el software.

Debido a la existencia de un indeterminado número de metodologías web, y la escasa investigación sobre el mejor modelo metodológico, es preciso realizar un estudio comparativo de estos métodos de estructuración, planificación y control del proceso de desarrollo de las aplicaciones.

La asistencia médica odontológica se ha transformado en un servicio imprescindible para la sociedad, tal necesidad es visible al hacer uso de la misma. El servicio dental trabaja con un sin número de clientes que generan gran cantidad de datos, que para una mejor atención

debe ser manipulada y asistida por algún tipo de aplicación informática de control, de lo contrario se tendrá como resultado una prestación de servicio deficiente debido al control de los grandes volúmenes de información. Muchos son los casos de instituciones de atención odontológica donde su información es manejada de manera personal mediante el uso de ficheros, carpetas y agendas físicas, en el cual la información es perceptible que no es respaldada, con un alto riesgo de ser extraviada, llegando incluso a causar el temor de ser mal manipulada.

Uno de los casos descrito anteriormente es el Consultorio Odontológico del Doctor Luis Andrango ubicado en la ciudad de Otavalo (Imbabura – Ecuador), el cual viene prestando sus servicios alrededor de 25 años. Debido a la masiva acogida por la calidad de servicio prestado del Medico Odontólogo y el gran almacenamiento de información física se hace necesaria la implementación de una aplicación informática web de control odontológica que garantice la calidad, respaldo y segura manipulación de la información de este centro de salud odontológico.

1.2 Situación actual

Actualmente se ha definido infinidad de sitios web que hace que el usuario se vuelva totalmente dependiente de sus servicios, sean estos, sitios sencillos como un buscador, plataformas y aulas virtuales o aplicaciones robustas que permitan realizar transacciones de comercio electrónico o a su vez procesos financieros con la interacción de servicios bancarios. Aunque este tipo de aplicaciones son utilizadas constantemente, a ciencia cierta se puede argumentar que, solo un pequeño grupo de sitios son generados en bases sólidas, de acuerdo con la función de presentación y propósito, y que apenas un grupo reducido de este permite aprovechar al máximo el recurso web (Rodríguez, 2010).

El problema deriva principalmente a que varios programadores no están utilizando metodologías de desarrollo, pues si bien existen determinados estándares metodológicos, muchas aplicativos son construidos sin un análisis previo y un control de proceso, siendo aplicaciones informáticas improvisadas y sin la utilización de métodos estructurados para la elaboración de sistemas.

A esto se añade que existen grupos de desarrolladores que aplican en el desarrollo de software cierta metodología general, que no cumple estándares sistemáticos para la creación de aplicaciones de calidad en la World Wide Web, esto debido a que en nuestro medio no se ha establecido el uso de alguna metodología web, siendo imprescindible realizar un estudio de una solución metodológica para el desarrollo de las aplicaciones web.

Actualmente el manejo y control de información de los clientes y pagos por servicio viene siendo controlado de manera manual, siendo almacenados en ficheros y carpetas con documentos físicos, en ocasiones material deteriorado y sin ningún respaldo de la misma, generando así problemas de pérdida de información, falencias en el registro de datos o a su vez el riesgo de que estos datos sean manipulados incorrectamente y sin seguridad en la información.

1.3 Prospectiva.

Al realizar el estudio comparativo de metodologías se planea documentar una información clara y concisa para el personal interesado en el desarrollo de aplicaciones informáticas mediante la utilización de métodos web, a su vez se establecerá parámetros de análisis y comparación que evalúen los estándares metodológicos para desarrollar aplicaciones de éxito.

Los argumentos de la comparativa permitirán realizar un estudio que fundamente la inclinación hacia el uso de una metodología específica, y de esta manera se ayudará a los diseñadores de software a seleccionar un método eficiente que aporte en el análisis, diseño e implementación de las aplicaciones de acuerdo con los lineamientos de la metodología destacada.

Un caso práctico de uso de metodología web será el desarrollo del sistema de control odontológico, donde se tiene previsto demostrar la efectividad de utilizar este tipo de herramienta metodológica en la creación de la aplicación.

Al implementar el sistema de control odontológico se planea automatizar el registro de información, pago de servicios y el control de las atenciones médicas dentales que tienen los pacientes, garantizando de esta manera la seguridad de la información.

El beneficio será directamente para médicos odontólogos y personal administrativo de la institución que tendrán un control automatizado de atenciones control y pagos de los pacientes, como también la seguridad de mantener la información íntegra, segura y confiable.

1.4 Planteamiento del problema.

¿Cómo influenciará el uso de una metodología web adecuada en el desarrollo de las aplicaciones informáticas?

Mediante el análisis comparativo de metodologías se seleccionará la herramienta metodológica más adecuada para el control del desarrollo de aplicaciones web, lo cual permitirá construir software de calidad, debido a que los modelos mencionados hacen un enfoque detallado en el análisis de requerimientos y en los procesos de modelado conceptual

y navegacional, analizando principalmente la interfaz abstracta del software desarrollado, garantizando de esta manera la estructuración formal y la correcta implementación de la aplicación informática.

¿Cómo ayudará la selección de una metodología web adecuada en el desarrollo de la aplicación odontológica?

El uso de una metodología adecuada ayudará en la construcción de una buena aplicación web, cumpliendo con los requerimientos detallados en la fase de análisis previo, realizando de igual manera el análisis del modelado conceptual, navegacional e interfaz abstracta, presentando un software eficiente, robusto y con una interfaz muy visual y aceptable por el usuario final, entregando de esta manera una herramienta informática muy útil y funcional, la cual será de gran utilidad para el sector médico odontológico.

1.5 Objetivos.

1.5.1 Objetivo general.

Analizar el benchmarking de metodologías web para el desarrollo de aplicaciones informáticas.

1.5.2 Objetivos Específicos.

- Estudiar determinadas metodologías que se adapten al desarrollo de aplicaciones web.
- Comparar las herramientas metodológicas que alcancen un proceso aceptable en la creación de aplicaciones web.
- Implementar una metodología eficiente para el desarrollo del sistema de control odontológico.
- Desarrollar los resultados del uso de una metodología web en la construcción del software odontológico.

1.6 Alcance.

La comparativa realizada analiza el uso de metodologías, seleccionando del amplio campo de la ingeniería web tres herramientas metodológicas de las cuales se va a seleccionar la más conveniente en conceptos de:

- Características,
- Ventajas,
- Desventajas.

Para el desarrollo de la investigación se seleccionará tres metodologías que entrarán en el proceso de comparación. Posterior al análisis y estudio de metodologías web, se desarrollará el sistema de control de odontológico donde se utilizará el modelo web más adecuado de acuerdo con los resultados obtenidos en la investigación.

1.7 Funcionalidad y módulos por desarrollarse

El sistema de control odontológico será una aplicación que estará alojada en un servidor en la web, el aplicativo será desarrollado en herramientas de software libre como:

- Base de Datos PostgreSQL 9.3
- JavaServer Faces (JSF)
- PrimeFaces
- JasperReports

A continuación, en la Figura 1. Se identifica el esquema del funcionamiento de la aplicación odontológica.

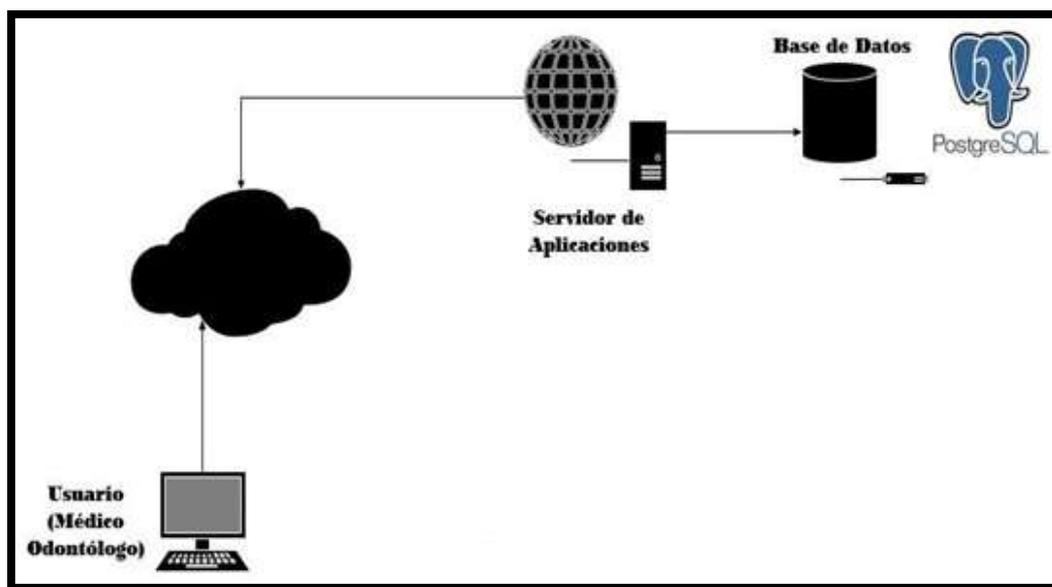


Fig. 1: Esquema de funcionamiento de la aplicación.
Fuente: Propia

El sistema de control odontológico ayudará eficientemente a optimizar los procesos de:

- Control de información y acceso de usuarios del sistema.
- Control de información de los pacientes y médicos.
- Control de procedimientos.
- Control odontológico.

La aplicación de control odontológica se estructurará de acuerdo con los módulos presentados en la Figura 2.

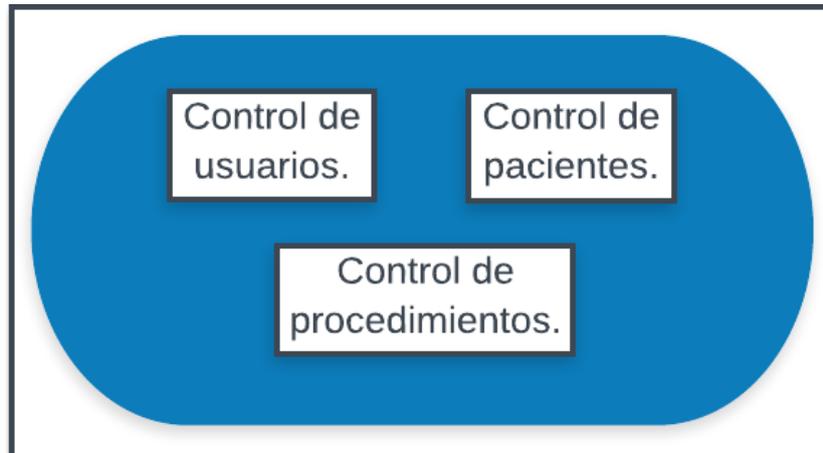


Fig. 2: Módulos del sistema.
Fuente: Propia

- **Módulo control de usuarios**
Permitirá el control total de usuarios que tendrán acceso al sistema odontológico.
- **Módulo control de pacientes**
Permitirá administrar la información de los pacientes y el control odontológico de los mismos.
- **Módulo control de procedimientos**
Permitirá llevar el control de procedimientos odontológicos que brinda la entidad dental.

1.8 Justificación.

Debido al desconocimiento, la desinformación y el escaso uso de metodologías web, se plantea realizar la investigación y comparativa entre estándares metodológicos que sean adaptables en el desarrollo de aplicaciones en base a aspectos que no son considerados al momento de aplicar otro tipo de metodologías sean estas tradicionales, ágiles o híbridas.

Haciendo uso de las metodologías estudiadas se desarrollará una aplicación basada en la web para un grupo médico odontológico que requiere la implementación de herramientas informáticas para el trabajo cotidiano en este sector de salud.

1.9 Impactos

1.9.1 Impacto Tecnológico

Se realiza la comparativa entre metodologías web porque el resultado del estudio aportará en el desarrollo de aplicaciones informáticas con estándares metodológicos de calidad, presentando un método web eficaz que ayude el control de análisis, diseño e implementación de una aplicación informática.

La utilización de la metodología preponderante ayudará en el proceso de elaboración de una aplicación web para el sector odontológico, donde se automatizará los procesos de control y de esta manera se innovará el entorno dental en base a las tecnológicas informáticas.

1.9.2 Impacto Económico

Al realizar la comparativa y encontrar una metodología adecuada se planea reducir el tiempo de desarrollo y costo de la aplicación, esto ayudará eficientemente la construcción de las aplicaciones web.

1.9.3 Impacto Social

Al ejecutar esta comparativa se quiere introducir al entorno de desarrollo de software, la necesidad de utilizar una determinada metodología web para desarrollar las aplicaciones, descartando el uso de cualquier de otras metodologías sean estas, tradicionales, ágiles o híbridas.

De esta manera el uso de un estándar metodológico ayudará en el proceso de desarrollo de la aplicación odontológica, impulsando a todo el grupo de médicos odontólogos a utilizar el aplicativo informático, siendo un software pretendido por los galenos del área odontológica.

Al implementar la aplicación web odontológica se busca modernizar el control de la información en el centro de salud, mediante la utilización de las tecnologías de la información, promoviendo el mejoramiento de la calidad en la prestación de servicios e impactando positivamente a la población, rigiéndose de esta manera a los lineamientos del objetivo 3 presentado en el Régimen Nacional de Gobierno Electrónico y el Plan Nacional del Buen Vivir (Secretaría Nacional de la Administración Pública, 2017).

1.9.4 Impacto Ambiental

Al hacer uso del sistema clínico odontológico se reducirá el uso de papel y material físico pues todos los datos e información generada en las oficinas odontológicas será almacenada en medios informáticos, reduciendo así el consumo de documento físicos y de esta manera aportando al medio ambiente.

CAPÍTULO 2

Marco teórico

2.1 Introducción

Este capítulo presenta el marco teórico orientado principalmente en la conceptualización básica de las metodologías, estructura, notación y el proceso que sigue para el análisis, diseño, desarrollo e implementación de las aplicaciones informáticas web.

De la misma manera, se tomará en cuenta los conceptos de las herramientas que permitirán el desarrollo del aplicativo de control odontológico.

2.2 Definiciones.

2.2.1 Ingeniería Web.

El término Ingeniería Web se define como “la aplicación de enfoques sistemáticos, disciplinados y cuantificables para el desarrollo económico y la evolución de las soluciones informáticas de alta calidad en la Web”.

Aunque la ingeniería web no es una disciplina contraria a la ingeniería de software, ya que propone, por una parte, validar aspectos relevantes como la Usabilidad, Funcionabilidad, Fiabilidad, Seguridad, Eficiencia y Mantenibilidad, y por otra parte mantener las actividades de control de calidad y garantía de la calidad del software, permitiendo de esta manera la construcción de aplicaciones web de excelencia.

2.2.2 Metodologías web.

El avance tecnológico de la World Wide Web (www) ha sido inmensurable, debido a que su alcance ha llegado a ámbitos como educación, economía, industria, negocios, administración, etc. Esto se debe a que en estas áreas se ha optado por implementar aplicativos informáticos que se adapten y mejoren la calidad de servicio del entorno. Sin embargo, existe la duda al desconocer si en la construcción de las aplicaciones se están utilizando estándares metodológicos que controlen el análisis previo, diseño conceptual, estructura navegacional e interfaz de usuario, pues la no utilización de las mismas podría generar aplicaciones improvisadas y que en poco tiempo quedan obsoletas e inutilizables. Por tal motivo es necesario el uso de metodologías que permitan controlar el análisis, diseño, desarrollo e implementación de las aplicaciones web.

Aunque el uso de estándares metodológicos se originó desde la década de los noventa, estos modelos fueron evolucionando e introduciendo nuevos conceptos, técnicas y notaciones que ayuden a mejorar el desarrollo de las aplicaciones web. Los autores (Moreno & Marciszack, 2014) argumentan que:

- Estas metodologías mantienen el ciclo de vida de estilo de desarrollo en cascada, siendo el objetivo de estos métodos principalmente controlar el modelo del contenido y su comportamiento, el modelado de acceso al contenido y cómo será la presentación del contenido al usuario final, ayudando de esta manera a reducir la complejidad de las aplicaciones (pág. 598).

A continuación, la Figura 3. Presenta las principales metodologías web.

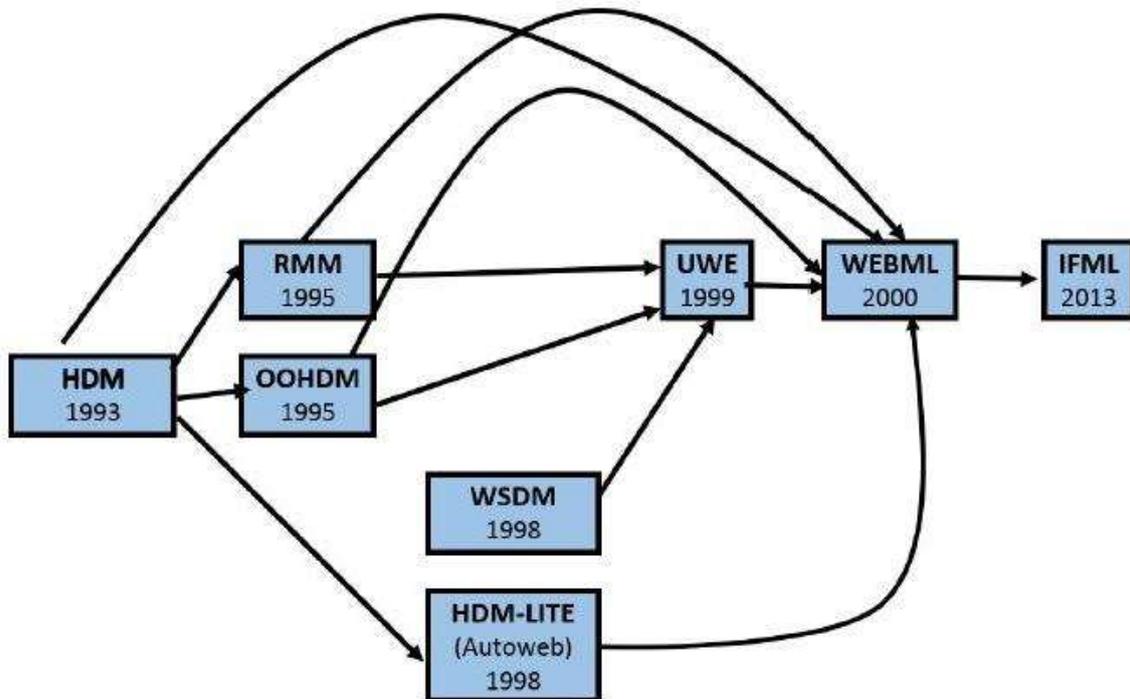


Fig. 3: Principales metodologías de modelado web.
Fuente: (Vera M., 2015)

Estas metodologías han introducido distintas etapas para el desarrollo de aplicaciones web, centrados básicamente en los conceptos presentados por la metodología pionera HDM, de esta manera, la mayoría de los modelos basados en su propia nomenclatura, controlan la construcción del software mediante las siguientes fases:

- Análisis conceptual
- Diseño navegacional
- Presentación
- Implementación.

En base a los principales modelos web a continuación se realiza la conceptualización de los estándares metodológicos y sus etapas para la construcción de aplicaciones web.

2.2.3 HDM – Hipertext design model.

El Modelo de Diseño de Hipertexto (HDM) fue creado por Franca Garzotto, Paolo Paolini y Daniel Schwabe en el año de 1993 como una solución al diseño de aplicaciones hipermedia mediante el modelado conceptual.

El modelo HDM presenta dos etapas de diseño de aplicaciones web:

- El diseño a gran escala (authoring in the large), se refiere a la especificación y diseño de aspectos globales y estructurales de la aplicación
- El diseño a pequeña escala (authoring in the small), que se refiere al desarrollo del contenido los nodos (herramientas DBMS, lenguajes de programación, etc.) (Vera M. , 2015)

Proceso.

Según (Vera M. , 2015) el proceso de desarrollo de HDM se conforma de los siguientes pasos:

1. Identificar el grupo de entidades del mundo real y los componentes. Las relaciones entre una entidad y sus componentes determinan la navegación estructural.
2. Identificar dentro de los objetos del mundo aquellos que tienen estructura y conexiones similares, es decir, los que son del mismo tipo.
3. Determinar el conjunto de objetos y sus miembros. En este paso se diseñan los enlaces de aplicación derivados de las colecciones.
4. Se determina si el tipo de navegación de cada colección es mediante un índice o vista guiada. (pág. 25)

Estructura.

La propuesta de HDM es insertar conceptos como:

- **Nodos.**

“Los nodos permiten agrupar la información. Los nodos son una unidad de información que contiene texto o elementos multimedia (imágenes, audio, video, etc.). Los nodos son representados por una figura ovalada, identificada por un nombre” (Rodríguez, 2010).

(Vilariño, 2010) define a las notaciones de los nodos como:

- **Unidad:** es un componente con una perspectiva predeterminada.
- **Componente:** una entidad dentro de HDM (nodo) es una recopilación de componentes, es decir, que un componente es una abstracción de un conjunto de entidades, las cuales funcionan como contenedores de la información.
- **Entidad:** son cada una de las estructuras de la información dentro del nodo (pág. 26).

- **Enlaces:** según (Rodríguez, 2010) “Los nodos se relacionan mediante enlaces. Un enlace conecta dos nodos unidireccionalmente y se relaciona a un nodo específico”. Los links o enlaces en HDM sirven para navegar y construir una red de navegación, el autor (Vera M. , 2015) clasifica a los enlaces de tres tipos:
- **Perspectiva:** no cambian el foco de atención del tema, sino que muestran otra perspectiva de la información.
- **Estructurales:** permiten conectar distintas secciones de información correspondientes a la misma entidad, sirven por ejemplo para navegar información jerárquica o dividida en varias páginas.
- **Aplicación:** son propios del dominio que se está modelando y permite especificar el tipo de la relación como una etiqueta dentro del mismo link por ejemplo “es autor de” puede vincular un escritor con un libro (pág. 25).

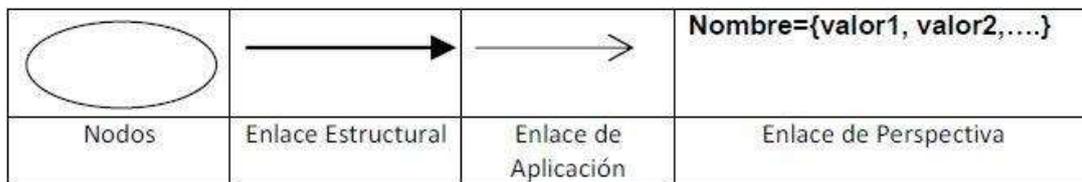


Fig. 4: Notación HDM.
Fuente: (Rodríguez, 2010)

- **Elementos de navegación.**

Según (Rodríguez, 2010) los elementos de navegación se los divide en colecciones y estructuras de acceso.

- **Colección:** es un conjunto de enlaces que apunta a nodos hipermedia. Las colecciones pueden ser de dos tipos: un índice que permite navegar directamente a los nodos que pertenecen a la colección y una vista guiada que muestra la colección de objetos a través de un recorrido secuencial hacia adelante y atrás.
- **Estructuras de acceso:** son componentes que, aunque no tengan un símbolo para especificarse se indican en el nombre del nodo como etiqueta (pág. 19).

2.2.4 RMM –Relationship Management Methodology.

El modelo RMM o Metodología de Administración de Relaciones establecido por Isakowitz, Stohr, & Balasubramanian en 1995, se centra en el proceso de análisis, diseño y construcción de aplicaciones, siguiendo de esta manera los lineamientos del ciclo de vida de desarrollo del software, su modelo de datos se fundamenta en el modelo HDM, pero a diferencia de este,

RMM describe el diseño hipermedia y el uso de una metodología de desarrollo (Vera M. , 2015).

Este modelo proporciona elementos principales como el modelo E-R para el modelado de datos y el modelo RMDM (Relationship Management Data Model) para la definición de entidades y relaciones entre éstas.

Proceso.

(Rodriguez, 2010) divide a las etapas de desarrollo de RMM en:

- **Diseño Entidad – Relación (E-R):** representa el dominio de la aplicación a través de un diagrama E-R mediante entidades, atributos y relaciones.
 - Las entidades y atributos: representan objetos abstractos o físicos y sus propiedades.
 - Las relaciones: describen las asociaciones entre las entidades. Las asociaciones pueden ser uno a uno (1-1) o uno a muchos (1-n).
- **Diseño de secciones (Slice Design):** consiste en dividir una entidad en secciones significativas (slices) y organizarlas dentro de una red de hipertexto.
- **Diseño navegacional:** en esta etapa se diseñan los caminos que existirán en el diagrama del documento hipermedia. Se analizan las relaciones y de acuerdo con ellas se establece el tipo de estructura de acceso:
 - Relaciones 1-1: hiperenlaces.
 - Relación 1-N: índices o vistas.
 - Menú: grupos de entidades de acuerdo con el interés o comodidad del usuario final, sirve para construir los caminos de acceso desde un origen.
- **Diseño de protocolo de conversión:** se establece un conjunto de reglas de conversión de los elementos RMM hacia la plataforma tecnológica de implementación.
- **Diseño de la interfaz de usuario:** consiste en el diseño de la distribución de los elementos en las interfaces del usuario final.
- **Comportamiento en tiempo de ejecución:** representará el historial de navegación y los enlaces de cada página.
- **Construcción y pruebas:** sigue las técnicas establecidas en la ingeniería de software, haciendo énfasis en las pruebas de los caminos navegacionales (pág. 21).

Estructura

(Cardenas, 2011) argumenta que: “la navegación de RMM es modelada a través de enlaces unidireccionales, enlaces bidireccionales, y estructuras de acceso; éstas pueden ser índices, vistas guiadas, índice-vista-guiada y grupos”.

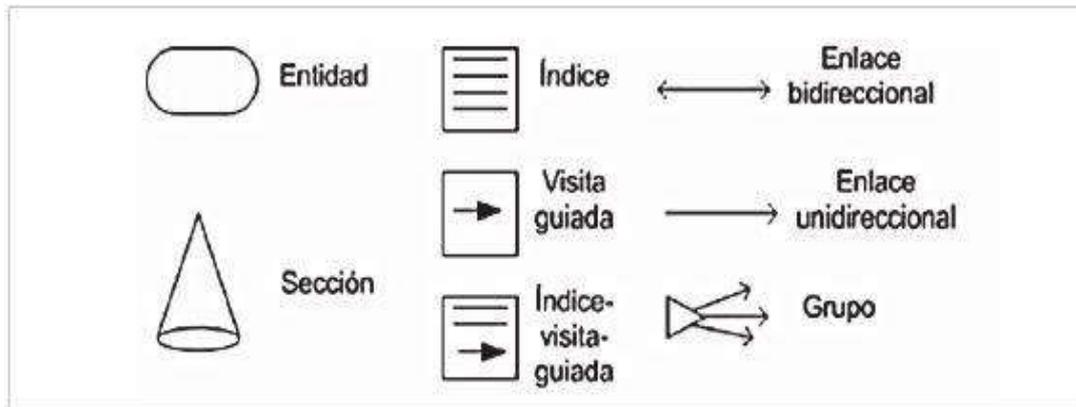


Fig. 5: Notación de RMM.
Fuente: (Cardenas, 2011)

La estructura navegacional de acuerdo (Cardenas, 2011) es representada mediante:

- **Enlaces:** conectan una entidad con sus secciones, o las secciones de una misma entidad entre sí. Las conexiones pueden ser bidireccionales o unidireccionales, y se denominan enlaces estructurales.
- **Grupo:** Es una construcción similar a un menú; está formado por una lista de enlaces (pág. 108).

2.2.5 OOHDM – Object Oriented Hypermedia Design Methodology

OOHDM o Método de Diseño Hipermedia Orientado a Objetos (Schwabe & Rossi, 1995) es una metodología por basada en HDM que se centra en el modelado orientado a objetos (Vera M. , 2015, pág. 26)

OOHDM realiza el modelado de la navegación a través del diagrama de clases navegacionales y diagrama de contextos. Este modelo establece cuatro actividades para el diseño hipermedia, según (Vera, Pons, González, Guilianelli, & Rodríguez, 2013) son:

- Modelado conceptual.
- Modelado de la navegación.
- Diseño de la interfaz abstracta.
- Implementación.

Proceso.

En la Tabla 1 se presenta la metodología OOADM con sus cinco etapas para la construcción de aplicaciones web:

Tabla 1: Etapas y sub etapas de OOADM.

ETAPAS	SUBETAPAS
Obtención de Requerimientos	Identificación de Roles y Tareas. Especificación de Escenarios. Especificación de Casos de Uso. Especificación de UIs.
Modelo Conceptual	Diagrama de Clases Diagrama de secuencias Diagrama de la Estructura de la Base de datos.
Modelo Navegacional	Aplicación de Diseño Navegacional. Esquema de Clases Navegacionales.
Diseño de Interfaz Abstracta	ADVs del Sistema
Implementación	Desarrollo del Sistema. Pruebas del Sistema.

Fuente: (Garcés & Hinojosa, 2011)

Obtención de Requerimientos: Mediante la utilización de los diagramas de casos de uso en esta fase se obtiene de manera clara y concisa los requerimientos y acciones del sistema.

Diseño conceptual: se construye el modelo orientado a objetos representado por clases, relaciones y colaboraciones entre ellos, así como el diagrama Entidad – Relación y script de base de datos (García, Gutiérrez, Camacho, Lorenzo, & Escalamilla, 2015).

El resultado de esta fase es un modelo de clases relacionadas que se dividen en subsistemas. Los autores (Collaguazo & Pulloquina, 2014) mencionan que en esta fase se debe cumplir con:

- **Productos:** diagramas de clases, división en subsistemas y relaciones.
- **Herramientas:** técnicas de modelado orientado a objeto y patrones de diseño.
- **Mecanismos:** clasificación, agregación, generalización y especialización.
- **Objetivos de diseño:** modelo semántico de la aplicación (pág. 4).

Diseño navegacional: el autor (Vera M. , 2015) menciona que esta etapa define la estructura de navegación de la aplicación, estableciendo uno o más contextos navegacionales. Sobre

un mismo modelo conceptual pueden definirse varios modelos navegacionales para definir distintas vistas sobre el mismo dominio (pág. 26).

Al igual que HDM y otros modelos, los autores (Collaguazo & Pulloquina, 2014) han presentado en su investigación cierta notación para esta etapa:

- **Nodos:** son contenedores básicos de información de las aplicaciones hipermedia.
- **Enlaces:** pueden actuar como un objeto intermedio o como un puente de conexión entre nodos.
- **Estructuras de acceso:** actúan como índices o diccionarios para encontrar de forma rápida la información deseada.
- **Los menús, índices o guías de ruta:** son ejemplos de estas estructuras. Las estructuras de acceso también se modelan como clases, compuestas por un conjunto de referencias a objetos que son accesibles desde ella y una serie de criterios de clasificación de las mismas (pág. 5).

Diseño de interfaz abstracta: se define la construcción de interfaces de usuario, es decir los elementos gráficos (cuadros de texto, botones, etc.) que se presentarán en cada página, los eventos y la comunicación entre distintos objetos navegacionales.

Implementación: en esta etapa se utilizará el diseño desarrollado y se realizará la construcción de la aplicación. Los autores (Collaguazo & Pulloquina, 2014) destacan que esta fase debe cumplir con los siguientes lineamientos:

- **Productos:** Aplicación ejecutable.
- **Herramientas:** el entorno del lenguaje de programación.
- **Mecanismos:** los ofrecidos por el lenguaje de desarrollo.
- **Objetivo de diseño:** obtener la aplicación ejecutable (pág. 6).

2.2.6 WSDM - Web site design method.

WSDM o método de diseño de sitios web (De Troyer O.M.F., 1998), es una metodología basada en el usuario, donde define un conjunto de usuarios potenciales del sitio web. Realiza una clasificación de dichos usuarios para adaptarse a distintos perfiles. Esta metodología está orientada a sitios web para consulta de información y no para aplicaciones web donde el foco está en administrar información (Vera M. , 2015, pág. 26).

Proceso.

Según (Rojo, 2013), el modelo WSDM consta de las siguientes fases:

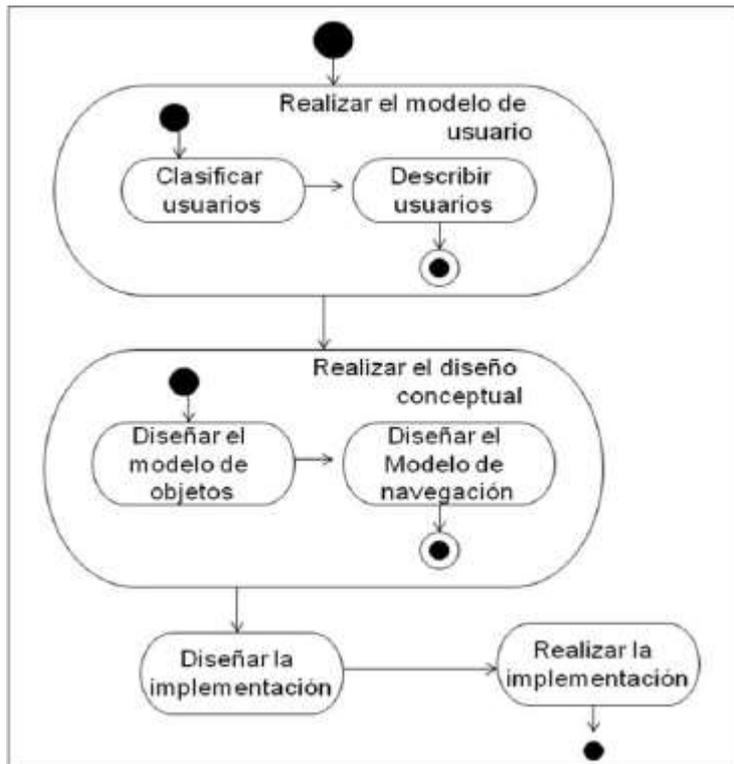


Fig. 6: Diagrama de fases de WSDM
Fuente: (Rojo, 2013)

- **Modelo de Usuario:** se centra en la identificación, clasificación, descripción de características y levantamiento de requerimientos de información, consta de dos subfases:
 - Clasificación de usuarios: identifica las actividades y personas involucradas en la aplicación.
 - Descripción de usuarios: describe las relaciones y tareas de los usuarios descritos.
- **Diseño Conceptual:** se realiza el diseño conceptual de los requerimientos levantados en la fase anterior y el diseño navegacional de la aplicación. Para esta fase se planteó los siguientes procesos:
 - Diseño modelo de objetos: desarrolla el diseño de objetos mediante un modelo conceptual. El proceso describe los objetos, relaciones, restricciones y comportamiento de los mismos.
 - Diseño modelo de navegación: establece la navegación del usuario en las páginas o sitios de la aplicación, mediante componentes, links o enlaces que conecten entre sí (pág. 39).

- **Diseño de Implementación:** se realiza el diseño de la interfaz que tendrá la aplicación web para cada usuario de la aplicación, esto ayuda a los usuarios a tener un modelo mental de cómo estará diseñada la aplicación.
El diseño de la implementación mucho dependerá de las tecnologías seleccionadas para el desarrollo del software.
- **Implementación:** en esta fase se realiza la construcción de la aplicación web en base al modelo obtenido anteriormente (Rojo, 2013).

2.2.7 HDM-Lite.

El autor (Vera M. , 2015) menciona que HDM-Lite (Piero Fraternali, 1998) es una evolución de HDM diseñada específicamente para sitios web. Permite modelar un sitio web definiendo su modelo conceptual formado por diferentes esquemas que permiten:

- Describir la estructura de información que maneja la aplicación.
- Definir la forma en que se accede a la información definida en el modelo anterior. Es decir, define la navegación del sistema.
- Definir la apariencia de las páginas de la aplicación, definiendo la distribución de la página y la ubicación de los controles.

Para lograr independencia de la interfaz, dinamismo y separación de conceptos, la metodología se plantea una transformación de modelo conceptual a un modelo lógico donde cada uno de los esquemas se mapea a repositorios en una base de datos relacional (pág. 27).

2.2.8 UWE – UML-Based Web Engineering.

UWE nace como modelo de Ingeniería Web basado en UML, establecido por Koch y Kraus en el 2002, siendo una propuesta metodológica detallada para el desarrollo de aplicaciones web con una definición exhaustiva del proceso de diseño que debe de ser utilizado. Este proceso espiral, interactivo e incremental, incluye flujos de trabajo (disciplinas) y puntos de control, sus fases coinciden con las propuestas en el Proceso de Desarrollo Unificado (RUP): Comienzo o inicio, Elaboración, Construcción y Transición (Silva, 2012, pág. 29).

UWE hace el uso de notación UML y diagramas basados totalmente en el Lenguaje de Modelado Unificado, siempre que sea posible utilizarlos en el análisis y diseño de las aplicaciones web, por lo que las características específicas de la web como y enlaces de la estructura de hipertexto UWE define estructuras, valores y restricciones definidas por los

elementos de modelado. La extensión UWE controla la navegación, presentación, procesos de negocio y aspectos de adaptación de la aplicación. (UWE, 2017)

(UWE, 2017) establece que el enfoque principal de UWE se centra en:

- Lenguaje de modelado específico de dominio basado en UML que incluye características de seguridad.
- Metodología basada en modelos.
- Soporte de herramienta para el diseño sistemático.
- Soporte total para la generación automática de aplicaciones web.

Proceso.

El autor (Vilariño, 2010) según el modelo UWE propone las siguientes etapas para el desarrollo de aplicaciones web:

- 1. Especificación de Requerimientos:** es donde se describen los requisitos funcionales de la aplicación a desarrollar. UWE propone el modelo de casos de uso de UML para el levantamiento de los requerimientos, ya que a través de esta herramienta se puede describir una parte del comportamiento de la aplicación sin revelar la estructura interna, así como la identificación de los distintos usuarios que interactuarán con la aplicación.
- 2. Modelo de contenido:** especifica los elementos del dominio de la aplicación. UWE propone la utilización del diagrama de clases de UML (pág. 43).
- 3. Modelo de Navegación:** establece las estructuras de navegación mediante elementos de acceso ya conocidos (nodos y enlaces), que permiten comunicarse entre los objetos del espacio navegacional. El diseño navegacional está constituido por menús, índices, vistas guiadas y formularios.
- 4. Modelo de presentación:** UWE relaciona la presentación con los elementos de la interfaz de usuario, se los define también como estereotipos de UML. Los elementos del diseño de presentación son: ventanas, entradas de texto, imágenes, audio, botones, etc. La estructura de presentación es modelada con clases y subclases especializadas que contengan la lógica interfaz de usuario y la navegación entre las clases respectivas.
- 5. Modelo de proceso:** presenta detalla a profundidad las acciones y procesos que se involucran en las actividades y funcionalidades de la aplicación web, este modelo describe dos sub fases:
 - **Modelo de estructura de proceso:** describe las relaciones entre las diferentes clases de proceso.

- **Modelo de flujo de proceso:** especifica las actividades conectadas en cada proceso (Coro, 2014).

2.2.9 WEBML – Web Modelling Language.

WEBML o Lenguaje de Modelado Web, se define como un lenguaje de modelado de alto nivel para la especificación de aplicaciones web. Se basa en sus modelos anteriores como HDM, HDM-Lite, RMM, OOHDM y Araneus (Atzeni Pablo, 1998) que es un repositorio de datos con administración web (Vera M. , 2015).

Esencialmente WebML consiste en gráficos simples y conceptuales para expresar hipertexto como un conjunto de páginas y operaciones, WebML representa una página como una estructura compuesta por unidades de contenidos y links o enlaces. Una aplicación web basada en WebML conste de tres grandes partes conceptuales, la estructura, el hipertexto y la presentación (Silvera, y otros, 2015).

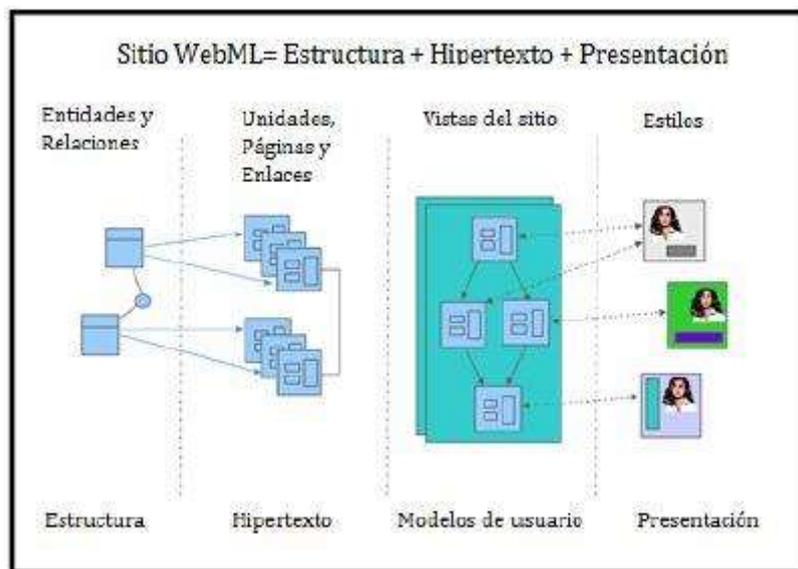


Fig. 7: Componentes de un sitio Web con WebML.
Fuente: (Silvera, y otros, 2015)

Proceso.

Según (Vera M. , 2015) el modelo WebML distingue cuatro etapas para el desarrollo de aplicaciones web:

1. **Modelo Estructural:** expresa en contenido de los datos de la aplicación, en términos de entidades y relaciones pertinentes. WebML no propone un nuevo lenguaje de modelado, sino que es compatible con notaciones como el modelo Entidad – Relación y los diagramas UML.

2. **Modelo de Hipertexto:** describe uno o más hipertextos que pueden ser publicados en el sitio. Cada hipertexto define una vista del sitio llamada View Site, que a su vez consta de dos submodelos:
 - **Modelo de Composición:** especifica que paginas componen el hipertexto y que unidades de contenido constituyen una página. Existen seis tipos de unidades de contenidos que se pueden utilizar para componer las páginas: datos, multi-datos, índices, filtros, unidades desplazador y directas. Las unidades de datos se utilizan para publicar la información de un solo objeto, mientras que los restantes tipos de unidades representan formas alternativas de navegar por un conjunto de objetos.
 - **Modelo de Navegación:** es donde se expresa la manera en la cual las páginas se vinculan con el contenido de las unidades para dar forma al hipertexto. (pág. 28)
3. **Modelo de Presentación:** en esta etapa se define la diagramación e interfaz de usuario de las páginas. Las especificaciones de la presentación pueden ser específicas o generalizadas.
4. **Modelo de Personalización:** Los tipos de usuarios que interactuarán con la aplicación son modelados explícitamente en un esquema estructural, donde cada entidad del esquema pertenece a cada tipo de usuario identificado. Las características de estas entidades se pueden utilizar para almacenar un grupo específico de contenido (Vilariño, 2010).

2.2.10 IFML – Interaction flow modeling language (IFML).

El modelo IFML nace en el 2013 como un estándar aprobado por el OMG, está basado en la metodología WebML y diseñado para expresar el comportamiento de contenidos, interacción con el usuario y el control del Front-End de las aplicaciones web, la versión 1.0 de este método fue presentada el 02 de febrero del 2015 (IFML, 2017).

El objetivo de IFML es proporcionar a arquitectos, ingenieros y desarrolladores de software una herramienta para la definición del modelado del flujo de interacción de aplicaciones, que ayude al usuario final a percibir una descripción sencilla de la estructura y comportamiento de la aplicación (IFML, 2017).

Según (Vera M. , 2015) al estar basado en MDD (Model Driven Development) es posible generar de forma automática el código fuente para el Front-End de las aplicaciones (pág. 30).

Proceso.

De acuerdo con los ejemplos presentados por (IFML, 2017) el modelo IFML distingue las siguientes fases para el diseño de aplicaciones web:

1. **Modelo de Contenidos:** especifica el contenido de la aplicación mediante el diseño entidad – relación, clases y sus objetos relacionados.
2. **Modelo de Procesos:** desarrolla el diagrama de procesos que seguirá la aplicación.
3. **Modelo de Interfaces:** diseña la interfaz gráfica del software desarrollado.
4. **Modelado del Sistema:** genera la aplicación web.

2.3 Conceptos tecnológicos.

2.3.1 Aplicación web.

Una aplicación web se considera una plataforma interactiva donde el usuario tiene como objetivo principal realizar acciones específicas como transacciones bancarias, mercadeo electrónico, etc. Su construcción conlleva un desarrollo complejo y para la ejecución es necesario utilizar un navegador web.

Tipos

Según la investigación de los autores (Oliveros, Danyans, & Mastropietro, 2014) las aplicaciones web se pueden clasificar en:

- **Centradas en documentos:** páginas web con HTML estático, residen en un servidor web y por lo general se actualizan manualmente.
- **Móviles:** su acceso es ubicuo ya que provee un servicio personalizado en cualquier momento, lugar y dispositivo.
- **Portales:** proveen un punto de acceso a fuentes separadas y heterogéneas de información de servicios.
- **E-Business (Basadas en flujo de trabajo):** permiten el manejo de flujos de trabajos entre diferentes compañías o usuarios.
- **Sociales:** Las personas proveen su identidad a una comunidad con intereses similares.
- **Colaborativas:** Empleadas en especial para colaboración en operaciones no estructuradas. Soportan información compartida y espacios de trabajo.
- **Transaccionales:** Proporcionan mayor interactividad, el usuario interactúa con la aplicación (pág. 8).

2.3.2 Servidor de aplicaciones.

Según el sitio web (Computer Hope, 2017) “Un servidor de aplicaciones es definido como un software de servidor, es decir, un entorno donde se puede ejecutar cualquier tipo de aplicaciones web sin importar sus funciones. Existe un diverso grupo de este tipo de servidores que permiten alojar aplicaciones desarrolladas en tecnologías Java, PHP, .Net, etc.

El uso de los servidores de aplicaciones denota una serie de ventajas tales como:

- Proporcionar integridad al permitir un enfoque más centralizado en las actualizaciones de las aplicaciones.
- Seguridad al centralizar la administración de autenticación y acceso a datos.
- El rendimiento puede aplicaciones de uso intensivo puede optimizarse, limitando el tráfico de red, mejorando así los tiempos de respuesta de las aplicaciones.”

2.4 Herramientas.

2.4.1 PostgreSQL.

PostgreSQL es un poderoso sistema de base de datos relacional de objetos de código abierto con más de 15 años de desarrollo activo y arquitectura comprobada que le ha valido para demostrar una sólida reputación de fiabilidad, integridad de datos y corrección. Se ejecuta en los principales sistemas operativos, además tiene soporte completo para índices, vistas, disparadores y procedimientos almacenados. Incluye la mayoría tipos de datos SQL como INTEGER, NUMERIC, BOOLEAN, CHAR, VARCHAR, DATE, INTERVAL y TIMESTAMP. Es compatible con el almacenamiento de objetos binarios, incluyendo imágenes, sonidos y video. Tiene interfaces de programación nativas para C/C++, Java, .Net, Perl, Python, Ruby, Tcl, ODBC entre otros.

Es considerada una base de datos de clase empresarial porque cuenta con funciones sofisticadas como Control de concurrencia multi versión (MVCC), recuperación puntual, tablespaces, replicación asíncrona, transacciones anidadas (puntos de guardado), copias de seguridad en línea (caliente), un sofisticado planificador/ optimizador de consultas y registro anticipado para tolerancia a fallas. Admite conjuntos de caracteres internacionales, codificación de caracteres multibyte, unicode, compatible con la configuración regional y distinción en el formateo de mayúsculas y minúsculas. Es altamente escalable tanto en la gran cantidad de datos que puede administrar como en la simultaneidad de usuarios que permite conectarse al gestor de base de datos (PostgreSQL, 2017).

2.4.2 Java EE (Java Enterprise Edition).

Es una plataforma de programación para desarrollo de aplicaciones empresariales basado en el lenguaje Java, permite utilizar una arquitectura N capas y la ejecución de sus componentes puede realizarse en un servidor de aplicaciones. Java EE ofrece una plataforma muy robusta e integra características que mejoran la portabilidad de las aplicaciones y mejorar la productividad del desarrollador, siendo un lenguaje de código abierto muy utilizado en la industria del desarrollo de software (Oracle, 2017).

2.4.3 JavaServer Faces (JSF).

Los autores (Roldán, Valderas, & Pastor, 2010) definen a JSF o Java Server Faces como un marco de trabajo para el desarrollo de aplicaciones web, donde permite construir la capa de presentación de manera ágil y sencilla, utilizando una serie de interfaces de usuario estándar permitiendo el desarrollo de componentes propios.

JSF define tres elementos importantes:

- Conjunto de elementos de interfaz de usuario (widgets), tales como botones, listas de selección, casillas de selección, hipervínculos, botones de acción, etc.
- Una arquitectura de componentes que establece mecanismos estándar para la construcción de widgets.
- La infraestructura de la aplicación a través de los ficheros de configuración de componentes, descripción de reglas de navegación, etc (pág. 211).

2.4.4 PrimeFaces.

El framework PrimeFaces es un marco de código abierto popular para JavaServer Faces con más de 100 componentes, touchkit móvil optimizado, validación del lado del cliente, motor de temas y más herramientas para soporte de los desarrolladores de aplicaciones informáticas (Primefaces, 2017).

2.4.5 OpenShift.

OpenShift es una Plataforma como Servicio (PaaS), que permite la línea de desarrollo de aplicaciones escalables host y el alojamiento en la nube de Red Hat.



Fig. 8: OpenShift y sus servicios en la nube.
Fuente: (Red Hat, 2017)

Debido a su amplio soporte de herramientas y tecnologías brindadas a profesionales y empresas la comunidad tecnológica ha establecido a OpenShift como la elección más estable al momento de utilizar una nube pública (Red Hat, 2017).

CAPÍTULO 3

Análisis comparativo de metodologías

3.1 Introducción.

En la actualidad existen diversas metodologías web, algunas muy similares entre sí, otras totalmente distintas, como también se pueden encontrar algunas que apuntan a resolver un solo tipo de problemas, de la misma forma hay otras que buscan abarcar mucho más. Para realizar este estudio de comparación de metodologías presentamos criterios apropiados a las metodologías que permitan distinguir las características únicas y distintas del resto de las metodologías. Además, se presentan criterios específicos que permiten hacer la diferenciación entre las metodologías que persiguen objetivos comunes (Villarroel & Rioseco, 2011).

En base a varios criterios es necesario realizar la aplicación de una encuesta a un determinado grupo de personas de la UTN, los cuales nos permitirán obtener datos estadísticos que ayudará en la selección de la metodología más adecuada para el desarrollo de aplicaciones web.

3.2 Aplicación de la encuesta.

3.2.1 Encuesta.

Es una herramienta para obtener información sobre una población determinada, mediante la elaboración de un cuestionario que se aplica a los individuos que conforman dicha población. La información requerida puede referirse a los atributos de los sujetos que componen esa población, así como también su comportamiento actitudes, opiniones, necesidades, características entre otras.

Para contribuir con la investigación de las metodologías web se realizó una encuesta, la misma que contiene preguntas que serán respondidas por los estudiantes de los últimos niveles de la carrera de Ingeniería en Sistemas Computacionales de la Universidad Técnica del Norte y personas que trabajen en empresas de desarrollo de software.

3.2.2 Población.

Corresponde al conjunto de referencia sobre el cual se va a desarrollar la investigación, en este caso se tomó como población a los estudiantes de niveles superiores, docentes de CISIC de la UTN y personal capacitado en el desarrollo de aplicaciones informáticas.

3.2.3 Encuesta aplicada a la población.

La encuesta se realizó a 35 personas entre estudiantes, profesionales que trabajen en empresas de desarrollo de software, de igual manera a docentes de la carrera de Ingeniería en Sistemas Computacionales de la UTN.

Cuestionario

Pregunta 1

¿Qué tipos de metodologías de desarrollo de software ha utilizado?

Tabla 2: Resultados de la pregunta 1

RESPUESTAS	PERSONAS	% FRECUENCIA
TRADICIONALES	13	37,1%
WEB	26	74,3%
HÍBRIDAS	6	17,1%
ÁGILES	24	68,6%
MÓVILES	3	8,6%
NINGUNA	1	2,9%

Fuente: Propia

Representación gráfica

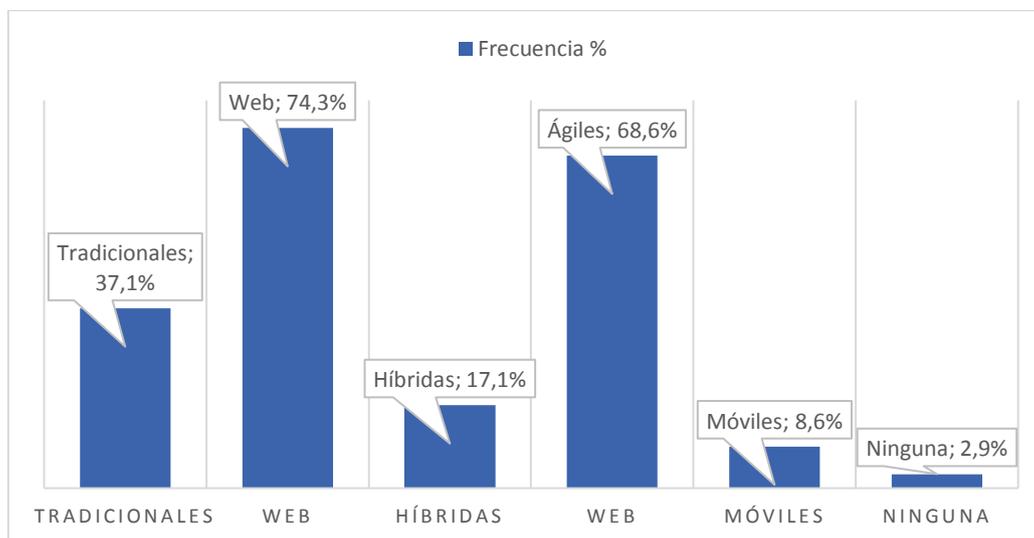


Fig. 9: Resultados de la pregunta 1

Fuente: Propia

Resultados:

De la población investigada, un 74,3% de los encuestados han utilizado metodologías web, representando el porcentaje más alto en comparación con las metodologías ágiles con 68,6%, a su vez las metodologías tradicionales un 37,1%, metodologías híbridas con 17,1%,

metodologías móviles un 8,6% y finalmente un 2,9% el personal que no ha utilizado ninguna metodología.

Los resultados resumen pueden reflejarse en la siguiente tabla:

Tabla 3: Resultados resumen de la pregunta 1

RESPUESTAS	VALOR PORCENTUAL
WEB	74,3%
ÁGILES	68,6%
TRADICIONALES	37,1%
HÍBRIDAS	17,1%
MÓVILES	8,6%
NINGUNA	2,9%

Fuente: Propia

Pregunta 2

¿Qué metodologías web ha utilizado?

Tabla 4: Resultados de la pregunta 2

	PERSONAS	FRECUENCIA %
OOHDM	13	37,1%
WSDM	12	34,3%
UWE	13	37,1%
WEBML	11	31,4%
W2000	3	8,6%
SOHDM	1	2,9%
IFML	1	2,9%
NINGUNA	1	2,9%

Fuente: Propia

Representación gráfica

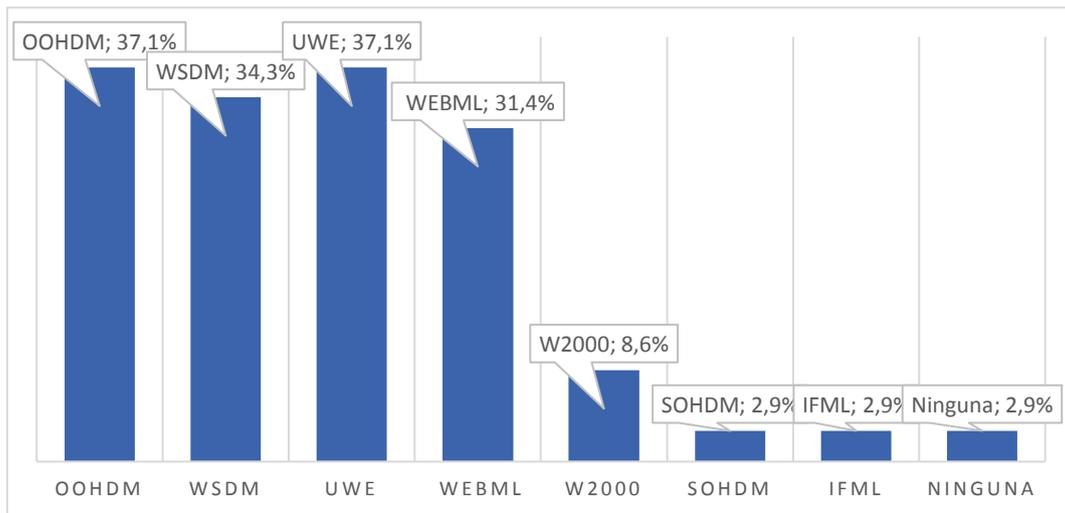


Fig. 10: Resultados de la pregunta 2
Fuente: Propia

Resultados

Los resultados de la encuesta han demostrado que un 37,1% han utilizado las metodologías OOHDM y UWE respectivamente, mientras que un 34,3% del grupo hizo uso del modelo WSDM, un 31,4% ha señalado que tiene conocimiento de WEBML, seguido de W2000 con un 8,6% y con un porcentaje mínimo del 2,9% las metodologías menos utilizadas han sido SOHDM, IFML y NINGUNA.

A continuación, se presenta una tabla resumen de las metodologías web que entrarán en el proceso del análisis comparativo.

Tabla 5: Resultados resumen de la pregunta 2

VALOR PORCENTUAL	
OOHDM	37,1%
UWE	37,1%
WSDM	34,3%
WEBML	31,4%
W2000	8,6%
SOHDM	2,9%
IFML	2,9%
NINGUNA	2,9%

Fuente: Propia

Pregunta 3

¿Considera que las metodologías web permitirán construir aplicaciones informáticas eficientes?

Tabla 6: Resultados de la pregunta 3

RESPUESTAS	PERSONAS	% FRECUENCIA
SI	32	91,4%
NO	2	5,7%
TAL VEZ	1	2,9%
TOTAL	35	100%

Fuente: Propia

Representación gráfica

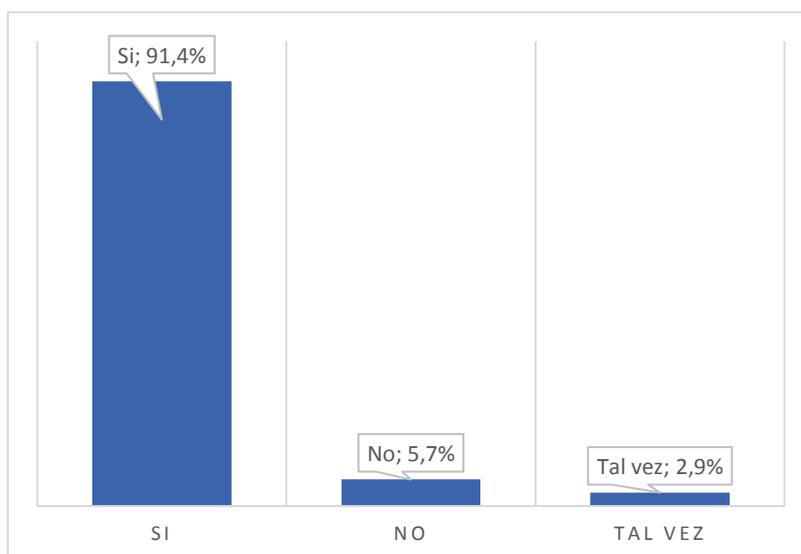


Fig. 11: Resultados de la pregunta 3

Fuente: Propia

Resultados

De la población encuestada un 90% considera que las metodologías web permitirán construir aplicaciones más robustas y eficientes en cambio un 6,7% del grupo argumenta que no es necesaria la utilización de una metodología y un 3% da como respuesta que Tal vez se requiera optar por el uso de un método estándar de desarrollo.

Tabla 7: Resultados resumen de la pregunta 3

RESPUESTAS	VALOR PORCENTUAL
SI	91,4%
NO	5,7%
TAL VEZ	2,9%
TOTAL	100%

Fuente: Propia

Pregunta 4

¿Considera que la usabilidad es un factor determinante al utilizar metodologías web?

Tabla 8: Resultados de la pregunta 4

RESPUESTAS	PERSONAS	% FRECUENCIA
SI	32	92,4%
NO	3	8,6%
TOTAL	35	100%

Fuente: Propia

Representación gráfica

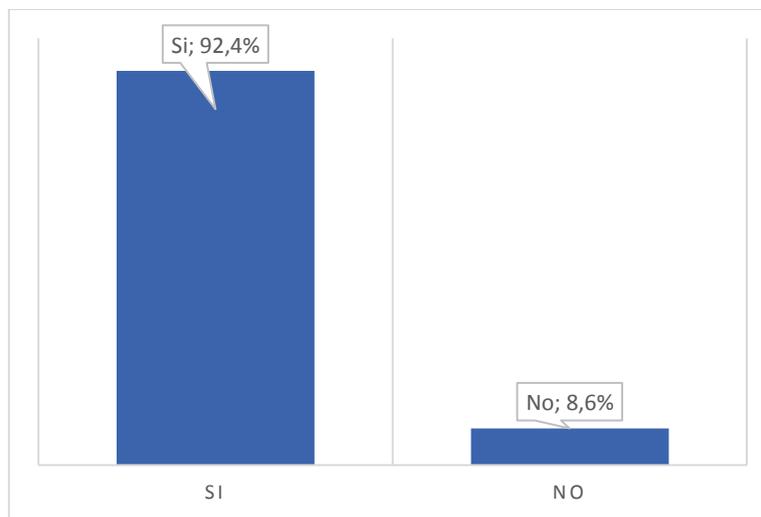


Fig. 12: Resultados de la pregunta 4

Fuente: Propia

Resultados

De la población encuestada un 92,4% considera que la usabilidad es un factor importante que analizan las metodologías web, mientras que un menor porcentaje del 8,6% argumenta que estos modelos no dan importancia a la usabilidad.

Estos resultados pueden visualizarse en la siguiente tabla resumen:

Tabla 9: Resultados resumen de la pregunta 4

RESPUESTAS	% FRECUENCIA
SI	92,4%
NO	8,6%
TOTAL	100%

Fuente: Propia

Pregunta 5

Seleccione las ventajas que brinda la utilización de metodologías web

Tabla 10: Resultados de la pregunta 5

	PERSONAS	FRECUENCIA %
REDUCCIÓN DE COSTOS	16	45,7%
REUTILIZACIÓN DE CÓDIGO	19	54,3%
DOCUMENTACIÓN DISPONIBLE	5	14,3%
TIEMPO DE DESARROLLO MÍNIMO	13	37,1%
BUEN FUNCIONAMIENTO DE LA APLICACIÓN	13	37,1%

Fuente: Propia

Representación gráfica

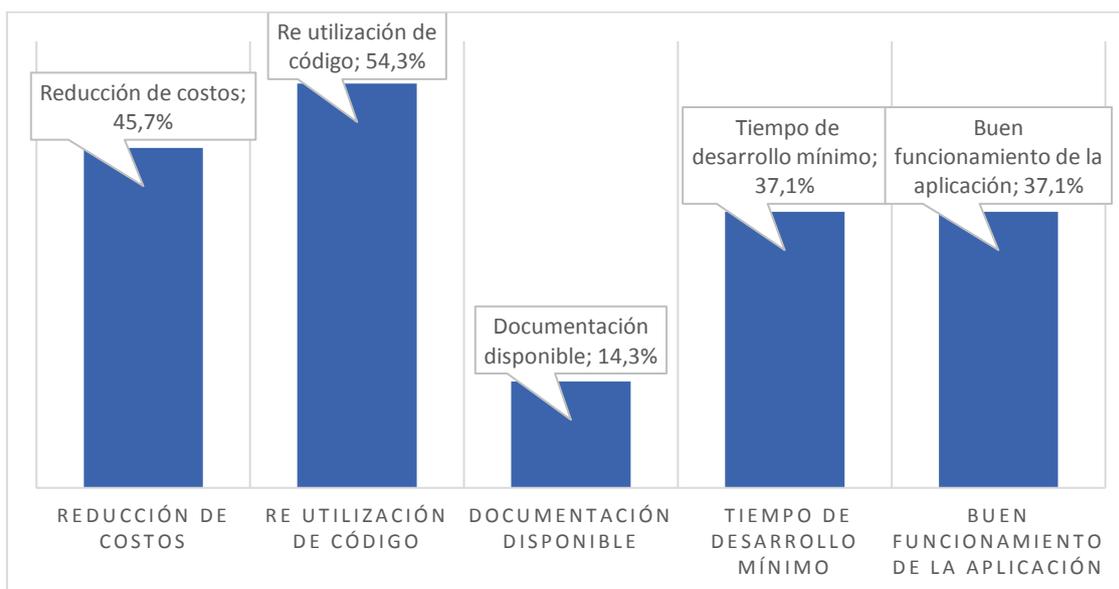


Fig. 13: Resultados de la pregunta 5

Fuente: Propia

Resultados

De la población investigada, el 54,3% del grupo selecciona que la mayor ventaja al utilizar metodología web es la reutilización de código, seguido por la reducción de costos con un 45,7%, el tiempo de desarrollo mínimo y el buen funcionamiento de la aplicación con un 37,1% y como un factor menos determinante la disponibilidad de documentación con un 14,3%.

Estos resultados pueden reflejarse en la siguiente tabla resumen:

Tabla 11: Resultados resumen de la pregunta 5

	VALOR PORCENTUAL
REUTILIZACIÓN DE CÓDIGO	54,3%
REDUCCIÓN DE COSTOS	45,7%
TIEMPO DE DESARROLLO MÍNIMO	37,1%
BUEN FUNCIONAMIENTO DE LA APLICACIÓN	37,1%
DOCUMENTACIÓN DISPONIBLE	14,3%

Fuente: Propia

Pregunta 6

Seleccione alguna desventaja al utilizar metodologías web

Tabla 12: Resultados de la pregunta 6

	PERSONAS	FRECUENCIA %
MAYOR TIEMPO DE DESARROLLO	14	40%
NO REUTILIZA CÓDIGO	4	11,4%
FALTA FIABILIDAD	11	31,4%
NO ÚTIL EN EL DESARROLLO DE APLICACIONES A LARGO PLAZO	6	17,1%
DESCONOCIMIENTO TOTAL DEL MÉTODO	1	2,9%
COMPLEJIDAD EN EL DESARROLLO	1	2,9%
POCA DOCUMENTACIÓN	1	2,9%

Fuente: Propia

Representación gráfica

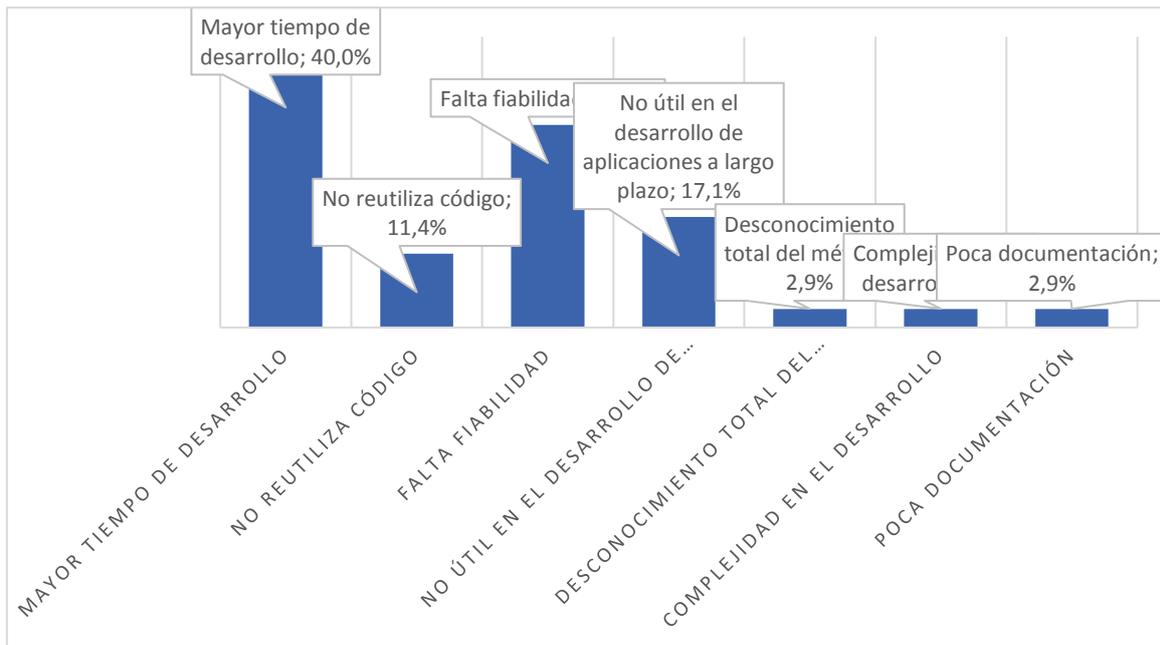


Fig. 14: Resultados de la pregunta 6
Fuente: Propia

Resultados

La población encuestada mediante los resultados entregados menciona que la desventaja más desfavorable de utilizar metodologías web es el mayor tiempo de desarrollo con un valor porcentual del 40%, a esto se añade la desventaja a ser tomada en cuenta es la falta de fiabilidad con 31,4%, no útil en el desarrollo de aplicaciones a largo plazo con 17,1%, no reutilización de código con 11,4% y en porcentajes menores se presentan el desconocimiento total del método, la complejidad en el desarrollo y la poca documentación con un porcentaje del 2,9% frente a los demás resultados.

Estos argumentos se pueden visualizar en la siguiente tabla resumen.

Tabla 13: Resultados resumen de la pregunta 6

	VALOR PORCENTUAL %
MAYOR TIEMPO DE DESARROLLO	40%
FALTA FIABILIDAD	31,4%
NO ÚTIL EN EL DESARROLLO DE APLICACIONES A LARGO PLAZO	17,1%
NO REUTILIZA CÓDIGO	11,4%
DESCONOCIMIENTO TOTAL DEL MÉTODO	2,9%
COMPLEJIDAD EN EL DESARROLLO	2,9%
POCA DOCUMENTACIÓN	2,9%

Fuente: Propia

3.2.4 Resultado de la encuesta aplicada a la población.

En base a los resultados de la encuesta planteada, se puede argumentar que el grupo investigado tiene un amplio conocimiento del uso de metodologías web, por lo que es necesario un análisis de los modelos web más seleccionados por la comunidad estudiada, en este caso las metodologías con mayor porcentaje de uso son OOHDM, WSDM, UWE.

Para realizar el análisis de las metodologías es necesario el desarrollo de prototipos en donde se utilicen los modelos elegidos anteriormente, esto permitirá seleccionar la mejor metodología web, en base a parámetros comparativos que se fundamenten en un modelo o norma estándar.

3.2.5 Desarrollo de prototipos.

En esta etapa se desarrollará un prototipo del Sistema de Control Odontológico mediante el uso de las metodologías OOHDM, WSDM, UWE.

El objetivo del desarrollo es reconocer las fases, diagramas y procesos que conlleva cada metodología, obteniendo de esta manera las ventajas y desventajas que brinda el uso de cada estándar metodológico.

3.2.6 Prototipo de la metodología OOHDM.

La metodología OOHDM (Object Oriented Hypermedia Design Methodology) es un modelo muy utilizado en el diseño de aplicaciones web, su proceso consta de las siguientes fases:

1. Análisis de Requerimientos

En este proceso se analiza los requisitos de la aplicación informática, su objetivo es describir la funcionalidad general del software.

Dentro de esta fase la metodología OOHDM ha establecido cinco sub etapas:

a) Identificación de Roles y Tareas

En esta etapa se identifica los roles y tareas que tendrán los usuarios en la aplicación.

El rol que se determinó es el usuario **Administrador**.

Administrador

- Acceso al sistema
- Control de usuarios.
- Control de médicos.
- Control de pacientes.
- Control odontológico.

- Control de procedimientos.
- Control de reportes

b) Especificación de escenarios

Los escenarios describen a los usuarios cómo administrarán la aplicación, en este caso se tiene un usuario **Administrador** y trabajará en las siguientes escenas.

Acceso al sistema

Tabla 14: Especificación de escenario de acceso al sistema

ENTRADA	PROCESO	SALIDA
INGRESO DE USUARIO Y CONTRASEÑA.	Verificar que la información ingresada esta previamente registrada en la base de datos.	Acceder al sistema.

Fuente: propia

Administración del sistema

Tabla 15: Administración del sistema

ENTRADA	PROCESO	SALIDA
INFORMACIÓN VALIDADA DEL USUARIO.	Presentar pantallas de control del sistema (usuarios, médicos, pacientes, pagos o procedimientos y reportes)	Visualizar las pantallas de control del sistema.

Fuente: propia

Reportes

Tabla 16: Reportes del sistema

ENTRADA	PROCESO	SALIDA
LISTA DE REPORTE NECESARIOS.	Presentar la información de usuarios, médicos, pacientes, pagos o procedimientos.	Mostrar la información requerida mediante un archivo .pdf y permitir la impresión del documento.

Fuente: propia

c) Especificación de casos de uso

En esta etapa se desarrolla el diagrama de casos de uso de la aplicación web.

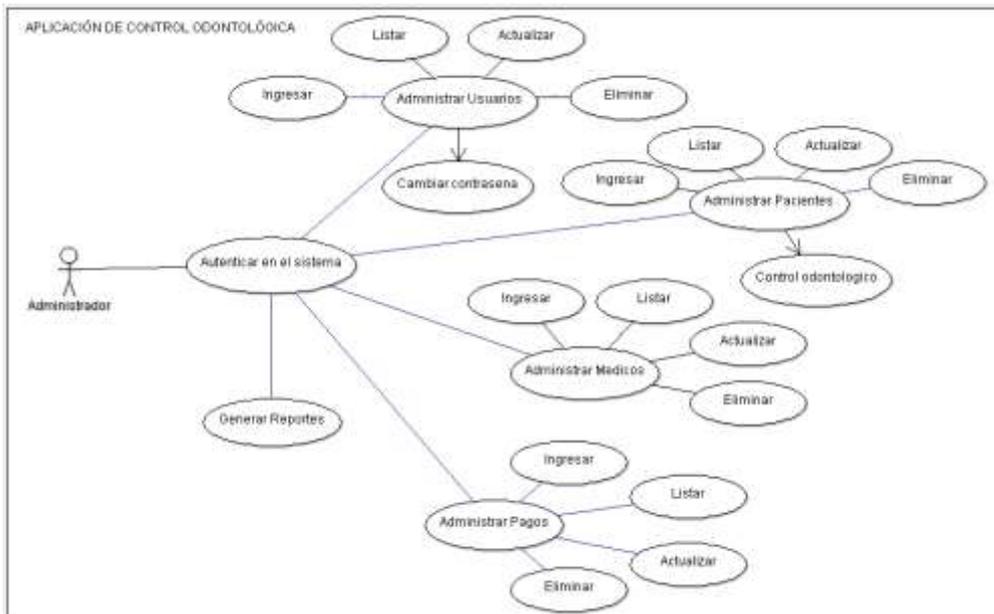


Fig. 15: Diagrama de casos de uso
Fuente: Propia

d) Especificación de UIs (Identificación de Usuarios)

Esta etapa permite la representación sencilla y descriptiva del diagrama de casos de uso desarrollado anteriormente.

UID Autenticación en el sistema



Fig. 16: UID caso de uso Autenticar en el sistema.
Fuente: Propia

Usuarios

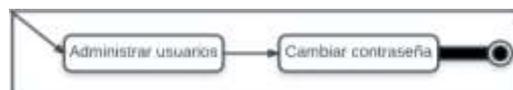


Fig. 17: UID caso de uso Administrar Usuarios.
Fuente: Propia

Pacientes

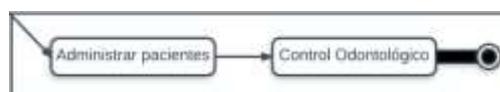


Fig. 18: UID caso de uso Administrar Pacientes.
Fuente: Propia

Médicos



Fig. 19: UID caso de uso Administrar Médicos.

Fuente: Propia

Pagos o procedimientos



Fig. 20: UID caso de uso Administrar Pagos o procedimientos.

Fuente: Propia

Reportes



Fig. 21: UID caso de uso Generar reportes.

Fuente: Propia

2. Diseño Conceptual

En esta fase se determina los diagramas de clases y secuencia de la aplicación web, de igual manera se presenta el esquema de la base de datos de la aplicación web.

a) Diagrama de clases

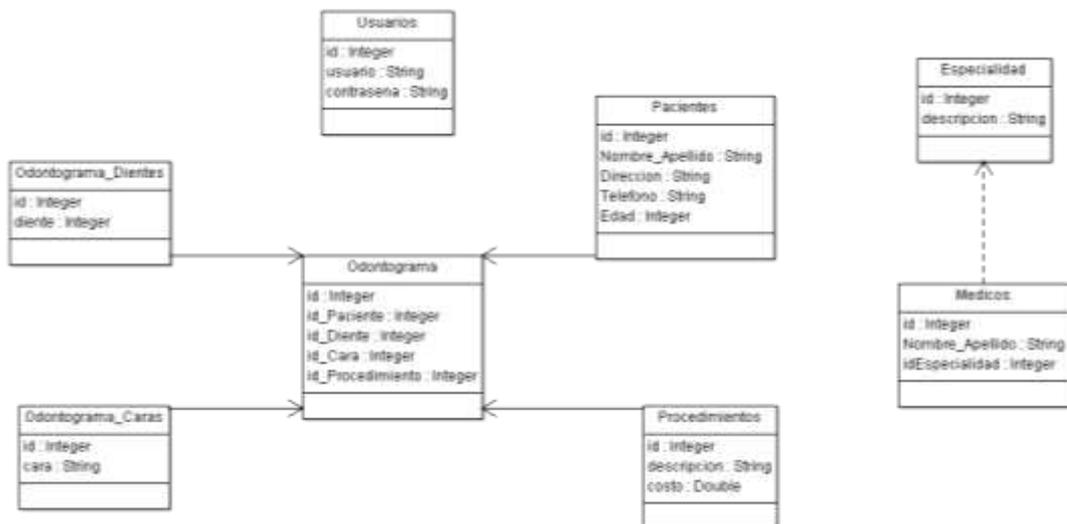


Fig. 22: Diagrama de clases

Fuente: Propia

b) Diagrama de secuencias

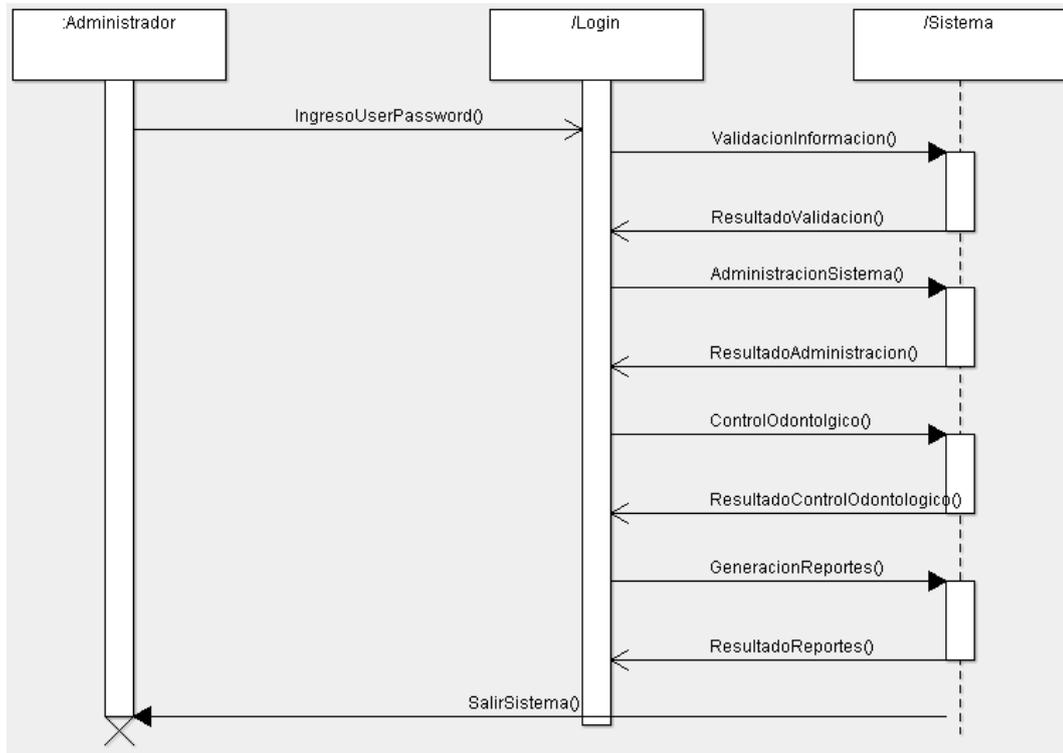


Fig. 23: Diagrama de secuencias

Fuente: Propia

3. Diseño Navegacional

En esta fase se define la estructura de navegación en el sistema, mediante un modelo navegacional que organiza el diseño y adapta la vista de acuerdo con la necesidad del usuario de la aplicación.

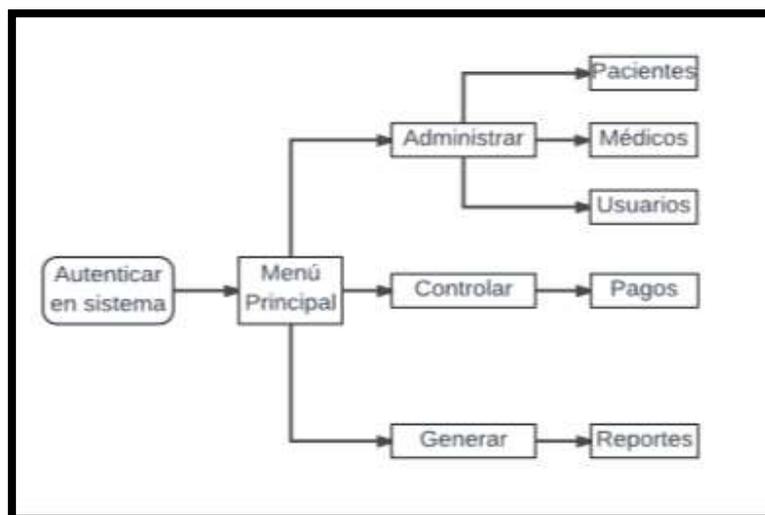


Fig. 24: Diagrama navegacional

Fuente: Propia

4. Diseño de la interfaz abstracta

Esta fase define la interfaz que tendrá la aplicación en base a los diagramas de Vistas de Datos Abstractos (ADVs) donde se incluye una vista para cada nodo navegacional, que fue establecido en el diagrama de navegación de la aplicación.

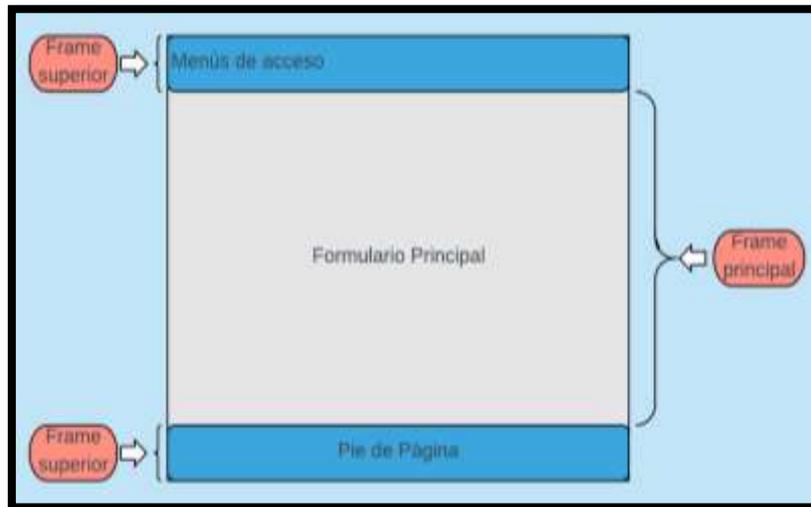


Fig. 25: ADV presentación de la aplicación web.
Fuente: Propia

5. Implementación

Después de haber levantado el modelo conceptual, navegacional, interfaz abstracta se procede a desarrollar la aplicación mediante el uso de las herramientas establecidas.

3.2.7 Prototipo de la metodología WSDM.

El modelo WSDM (Web Site Design Method) desarrolla las aplicaciones web mediante las siguientes fases:

1. Modelo de Usuario

Esta fase identifica a los usuarios que van a hacer uso de la aplicación web y sus funciones.

- **Clasificación de usuarios.**

En la aplicación se ha identificado a un único usuario:

Tabla 17: Clasificación de usuario

USUARIO	ACTIVIDAD
ADMINISTRADOR	Es el encargado general de la aplicación web.

Fuente: Propia

- **Descripción de usuarios.**

En usuario identificado como **Administrador** cumplirá las siguientes actividades.

- Acceso al sistema
- Control de usuarios.
- Control de médicos.
- Control de pacientes.
- Control odontológico.
- Control de procedimientos.
- Control de reportes.

2. Diseño conceptual

En esta fase el modelo WSDM integra el modelado de objetos y el diseño navegacional lo que en la metodología se conoce como el diseño conceptual y navegacional.

Modelo conceptual

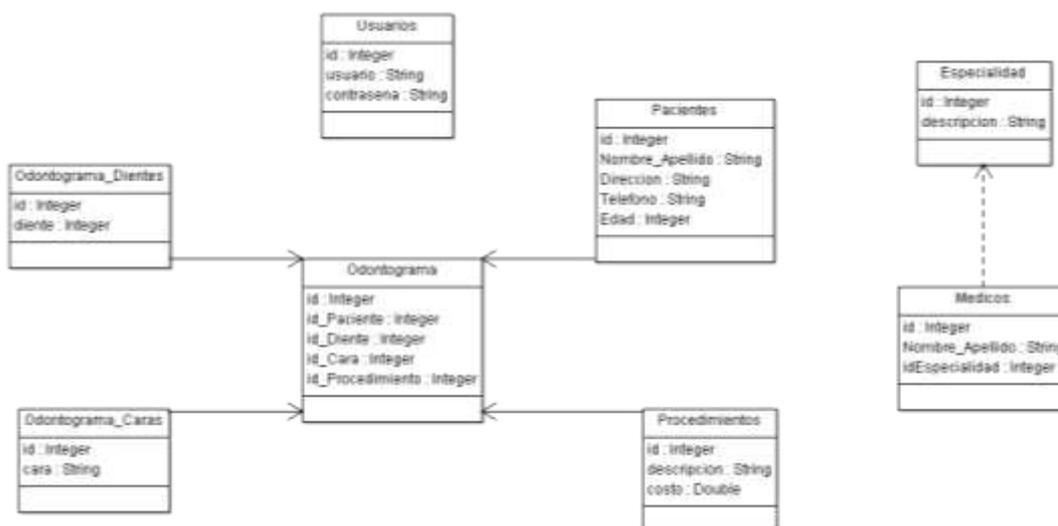


Fig. 26: Modelo conceptual

Fuente: Propia

Modelo navegacional

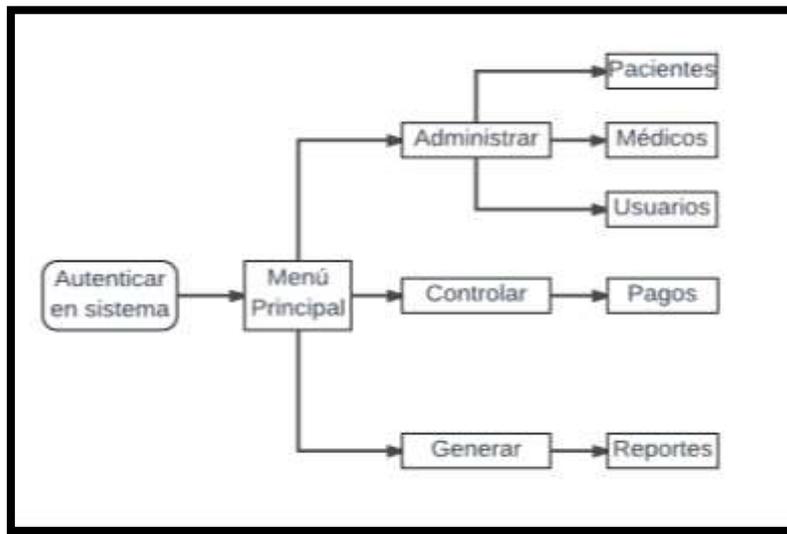


Fig. 27: Diagrama navegacional
Fuente: Propia

3. Diseño de Implementación

En esta fase se establece el diseño de la interfaz gráfica de la aplicación web.

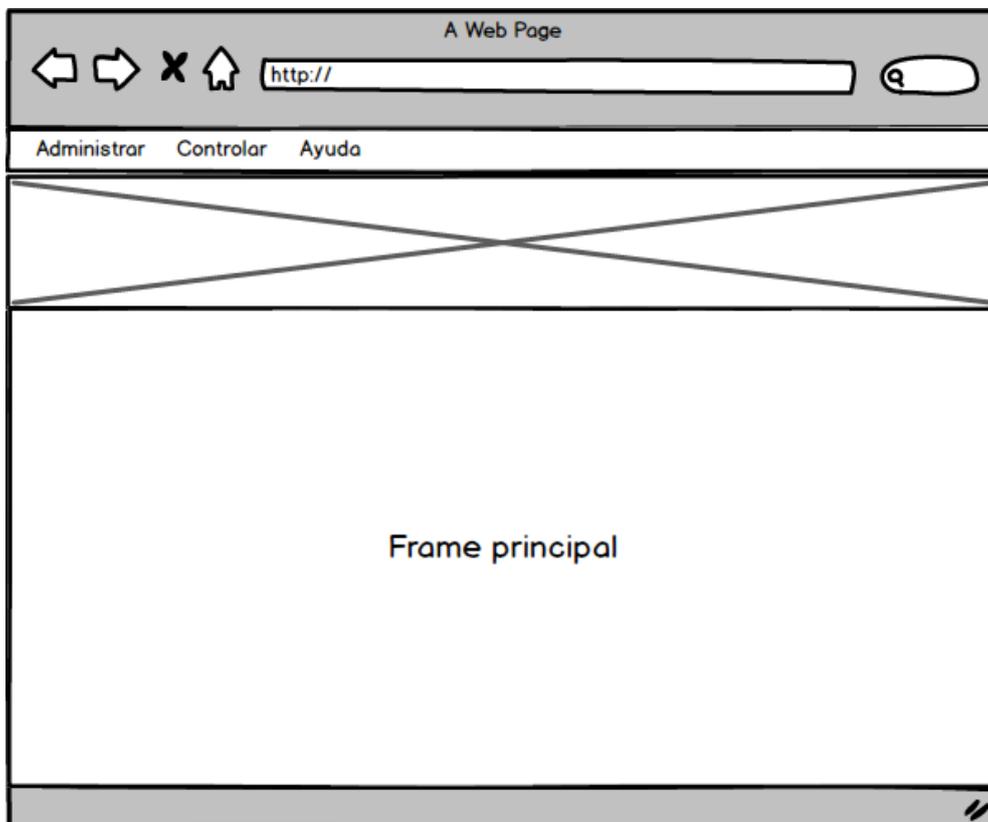


Fig. 28: Diagrama de Implementación
Fuente: Propia

4. Implementación

En esta etapa se diseña la interfaz gráfica que tendrá la aplicación web e inmediatamente se procede con la construcción del software deseado.

3.2.8 Prototipo de la metodología UWE.

El modelo UWE (UML Web Engineering) es un estándar web basado a la notación UML que establece las siguientes fases de desarrollo:

1. Especificación de Requerimientos

Esta fase propone el levantamiento de requerimientos de la aplicación web, además de integrar el diagrama de casos de uso para conocer el comportamiento de la aplicación y los usuarios que administrarán el software.

Requerimientos funcionales

La aplicación de control odontológica debe tener los siguientes requisitos funcionales.

- **Control de usuarios:** permitirá el acceso y control de la aplicación informática.
- **Control de pacientes:** debe permitir ingresar la información básica de los pacientes y almacenarla para ser utilizada de manera oportuna.
- **Control de médicos:** debe permitir almacenar la información de los galenos que trabajan en la institución.
- **Control de procedimientos:** permitirá almacenar los datos de costos por servicio prestado en el establecimiento.
- **Control de reportes:** permitirá obtener los datos impresos de acuerdo con la necesidad del usuario.

Diagrama de casos de uso

Permitirá identificar a los actores de la aplicación y las funciones que deben cumplir cada usuario.

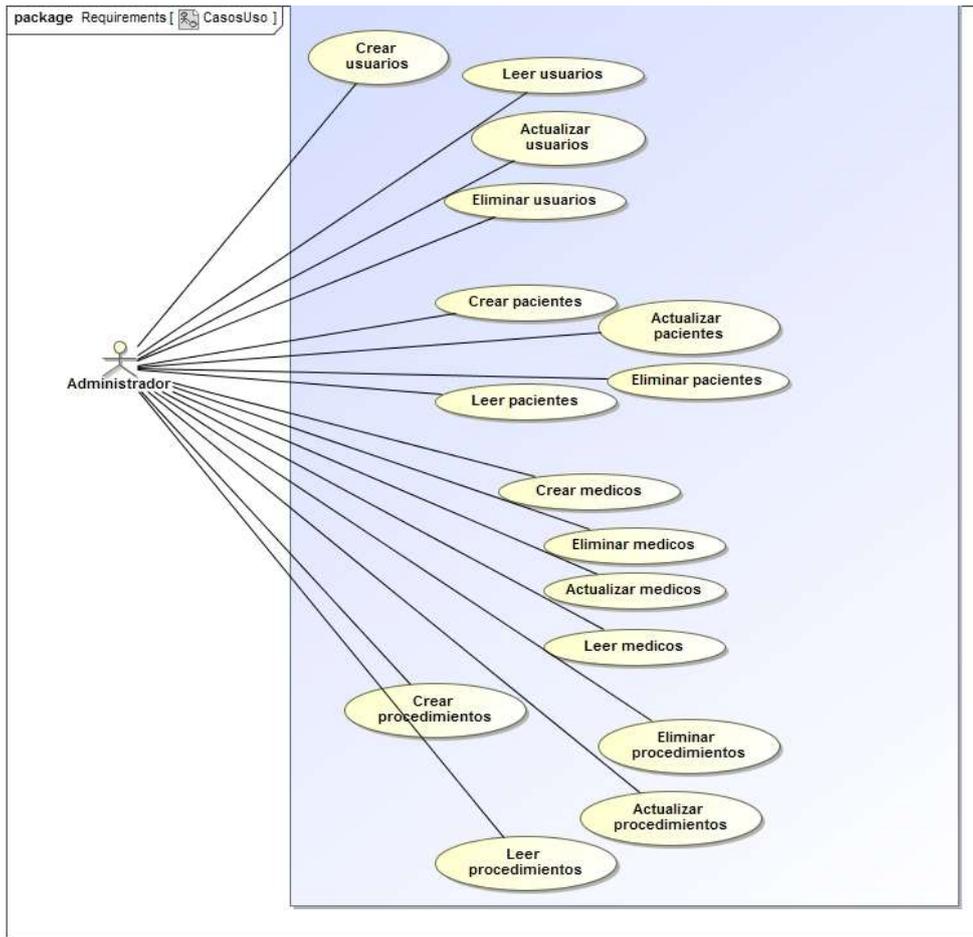


Fig. 29: Diagrama de casos de uso
Fuente: Propia

2. Modelo Lógico – Conceptual

En esta fase se genera el modelo de clases en base a los requisitos levantados en la fase anterior.

Diagrama de clases

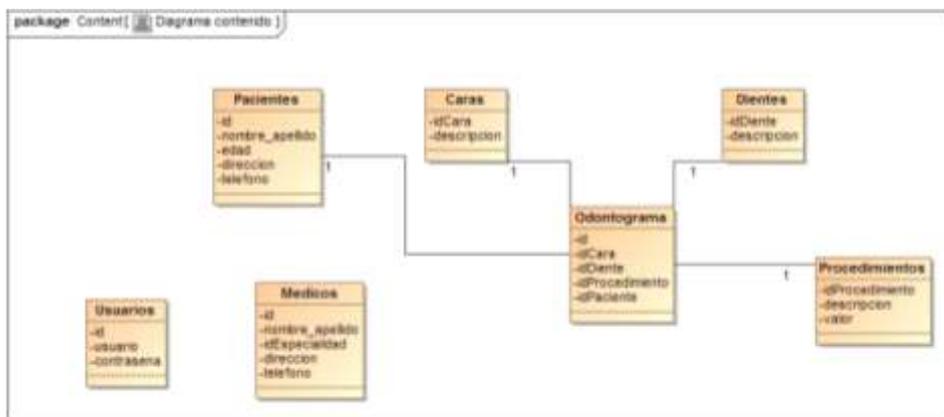


Fig. 30: Diagrama de clases
Fuente: Propia

3. Modelo Navegacional

Establece la estructura de navegación de la aplicación web, UWE utiliza la notación UML para el desarrollo del modelo.

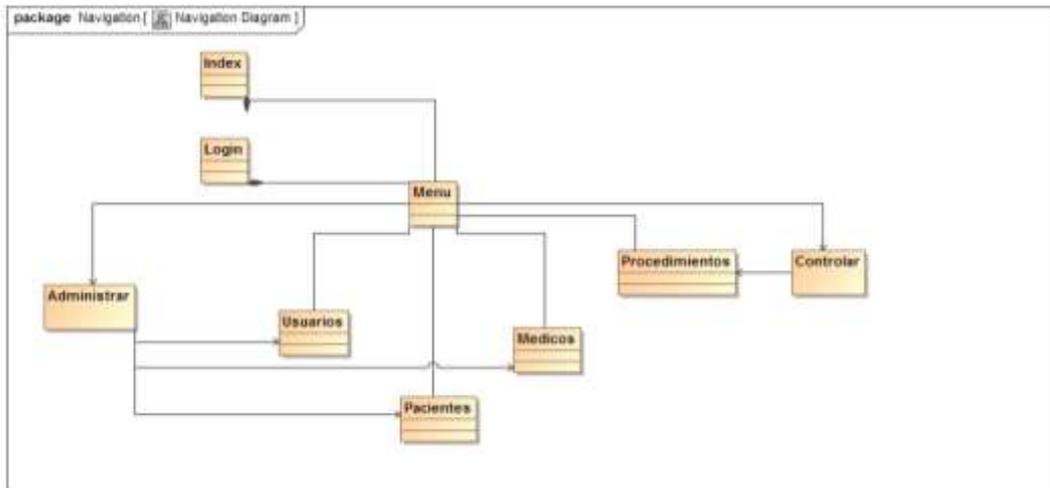


Fig. 31: Diagrama navegacional
Fuente: Propia

4. Modelo de Presentación

Esta fase establece una visión abstracta de interfaz de usuario de la aplicación, su proceso se desarrolla mediante el uso del diagrama de presentación, que integrará la IU principal de la web construida.

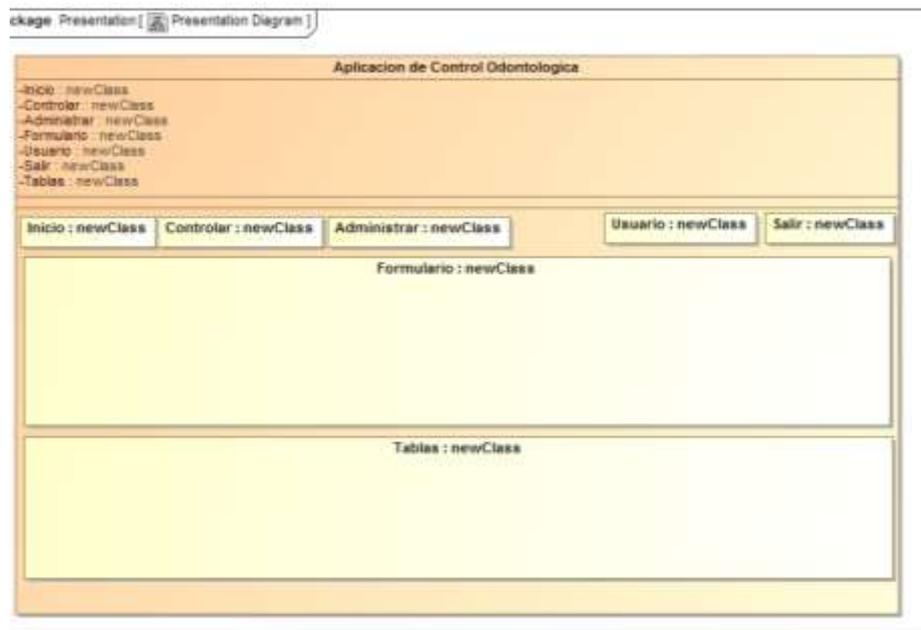


Fig. 32: Diagrama de presentación
Fuente: Propia

5. Modelo de Proceso

Representa el aspecto que tienen las acciones de las clases de proceso. En esta fase se desarrolla los diagramas de estructura de proceso y de flujo de proceso.

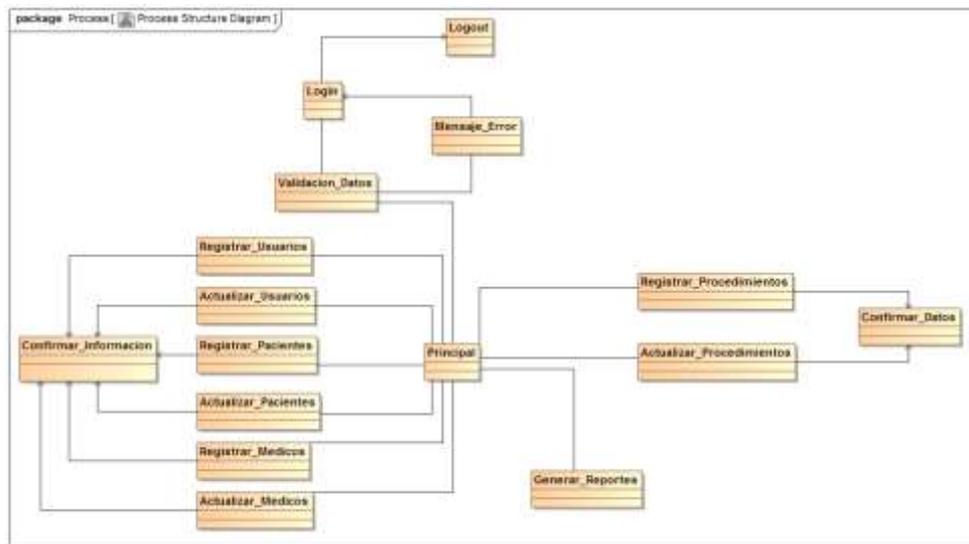


Fig. 33: Diagrama de estructura de proceso
Fuente: Propia

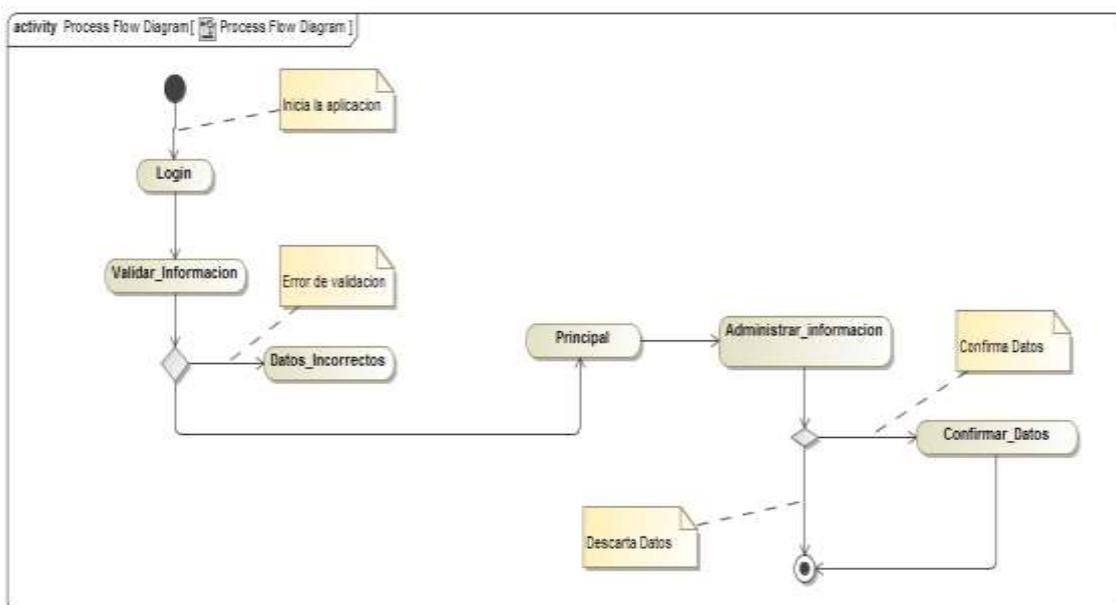


Fig. 34: Diagrama de flujo de proceso
Fuente: Propia

3.2.9 Resultados del desarrollo de prototipos.

Al utilizar los modelos OOHDM, WSDM, UWE en el desarrollo de prototipos del software odontológico, se pudo conocer las características, proceso, fases y sub fases que sigue cada metodología, debido a esto a se ha consolidado una amplitud de conocimientos en los

patrones analizados, por lo que es necesario realizar un análisis comparativo de los modelos, considerando parámetros de calificación en base a una guía de evaluación para la selección del mejor estándar metodológico.

3.3 Análisis Comparativo.

Para el análisis comparativo de modelos web se valorará criterios que están establecidos en un modelo de evaluación de metodologías de desarrollo de software, a su vez se utilizará parámetros de comparación de la norma ISO 9126 que es un modelo de control de calidad del software.

3.3.1 Ponderación y escalamiento de Likert.

Según el autor (Llauradó, 2014) el escalamiento de Likert es una herramienta de medición que permite valorar las actitudes y conformidades de una investigación, se utiliza principalmente para:

- Acordar el nivel de importancia y afirmación de un factor,
- Medir la frecuencia de una actividad,
- Valorar una actividad, producto o servicio,

Ventajas

- Fácil de desarrollar,
- Puede ajustar las opiniones,

Desventajas

- Puede existir resultados similares, aunque las elecciones sean diferentes,
- Existe dificultad al tratar resultados neutros (pág. 1).

La escala de Likert permite un valor de ponderación en rango de 1 a 5, siendo 5 el valor de mayor criterio en la matriz comparativa y en la tabla de características de los tres modelos investigados, al final se obtendrá un valor cuantitativo que permitirá establecer la mejor metodología web.

Tabla 18: Representación de la escala de Likert

ESCALA	NÚMERO DE RANGO	RANGOS	
		Descripción	Valor
ESCALA DE LIKERT	5	Totalmente de acuerdo	5
		De acuerdo	4
		Neutral	3
		En desacuerdo	2
		Totalmente en desacuerdo	1

Fuente: (Llauradó, 2014)

3.3.2 Modelo de evaluación en base a requisitos.

Los autores Rafael Menéndez – Barzanallana Asensio plantearon trece requisitos deseables para la construcción o selección de una metodología de desarrollo de software (Méndez, 2010).

Los trece requerimientos detallados en la siguiente tabla servirán como parámetros de análisis y comparación en la presente investigación.

Tabla 19: Requisitos deseables para la evaluación de metodologías

Nro.	Parámetro	Concepto
1	La metodología debe ajustarse a los objetivos.	Cada aproximación al desarrollo de software está basada en unos objetivos. Por ello la metodología que se elija debe recoger el aspecto filosófico de la aproximación deseada, es decir que los objetivos generales del desarrollo deben estar implementados en la metodología de desarrollo.
2	La metodología debe cubrir el ciclo entero de desarrollo de software.	Para ello la metodología ha de realizar unas etapas: investigación, análisis de requisitos, diseño.
3	La metodología debe integrar las distintas fases del ciclo de desarrollo.	Rastreabilidad. Es importante poder referirse a otras fases de un proyecto y fusionarlo con las fases previas. Es importante poder moverse no solo hacia adelante en el ciclo de vida, sino hacia atrás de forma que se pueda comprobar el trabajo realizado y se pueda efectuar correcciones. Fácil interacción entre etapas del ciclo de desarrollo. Es necesaria una validación formal de cada fase antes de pasar a la siguiente. La información que se pierde en una fase quedará perdida para siempre, con un impacto en el sistema resultante.
4	La metodología debe incluir la realización de validaciones.	La metodología debe detectar y corregir los errores cuanto antes. Uno de los problemas más frecuentes y costosos es el aplazamiento de la detección y corrección de problemas en las etapas finales del proyecto. Cuando más tarde sea detectado el error más caro será corregirlo. Por lo tanto, cada fase del proceso de desarrollo de software deberá incluir una actividad de validación implícita.

5	La metodología debe soportar la determinación de la exactitud del sistema a través del ciclo de desarrollo.	La exactitud del sistema implica muchos asuntos, incluyendo la correspondencia entre el sistema y sus especificaciones, así como que el sistema cumple con las necesidades del usuario. Por ejemplo, los métodos usados para el análisis y especificación del sistema deberían colaborar a terminar con el problema del entendimiento entre los informáticos, los usuarios y otras partes implicadas. Esto implica una comunicación entre el usuario y técnico amigable y sencillo, exento de consideraciones técnicas.
6	La metodología debe ser la base de una comunicación efectiva.	Debe ser posible gestionar a los informáticos, y éstos deben ser capaces de trabajar conjuntamente. Ha de haber una comunicación efectiva entre analistas, programadores, usuarios y gestores, con pasos bien definidos para realizar progresos visibles durante la actividad del desarrollo.
7	La metodología debe funcionar en un entorno dinámico orientado al usuario.	A lo largo de todo el ciclo de vida del desarrollo se debe producir una transferencia de conocimientos hacia el usuario. La clave del éxito es que todas las partes implicadas han de intercambiar información libremente. La participación del usuario es de importancia vital debido a que sus necesidades evolucionan constantemente. Por otra parte, la adquisición de conocimientos del usuario permitirá la toma de decisiones correctas. Para involucrar al usuario en el análisis, diseño y administración de datos, es aconsejable el empleo de técnicas estructuradas lo más sencillas posible. Para esto, es esencial contar con una buena técnica de diagramación.
8	La metodología debe especificar claramente los responsables de resultados.	Debe especificar claramente quienes son los participantes de cada tarea a desarrollar, debe detallar de una manera clara los resultados de los que serán responsables.
9	La metodología debe emplearse en un entorno amplio de proyectos software.	Una empresa deberá adoptar una metodología que sea útil para un gran número de sistemas que vaya a construir. Por esta razón no es práctico adoptar varias metodologías en una misma empresa. Tamaño, vida. Las metodologías deberán ser capaces de abordar sistemas de distintos tamaños y rangos de vida. Complejidad. La metodología debe servir para sistemas de distinta complejidad, es decir puede abarcar un departamento, varios departamentos o varias empresas. Entorno, La metodología debe servir con independencia de la tecnología disponible en la empresa.
10	La metodología debe ser posible enseñarse.	Incluso en una organización sencilla, serán muchas las personas que la van a utilizar, incluso los que se incorporen posteriormente a la empresa. Cada persona debe entender las técnicas específicas de la metodología, los procedimientos organizativos y de gestión que la hacen efectiva, las herramientas automatizadas que soportan la metodología y las motivaciones que subyacen en ella.
11	La metodología debe estar soportada por herramientas CASE.	La metodología debe estar soportada por herramientas automatizadas que mejoren la productividad, tanto del ingeniero de software en particular, como la del desarrollo en general. El uso de estas herramientas reduce el número de personas requeridas y la sobrecarga de comunicación, además

		de ayudar a producir especificaciones y diseños con menos errores, más fáciles de probar, modificar y usar.
12	La metodología debe soportar la eventual evolución del software.	Normalmente durante su tiempo de vida los sistemas tienen muchas versiones, pudiendo durar incluso más de 10 años. Existen herramientas CASE para la gestión de la configuración y otras denominadas "Ingeniería inversa" para ayudar en el mantenimiento de los sistemas no estructurados, permitiendo estructurar componentes de éstos facilitando así su mantenimiento.
13	La metodología debe contener actividades contundentes a mejorar el proceso de desarrollo de software.	Para mejorar el proceso es básico disponer de datos numéricos que evidencian la efectividad de la aplicación del proceso con respecto a cualquier producto software resultante del proceso. Para disponer de estos datos, la metodología debe contener un conjunto de mediciones de proceso para identificar la calidad y coste asociado de cada etapa del proceso. Sería ideal el uso de herramientas CASE.

Fuente: (Méndez, 2010)

3.3.3 ISO/IEC 9126.

La norma ISO/IEC 9126 Software Product Evaluation (Evaluación de los productos de software), ha establecido un estándar internacional para la evaluación de la calidad de los productos de software, el cual presenta seis características de calidad y los lineamientos para su uso, con el fin de dar soporte a las necesidades (SENA, 2015).

La Figura 35. Presenta los lineamientos de evaluación de calidad de la norma ISO/IEC 9126.

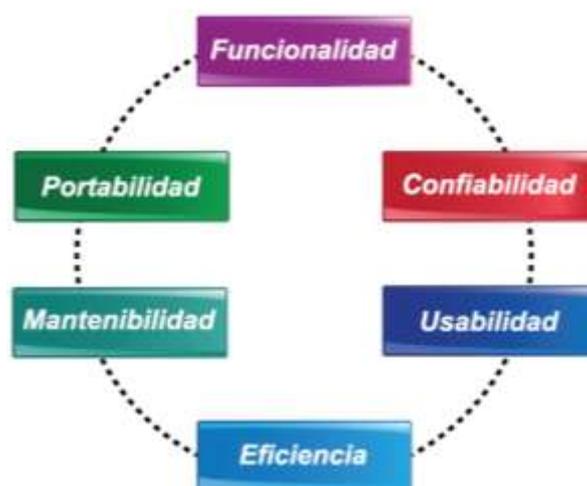


Fig. 35: Atributos de la norma ISO/IEC 9126

Fuente: (SENA, 2015)

Análisis comparativo de parámetros

La Tabla 20 presenta una comparativa de las principales características de calidad de la norma ISO/IEC 9126, para una posterior selección de parámetros que ayuden en el análisis comparativo de la investigación en curso.

Tabla 20: Comparativa de las características de la norma ISO/IEC 9126

NRO.	PARÁMETRO	INDICADOR	APLICA	NO APLICA
1	Funcionalidad	Adecuación	✓	
		Exactitud		x
		Operabilidad		x
		Seguridad de acceso		x
		Cumplimiento de Funcionalidad		x
2	Confiabilidad (Fiabilidad)	Madurez		x
		Tolerancia a fallos		x
		Recuperabilidad		x
		Cumplimiento de Fiabilidad		x
3	Factibilidad de uso (Usabilidad)	Capacidad de ser entendido	✓	
		Capacidad de ser aprendido	✓	
		Capacidad de ser operado	✓	
		Capacidad de atracción	✓	
		Cumplimiento de usabilidad	✓	
4	Eficiencia	Comportamiento temporal	✓	
		Utilización de recursos	✓	
		Cumplimiento de eficiencia	✓	
5	Mantenibilidad	Facilidad de análisis		x
		Estabilidad		x
		Facilidad de prueba		x
		Capacidad de ser cambiado		x
		Adaptabilidad		x
6	Portabilidad	Facilidad de instalación		x
		Coexistencia		x
		Reemplazabilidad	✓	

Fuente: propia

Resultados del análisis comparativo de parámetros

Al analizar los lineamientos de calidad de la norma ISO/IEC 9126 se pudo determinar que los parámetros Usabilidad y Eficiencia pueden ser utilizados en el análisis comparativo de la investigación, gracias a que todos sus indicadores pueden ser aplicados en la comparación de las metodologías estudiadas, contrario a los indicadores de los demás parámetros, los cuales en su mayoría no son factores medibles en la valoración de metodologías y por lo cual, no pueden ser aplicados al estudio comparativo de metodologías web.

Parámetros

De acuerdo con los resultados del análisis comparativo de características de la norma ISO/IEC 9126, se establece utilizar los parámetros Usabilidad y Eficiencia para la complementación del estudio comparativo de metodologías web. A continuación, la Tabla 21, describe los parámetros y sus conceptos de evaluación.

Tabla 21: Parámetros de evaluación

NRO.	PARÁMETRO	CONCEPTO
1	Eficiencia	Mide los tiempos de desarrollo, utilización de recursos y disponibilidad de documentación.
2	Usabilidad	Mide la facilidad del uso y acompañamiento de la metodología.

Fuente: (Velarde & Pilco, 2014)

Indicadores

En base a los parámetros seleccionados anteriormente, se presenta en la siguiente tabla los indicadores para la evaluación de las metodologías web.

Tabla 22: Indicadores de Eficiencia

INDICADOR	CONCEPTO
COMPORTAMIENTO TEMPORAL	Tiempos de proceso utilizado en el desarrollo del modelo.
UTILIZACIÓN DE RECURSOS	Uso de recursos adecuados para cumplir las funciones del modelo.
CUMPLIMIENTO DE EFICIENCIA	Capacidad de adaptación a normas relacionadas con la eficiencia.

Fuente: (SENA, 2015)

Tabla 23: Indicadores de Usabilidad

INDICADOR	CONCEPTO
CAPACIDAD PARA SER ENTENDIDO	Permite al usuario entender las tareas y condiciones del modelo.
CAPACIDAD PARA SER APRENDIDO	La facilidad de aprender sobre el modelo
CAPACIDAD PARA SER OPERADO	Permitir al usuario controlar con facilidad al modelo.
CAPACIDAD DE ATRACCIÓN	Aceptación del modelo por el usuario.
CUMPLIMIENTO DE USABILIDAD	Capacidad de adaptación de normas, guías de estilo relacionadas con la usabilidad.

Fuente: (SENA, 2015)

3.3.4 Criterios de valoración.

La ponderación determinada para la evaluación de metodologías de desarrollo de software se obtiene mediante valores porcentuales establecidos en la escala de Likert, a continuación, se presentan las valoraciones para el análisis comparativo:

Tabla 24: Escala de valoración para la evaluación de metodologías

DESCRIPCIÓN	RANGO	VALOR PORCENTUAL	PONDERACIÓN
FORTALEZA IMPORTANTE	5	100%	1
FORTALEZA MENOR	4	80%	0,8
NEUTRAL	3	60%	0,6
DEBILIDAD MENOR	2	40%	0,4
DEBILIDAD GRAVE	1	20%	0,2

Fuente: propia

3.3.5 Comparativa de metodologías web

Actualmente existen diversos modelos basados en la web, que son aplicados en el desarrollo de software, tal es el caso de las metodologías OOHDM, WSDM, UWE que son parte del análisis comparativo de esta investigación.

Según (Calva & Romero, 2010) con la aparición del paradigma de orientación a objetos se dio campo a otras metodologías como OOHDM, que trata de abarcar en forma general el ciclo de desarrollo y ha tenido una gran aceptación, por las nuevas tendencias que incorpora, como el hecho de separar el modelo conceptual, del navegacional y la interfaz abstracta de manera independiente. La metodología WSDM que se centra en el análisis y estudio de los grupos de usuarios, mientras que las otras se centran en realizar un modelo de clases a alto nivel para representar el modelo conceptual del sistema. Por otra parte, UWE mantiene la técnica de modelado orientado a objetos, pero introduciendo una notación estándar basada en el uso de UML para cada etapa del estándar metodológico (pág. 5).

Los autores (Calva & Romero, 2010) presentan una comparación de los modelos analizados, especificando varias características donde se detallan su proceso, técnicas de modelado, representación gráfica, notación y herramientas de soporte.

Tabla 25: Características generales de las metodologías

	PROCESO	TÉCNICA DE MODELO	REPRESENTACIÓN GRÁFICA	NOTACIÓN	HERRAMIENTA SOPORTE
OOHDM	Análisis de requerimientos Diseño conceptual Diseño navegacional Diseño abstracto de la UI Implementación	O-O	Diagramas de clases Diagrama navegacional, clase más contexto Diagrama de configuración de ADV Diagrama ADV	OMT UML Propio	OOHDM – Web (Sin soporte)
WSDM	Modelo del usuario Diseño conceptual Modelo de objetos Diseño navegacional Diseño de Implementación Implementación	E-R/O-O	Diagrama E -R Capas de navegación	E -R OMT Propio	No tiene
UWE	Análisis de requisitos Modelo de contenido Modelo de navegación Modelo de presentación Modelo de proceso	O-O	Diagrama de casos de uso Diagrama de clases Diagrama de presentación	UML	MagicUWE

Fuente:(Calva, 2010)

A su vez los autores (Calva & Romero, 2010) enlistan las ventajas y desventajas de las metodologías analizadas en la siguiente tabla:

Tabla 26: Ventajas y desventajas de las metodologías web propuestas.

	VENTAJAS	DESVENTAJAS
OOHDM	<ul style="list-style-type: none"> - Toma en cuenta el análisis de requisitos. - Utiliza un alto nivel de especificación de lenguaje para diseñar aplicaciones web. - Incluye una explícita noción de la vista del sitio. - Cubre avanzados aspectos del modelado del sitio web. - Efectivo en soportar múltiples dispositivos de sitios web. 	<ul style="list-style-type: none"> - Deja a un lado la funcionalidad del sistema. - No ofrece mecanismos para trabajar con múltiples actores. - Solo para aplicaciones sencillas, donde la complejidad funcional sea mínima.
WSDM	<ul style="list-style-type: none"> - Toma en cuenta el análisis de requisitos - Ofrece visión del tratamiento de usuarios. - Diseña la aplicación en base a grupos de usuarios. 	<ul style="list-style-type: none"> - Limitada para el desarrollo de kioscos web. - No trabaja aspectos como funcionalidad, seguridad
UWE	<ul style="list-style-type: none"> - Toma en cuenta el análisis de requisitos. - Útil en el desarrollo de aplicaciones de gran tamaño, así como sistemas sencillos. - Utiliza una notación estándar en base al uso de UML. - Establece un formalismo más estricto. 	<ul style="list-style-type: none"> - Uso de restricciones escritas.

Fuente: (Calva & Romero, 2010)

3.3.6 Evaluación mediante los parámetros del modelo de requisitos.

Mediante la matriz de evaluación de metodologías se hará una comparativa detallada de los requisitos deseables para seleccionar un estándar adecuado. Posterior al análisis se aplicará los criterios de valoración a los parámetros estudiados, con el fin de obtener resultados numéricos que distingan el modelo web más sobresaliente.

Tabla 27: Matriz de análisis de metodologías

NRO.	PARÁMETRO	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
1	La metodología debe ajustarse a los objetivos.	Cumple con los objetivos.	Cumple con los objetivos.	Cumple con los objetivos.
2	La metodología debe cubrir el ciclo entero de desarrollo de software.	Cubre casi todo el ciclo del software, a excepción de la fase del mantenimiento.	No cumple con todas las fases del ciclo de desarrollo de software.	No cumple con todas las fases del ciclo de desarrollo del software.
3	La metodología debe integrar las distintas fases del ciclo de desarrollo.	Integra las fases de desarrollo.	Integra las fases de desarrollo.	Integra las fases de desarrollo.
4	La metodología debe incluir la realización de validaciones.	No incluye ninguna actividad de validación.	No incluye ninguna actividad de validación.	No incluye ninguna actividad de validación.
5	La metodología debe soportar la determinación de la exactitud del sistema a través del ciclo de desarrollo.	Soporta la determinación de la exactitud del sistema, gracias al desarrollo detallado de las fases y sub fases de la metodología.	No soporta la determinación de la exactitud del sistema.	Soporta la determinación de la exactitud del sistema, gracias al desarrollo detallado de las fases de la metodología.
6	La metodología debe ser la base de una comunicación efectiva.	Permite una comunicación medianamente efectiva, debido a la notación que utiliza el modelo.	Permite una comunicación medianamente efectiva, debido a la notación que utiliza el modelo.	Permite una comunicación totalmente efectiva, gracias a su notación UML.
7	La metodología debe funcionar en un entorno dinámico orientado al usuario.	Emplea técnicas medianamente sencillas, debido a su norma de diagramación propia.	Emplea técnicas medianamente sencillas, debido a su norma de diagramación propia.	Emplea técnicas claras y sencillas gracias al uso de un lenguaje estándar de diagramación.
8	La metodología debe especificar claramente los responsables de resultados.	Muy específica.	Medianamente específica.	Muy específica.
9	La metodología debe emplearse en un entorno amplio de proyectos software.	No adaptable, pues es una metodología orientada a proyectos pequeños.	Adaptable.	Adaptable.
10	La metodología debe ser posible enseñarse.	Permite la enseñanza medianamente, debido al desconocimiento de sus modelos.	Permite la enseñanza medianamente, debido al desconocimiento de sus modelos.	Permite la enseñanza totalmente, pues su lenguaje de modelado es universalmente funcional.
11	La metodología debe estar soportada por herramientas CASE.	Sin soporte, ya que en la actualidad ya no se utiliza la herramienta OOHDM-Web.	No tiene	MagicUWE

NRO	PARÁMETRO	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
12	La metodología debe soportar la eventual evolución del software.	No soporta la evolución del sistema.	No soporta la evolución del sistema.	Soporta la evolución eventual del sistema.
13	La metodología debe contener actividades contundentes a mejorar el proceso de desarrollo de software.	No contiene.	No contiene.	No contiene.

Fuente: propia

Tabla 28: Valoración cualitativa de metodologías

NRO.	PARÁMETRO	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
1	La metodología debe ajustarse a los objetivos.	Fortaleza importante.	Fortaleza menor.	Fortaleza importante.
2	La metodología debe cubrir el ciclo entero de desarrollo de software.	Fortaleza menor.	Debilidad menor.	Debilidad menor.
3	La metodología debe integrar las distintas fases del ciclo de desarrollo.	Fortaleza importante.	Fortaleza importante.	Fortaleza importante.
4	La metodología debe incluir la realización de validaciones.	Debilidad grave.	Debilidad grave.	Debilidad grave.
5	La metodología debe soportar la determinación de la exactitud del sistema a través del ciclo de desarrollo.	Fortaleza importante.	Fortaleza menor.	Fortaleza importante.
6	La metodología debe ser la base de una comunicación efectiva.	Fortaleza menor.	Fortaleza menor.	Fortaleza importante.
7	La metodología debe funcionar en un entorno dinámico orientado al usuario.	Fortaleza menor.	Fortaleza menor.	Fortaleza importante.
8	La metodología debe especificar claramente los responsables de resultados.	Fortaleza importante.	Fortaleza menor.	Fortaleza importante.
9	La metodología debe emplearse en un entorno amplio de proyectos software.	Debilidad grave.	Fortaleza importante.	Fortaleza importante.
10	La metodología debe ser posible enseñarse.	Fortaleza menor.	Fortaleza menor.	Fortaleza importante.
11	La metodología debe estar soportada por herramientas CASE.	Debilidad menor.	Debilidad grave.	Fortaleza importante.
12	La metodología debe soportar la eventual evolución del software.	Debilidad menor.	Debilidad menor.	Fortaleza menor.
13	La metodología debe contener actividades contundentes a mejorar el proceso de desarrollo de software.	Debilidad grave.	Debilidad grave.	Debilidad menor.

Fuente: propia

De acuerdo con los criterios establecidos por modelo de evaluación y después de realizar el respectivo análisis comparativo de las propuestas seleccionadas se establece la siguiente matriz de evaluación:

Tabla 29: Valoración cuantitativa de metodologías

FACTORES DE EVALUACIÓN	METODOLOGÍAS INVESTIGADAS					
	OOHDM		WSDM		UWE	
	Rango	Ponderación	Rango	Ponderación	Rango	Ponderación
LA METODOLOGÍA DEBE AJUSTARSE A LOS OBJETIVOS.	5	1,0	4	0,8	5	1,0
LA METODOLOGÍA DEBE CUBRIR EL CICLO ENTERO DE DESARROLLO DE SOFTWARE.	4	0,8	2	0,4	2	0,4
LA METODOLOGÍA DEBE INTEGRAR LAS DISTINTAS FASES DEL CICLO DE DESARROLLO.	5	1,0	5	1,0	5	1,0
LA METODOLOGÍA DEBE INCLUIR LA REALIZACIÓN DE VALIDACIONES.	1	0,2	1	0,2	1	0,2
LA METODOLOGÍA DEBE SOPORTAR LA DETERMINACIÓN DE LA EXACTITUD DEL SISTEMA A TRAVÉS DEL CICLO DE DESARROLLO.	5	1,0	4	0,8	5	1,0
LA METODOLOGÍA DEBE SER LA BASE DE UNA COMUNICACIÓN EFECTIVA.	4	0,8	4	0,8	5	1,0
LA METODOLOGÍA DEBE FUNCIONAR EN UN ENTORNO DINÁMICO ORIENTADO AL USUARIO.	4	0,8	4	0,8	5	1,0
LA METODOLOGÍA DEBE ESPECIFICAR CLARAMENTE LOS RESPONSABLES DE RESULTADOS.	5	1,0	4	0,8	5	1,0
LA METODOLOGÍA DEBE EMPLEARSE EN UN ENTORNO AMPLIO DE PROYECTOS SOFTWARE.	1	0,2	5	1,0	5	1,0
LA METODOLOGÍA DEBE SER POSIBLE ENSEÑARSE.	4	0,8	2	0,4	5	1,0
LA METODOLOGÍA DEBE ESTAR SOPORTADA POR HERRAMIENTAS CASE.	2	0,4	1	0,2	5	1,0
LA METODOLOGÍA DEBE SOPORTAR LA EVENTUAL EVOLUCIÓN DEL SOFTWARE.	2	0,4	2	0,4	4	0,8
LA METODOLOGÍA DEBE CONTENER ACTIVIDADES CONTUNDENTES A MEJORAR EL PROCESO DE DESARROLLO DE SOFTWARE.	1	0,2	1	0,2	2	0,2
TOTAL		8,6		7,8		10,6

Fuente: propia

3.3.7 Interpretación de resultados del modelo de requisitos.

Posterior al análisis y evaluación de metodologías web en base al modelo de requisitos se puede definir los siguientes resultados:

- **La metodología debe ajustarse a los objetivos.** Debido a que las metodologías OOHDM y UWE contienen en sus fases una clara descripción de los objetivos de la aplicación a desarrollarse, han sido puntuadas en mayor escala a diferencia de la metodología WSDM que su cumplimiento de objetivos es relativamente menor.
- **La metodología debe cubrir el ciclo entero de desarrollo de software.** Considerando que solamente OOHDM cumple con la mayoría de las fases del ciclo de desarrollo de software tales como análisis, diseño, implementación y pruebas, ha resultado mayor puntuada con relación a UWE y WSDM, que no cumplen con el requisito deseado.
- **La metodología debe integrar las distintas fases del ciclo de desarrollo.** Gracias a que OOHDM, WSDM, UWE utiliza fases secuenciales se pudo argumentar que las tres propuestas son aceptables, dando una valoración igual a los respectivos modelos.
- **La metodología debe incluir la realización de validaciones.** Una de las desventajas que tienen las metodologías analizadas no cumplen con una actividad de validación en las fases de desarrollo.
- **La metodología debe soportar la determinación de la exactitud del sistema a través del ciclo de desarrollo.** En este parámetro se valorizan mayormente a los modelos OOHDM y UWE debido a la clara especificación de requerimientos y el uso de diagramas de casos de uso, a diferencia del modelo WSDM y su deficiencia en el análisis de requisitos.
- **La metodología debe ser la base de una comunicación efectiva.** Debido a su notación estándar basada en UML el modelo UWE lleva ventaja a las demás metodologías obteniendo un mejor puntaje en relación de las otras propuestas que utilizan una notación propia lo que dificulta al grupo de trabajo tener una comunicación efectiva.
- **La metodología debe funcionar en un entorno dinámico orientado al usuario.** Dado que UWE utiliza técnicas estándares de diagramación resultó ser la metodología

mejor calificada, ya que los modelos WSDM y OOHDM utilizan técnicas personalizadas que dificultan la interpretación de las mismas y por lo tal se realiza más compleja la transferencia de conocimientos al usuario.

- **La metodología debe especificar claramente los responsables de resultados.** Las metodologías UWE y OOHDM son los estándares mejor puntuados debido a la descripción detallada de las fases y sub fases que ayudan en la identificación de los responsables de desarrollo en cada etapa.
- **La metodología debe emplearse en un entorno amplio de proyectos software.** La investigación demuestra que UWE y WSDM son metodologías orientados a proyectos grandes por lo cual han resultados mejor valorados, a diferencia de OOHDM que es un modelo orientado al desarrollo de aplicaciones pequeñas.
- **La metodología debe ser posible enseñarse.** Debido a que las técnicas y procedimientos estandarizados de UWE pueden ser comprendidas por distintos desarrolladores se ponderó al modelo en mayor escala con relación a OOHDM y WSDM.
- **La metodología debe estar soportada por herramientas CASE.** La metodología UWE es el estándar mejor puntuado debido a que tiene su propia herramienta CASE, mientras que OOHDM retiro su software OOHDM – Web y WSDM no contó en ninguna situación con soporte CASE.
- **La metodología debe soportar la eventual evolución del software.** Gracias al soporte de herramientas se puede considerar que UWE permitirá realizar el mantenimiento del software por lo que se dio una valoración mayor en comparación a las demás propuestas metodológicas.
- **La metodología debe contener actividades contundentes a mejorar el proceso de desarrollo de software.** Entre las debilidades que muestran OOHDM, WSDM y UWE se considera la falta de un conjunto de mediciones de proceso que no permite evaluar la calidad en cada etapa desarrollada por la metodología.

3.3.8 Valoración porcentual del modelo de requisitos.

Posterior al análisis y evaluación de metodologías web en base al modelo de requisitos se puede definir la siguiente fórmula que determinará las valoraciones de cada metodología:

$$\text{MODELO} = \frac{\text{Total valor ponderacion} * 100}{\text{Nro de requisitos}} \%$$

$$\text{OOHDM} = \frac{8,6 * 100}{13}$$

$$\text{OOHDM} = 66,15\%$$

$$\text{WSDM} = \frac{7,8 * 100}{13}$$

$$\text{WSDM} = 60\%$$

$$\text{UWE} = \frac{10,6 * 100}{13}$$

$$\text{UWE} = 81,54\%$$

Las valoraciones porcentuales correspondientes al modelo de requisitos se resumen en la siguiente tabla:

Tabla 30: Valoración de metodologías

METODOLOGÍA	VALOR PORCENTUAL
OOHDM	66,15%
WSDM	60%
UWE	81,54%

Fuente: propia

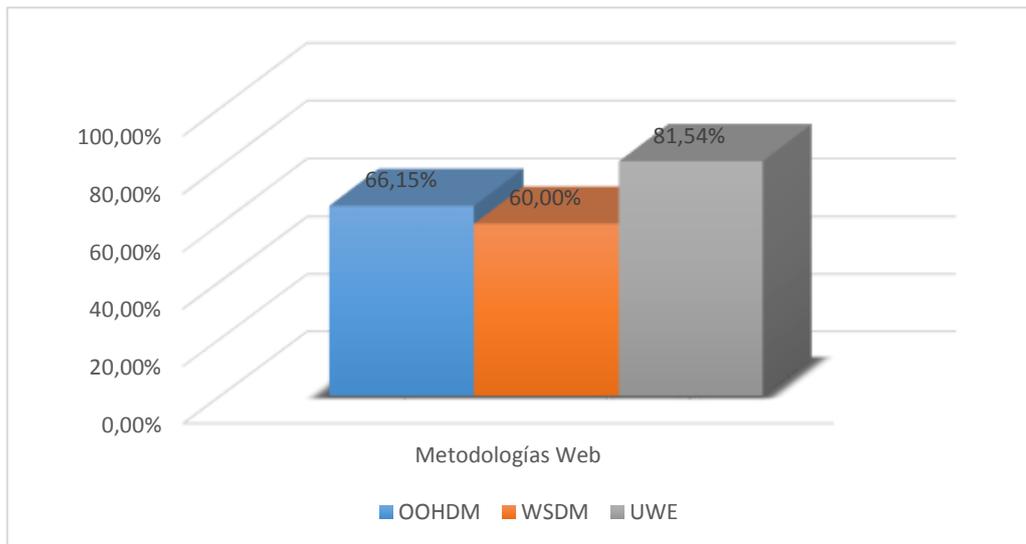


Fig. 36: Valoración porcentual del modelo de requisitos
Fuente: Propia

3.3.9 Resultados del análisis comparativo del modelo de requisitos.

De acuerdo con el modelo de requisitos se puede determinar que, la metodología OOHDM ha obtenido una valoración del 66,15%, frente al estándar WSDM que consiguió el porcentaje del 60% y el modelo UWE que alcanzó el valor del 81,54%.

3.3.10 Evaluación del parámetro Eficiencia.

Mediante el parámetro Eficiencia se realizará un análisis de indicadores, los cuales serán valorados cuantitativa y cualitativamente con el fin de seleccionar la metodología más adecuada para el desarrollo de aplicaciones web, la investigación se presenta en la siguiente tabla:

Tabla 31: Matriz de análisis del parámetro Eficiencia

NRO.	INDICADOR	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
1	Comportamiento temporal.	Demanda de mayor tiempo de desarrollo debido a la falta de herramientas CASE, desconocimiento del modelo y notación propia.	Demanda de mayor tiempo de desarrollo debido a la falta de herramientas CASE, desconocimiento del modelo y notación propia.	Menor tiempo gracias a su herramienta CASE.
2	Utilización de recursos.	Debido a su notación utiliza más recursos (humano, herramientas)	Debido a su notación utiliza más recursos (humano, herramientas)	Utiliza menos recursos debido a notación estándar.

3	Cumplimiento de eficiencia.	No podría adherirse a normas de eficiencia.	No podría adherirse a normas de eficiencia.	Podría cumplir con normas de eficiencia.
---	-----------------------------	---	---	--

Fuente: propia

A continuación, se presenta la tabla de valoración cualitativa del parámetro eficiencia:

Tabla 32: Valoración cualitativa de Eficiencia

NRO.	INDICADOR	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
1	Comportamiento temporal.	Debilidad menor	Debilidad menor	Fortaleza importante
2	Utilización de recursos.	Debilidad menor	Debilidad menor	Fortaleza importante
3	Cumplimiento de eficiencia.	Debilidad menor	Debilidad menor	Fortaleza importante

Fuente: propia

Para valorar la investigación realizada, se utilizará la matriz aplicada en la evaluación de metodologías por el modelo de requisitos y sus respectivas estimaciones, como se presenta la siguiente tabla:

Tabla 33: Valoración cuantitativa de Eficiencia

FACTORES EVALUACIÓN	DE	METODOLOGÍAS INVESTIGADAS					
		OOHDM		WSDM		UWE	
		Rango	Ponderación	Rango	Ponderación	Rango	Ponderación
COMPORTAMIENTO TEMPORAL.		2	0,4	2	0,4	5	1,0
UTILIZACIÓN DE RECURSOS.		2	0,4	2	0,4	5	1,0
CUMPLIMIENTO DE EFICIENCIA.		2	0,4	2	0,4	4	0,8
TOTAL			1,2		1,2		2,8

Fuente: propia

3.3.11 Interpretación de resultados del parámetro Eficiencia.

Posterior al análisis y evaluación del parámetro eficiencia en las metodologías web se determina los siguientes resultados.

- **Comportamiento temporal.** En comparación de tiempos de creación de software las metodologías OOHDM y WSDM han sido calificadas en menor valor debido a que desarrollo se realiza durante mayor tiempo por razones de la notación propia, falta de una herramienta CASE y el proceso detallado de sus fases, en cambio la elaboración del modelo UWE se ejecuta con mayor rapidez gracias a su herramienta CASE y notación estándar.

- **Utilización de recursos.** Se considera que la utilización de recursos (humanos, software) en las metodologías OOHDM y WSDM es mayor, por lo que la valoración es baja referente a UWE pues al contar con una herramienta de diseño y su notación estándar necesita menos recursos, obteniendo de esta manera una calificación más alta.
- **Cumplimiento de eficiencia.** Como se pudo apreciar las metodologías OOHDM y WSDM son menos adaptables al cumplimiento de eficiencia por lo que son ponderadas en menor valor, mientras que UWE cumple con el parámetro de eficiencia, por lo tanto, es calificada en mayor valor.

3.3.12 Valoración porcentual del parámetro Eficiencia.

Posterior al análisis y evaluación de metodologías web en base al parámetro eficiencia se puede definir la siguiente fórmula que determinará las valoraciones de cada metodología:

$$MODELO = \frac{Total\ valor\ ponderacion * 100}{Nro\ de\ indicadores} \%$$

$$OOHDM = \frac{1,2 * 100}{3}$$

$$OOHDM = 40\%$$

$$WSDM = \frac{1,2 * 100}{3}$$

$$WSDM = 40\%$$

$$UWE = \frac{2,8 * 100}{3}$$

$$UWE = 93,33\%$$

Las valoraciones porcentuales correspondientes a la evaluación del parámetro Eficiencia se resumen en la siguiente tabla:

Tabla 34: Valoración de metodologías

METODOLOGÍA	VALOR PORCENTUAL
OOHDM	40%
WSDM	40%
UWE	93,33%

Fuente: propia

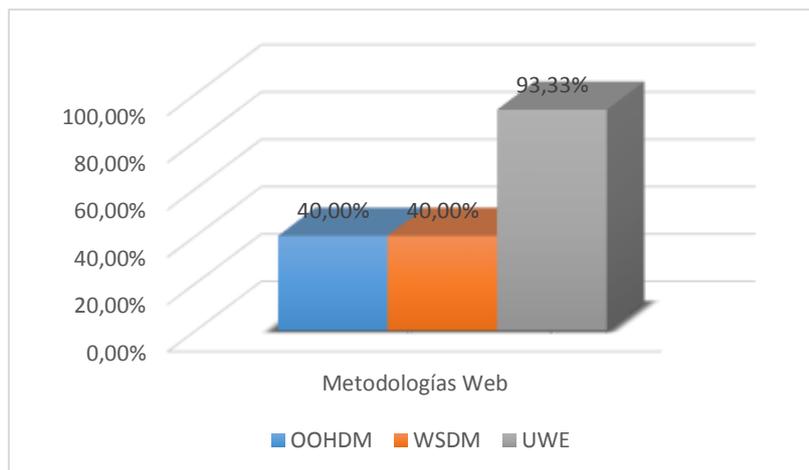


Fig. 37: Valoración porcentual del parámetro eficiencia

Fuente: Propia

3.3.13 Resultados del análisis comparativo del parámetro Eficiencia.

De acuerdo con el parámetro eficiencia se puede determinar que, la metodología OOHDM ha obtenido una valoración del 40%, frente al estándar WSDM que consiguió el porcentaje del 40% y el modelo UWE que alcanzó el valor del 93,33%.

3.3.14 Evaluación del parámetro Usabilidad.

Mediante el parámetro Usabilidad se realizará un análisis de indicadores, los cuales serán valorados cuantitativa y cualitativamente con el fin de seleccionar la metodología más adecuada para el desarrollo de aplicaciones web, la investigación se presenta en la siguiente tabla:

Tabla 35: Matriz de análisis del parámetro Usabilidad

NRO.	INDICADOR	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
1	Capacidad de ser entendido.	Difícil de entender, debido al desconocimiento.	Difícil de entender, debido al desconocimiento.	Más entendible, por su estandarización.
2	Capacidad de ser aprendido.	Complicado mientras no se haya estudiado el modelo.	Complicado mientras no se haya estudiado el modelo.	Menos complicado.
3	Capacidad de ser operado.	Medianamente operable gracias a la descripción de sus fases y sub fases.	No operable.	Operable.
4	Capacidad de atracción.	Medianamente atractivo.	No atractivo	Atractivo.
5	Cumplimiento de usabilidad.	Cumple medianamente la usabilidad.	No cumple con la usabilidad.	Cumple con la usabilidad.

Fuente: propia

A continuación, se presenta la tabla de valoración cualitativa del parámetro Usabilidad:

Tabla 36: Valoración cualitativa de Usabilidad

NRO.	INDICADOR	MODELOS INVESTIGADOS		
		OOHDM	WSDM	UWE
1	Capacidad de ser entendido.	Debilidad menor	Debilidad menor	Fortaleza importante
2	Capacidad de ser aprendido.	Debilidad menor	Debilidad menor	Fortaleza menor
3	Capacidad de ser operado.	Fortaleza menor	Debilidad grave	Fortaleza importante
4	Capacidad de atracción.	Fortaleza menor	Debilidad grave	Fortaleza importante
5	Cumplimiento de usabilidad.	Fortaleza menor	Debilidad grave	Fortaleza importante

Fuente: propia

Para valorar la investigación realizada, se utilizará la matriz aplicada en la evaluación de metodologías por el modelo de requisitos y sus respectivas estimaciones, como se presenta la siguiente tabla:

Tabla 37: Valoración cuantitativa de Usabilidad

FACTORES DE EVALUACIÓN	METODOLOGÍAS INVESTIGADAS					
	OOHDM		WSDM		UWE	
	Rango	Ponderación	Rango	Ponderación	Rango	Ponderación
CAPACIDAD DE SER ENTENDIDO.	2	0,4	2	0,4	5	1,0
CAPACIDAD DE SER APRENDIDO.	2	0,4	2	0,4	4	0,8
CAPACIDAD DE SER OPERADO.	4	0,8	1	0,2	5	1,0
CAPACIDAD DE ATRACCIÓN.	4	0,8	1	0,2	5	1,0
CUMPLIMIENTO DE USABILIDAD.	4	0,8	1	0,2	5	1,0
TOTAL		3,2		1,4		4,8

Fuente: propia

3.3.15 Interpretación de resultados del parámetro Usabilidad.

Posterior al análisis y evaluación del parámetro eficiencia en las metodologías web se determina los siguientes resultados.

- **Capacidad de ser entendido.** Debido al desconocimiento del método los modelos OOHDM y WSDM resultaron menor valorados, a diferencia de UWE que pese a ser

un modelo no utilizado su notación UML en todo su proceso lo hace más fácil entenderse por lo que su valoración es mayor.

- **Capacidad de ser aprendido.** OOHDM y WSDM son estándares que deben ser estudiados para poder aprender y comprender sus procesos, por tal motivo la ponderación que tuvieron es baja, a diferencia del modelo UWE.
- **Capacidad de ser operado.** OOHDM muestra su fortaleza medianamente debido a que puede ser manipulado en breves rasgos por el desarrollador, mientras que, WSDM es un modelo que tiene su dificultad de utilización, a diferencia de UWE que es un modelo que no tiene ninguna dificultad al momento de ser operado, por lo que su valoración resulta mayor a las demás propuestas.
- **Capacidad de atracción.** Los modelos OOHDM y WSDM resultan menos atractivos al usuario por su diagramación propia, pero UWE demostró que gracias a su diagramación estándar y entendible es más atraíble al usuario, por lo que su puntuación es mayor que los estándares analizados.
- **Cumplimiento de usabilidad.** Las propuestas OOHDM y WSDM no cumplen con el cumplimiento de usabilidad por lo que su calificación ha sido menor a la metodología UWE que cumple con el parámetro de usabilidad en el desarrollo de aplicaciones web, por lo que ha sido valorada altamente.

3.3.16 Valoración porcentual del parámetro Usabilidad.

Posterior al análisis y evaluación de metodologías web en base al parámetro Usabilidad se puede definir la siguiente fórmula que determinará las valoraciones de cada metodología:

$$\text{MODELO} = \frac{\text{Total valor ponderacion} * 100}{\text{Nro de indicadores}} \%$$

$$\text{OOHDM} = \frac{3,2 * 100}{5}$$

$$\text{OOHDM} = 64\%$$

$$\text{WSDM} = \frac{1,4 * 100}{5}$$

$$\text{WSDM} = 28\%$$

$$\text{UWE} = \frac{4,8 * 100}{5}$$

$$\text{UWE} = 96\%$$

Las valoraciones porcentuales correspondientes a la evaluación del parámetro Usabilidad se resumen en la siguiente tabla:

Tabla 38: Valoración de metodologías

METODOLOGÍA	VALOR PORCENTUAL
OOHDM	64%
WSDM	28%
UWE	96%

Fuente: propia

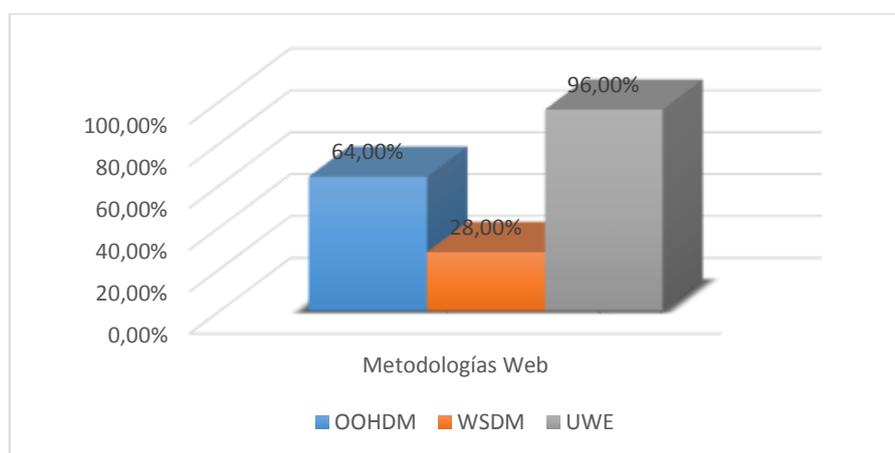


Fig. 38: Valoración porcentual del parámetro Usabilidad

Fuente: Propia

3.3.17 Resultados del análisis comparativo del parámetro Usabilidad.

De acuerdo con el parámetro Usabilidad se puede determinar que, la metodología OOHDM ha obtenido una valoración del 64%, frente al estándar WSDM que consiguió el porcentaje del 28% y el modelo UWE que alcanzó el valor del 96%.

3.4 Interpretación de resultados finales.

De acuerdo con el análisis comparativo mediante el modelo evaluativo de requisitos se pudo verificar en la Tabla 30, que las propuestas OOHDM con 66,15% y WSDM con 60% son menos adaptables al estándar de evaluación, a diferencia de UWE con un valor porcentual del 81,54%, que resultó ser la metodología que cumple con la mayoría de los requerimientos establecidos en el esquema estudiado.

Mediante el parámetro de Eficiencia se pudo visualizar en la Tabla 34, que los modelos OOHDM y WSDM con 40% respectivamente, utilizan mayor tiempo de desarrollo y más recursos sean estos consultores profesionales (recurso humano) y herramientas de apoyo (software) cumpliendo así menormente con la eficiencia por lo que han sido valoradas en baja escala, contrario a la metodología UWE con un valor porcentual de 93,33%, realiza su proceso

en menor tiempo y con su propia herramienta CASE acatando a los indicadores de la eficiencia y siendo calificada en mayor escala.

Mediante el parámetro de Usabilidad se pudo conocer en la Tabla 38, que los estándares con 64% y WSDM con 28% fueron calificados en menor porcentaje debido al desconocimiento de los modelos y su independencia de notación, a su vez UWE, con un valor porcentual de 96%, y gracias a sus técnicas de diagramación basado en UML es un estándar de fácil aprendizaje, comprensión y desarrollo, por lo que ha sido valorada en mayor porcentaje.

En la Tabla 39 se resume los resultados de la investigación.

Tabla 39: Resultados del análisis comparativo de metodologías web

MODELO DE EVALUACIÓN		OOHDM	WSDM	UWE
RAFAEL MENÉNDEZ – BARZANALLA ASENSIO		66,15%	60%	81,54%
ISO/IEC 9126	Eficiencia	40%	40%	93,33%
	Usabilidad	64%	28%	96%

Fuente: propia

De acuerdo con la información detallada en la tabla anterior se realiza una valoración de los resultados obtenidos en la investigación mediante la calificación porcentual presentada en la Tabla 40.

Tabla 40: Valoración de resultados

MODELO DE EVALUACIÓN	VALORACIÓN PORCENTUAL	
RAFAEL MENÉNDEZ – BARZANALLA ASENSIO	50%	
ISO/IEC 9126	Eficiencia	25%
	Usabilidad	25%
TOTAL	100%	

Fuente: propia

Al aplicar la estimación anterior a los resultados de la investigación, se obtiene una valoración general de las metodologías, los valores pueden identificarse en la Tabla 41.

Tabla 41: Resultados generales del análisis de metodologías web

MODELO DE EVALUACIÓN	OOHDM	WSDM	UWE
RAFAEL MENÉNDEZ – BARZANALLA ASENSIO	33,08%	30%	40,77%
ISO/IEC 9126	Eficiencia	10%	23,33%
	Usabilidad	16%	24%
TOTAL	59,08%	47%	80,1%

Fuente: propia

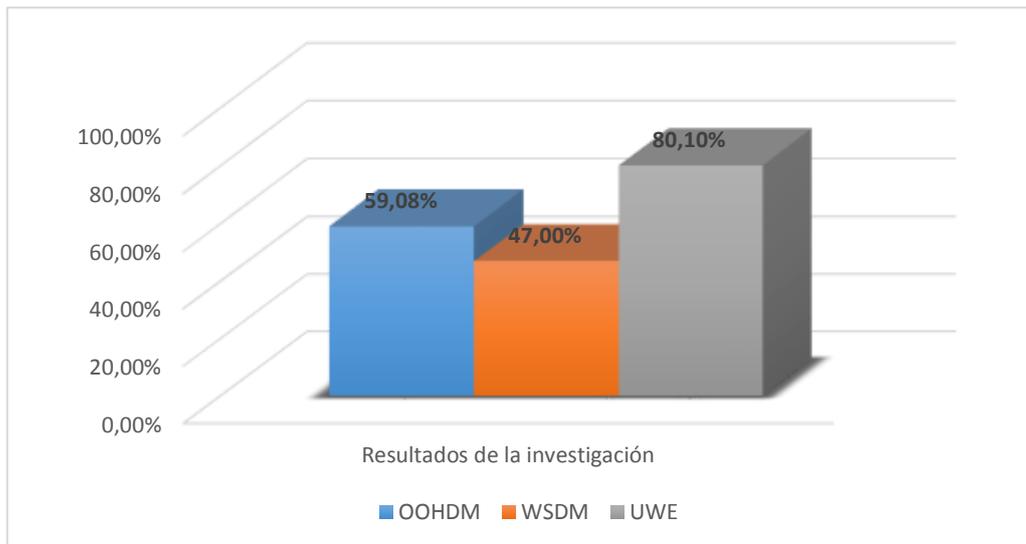


Fig. 39: Resultados del análisis comparativo de metodologías web
Fuente: Propia

3.5 Análisis de resultados finales.

Mediante el modelo de evaluación de metodologías de desarrollo de software de los autores Rafael Menéndez – Barzanallana Asensio y la norma ISO/IEC 9126 se pudo determinar que la metodología ganadora es UWE (UML-Based Web Engineering) pues ha cumplido con los requisitos deseables para la selección de estándares metodológicos y también con los parámetros de usabilidad y eficiencia presentados en la norma mencionada. Por consiguiente, se desarrollará la aplicación odontológica mediante el uso de la metodología web UWE.

CAPÍTULO 4

Desarrollo de la metodología UWE (UML-Based Web Engineering)

4.1 Introducción

Este capítulo hace enfoque en la creación e implementación de la aplicación de control odontológica mediante el uso de la metodología UWE que resultó ser el estándar metodológico mejor adaptable en el desarrollo de aplicaciones web.

El software odontológico fue desarrollado con el objetivo de automatizar los procesos de control, y de esta manera garantizar la calidad y seguridad de la información en la institución dental.

Para el levantamiento de requerimientos se realizó una entrevista al odontólogo beneficiario de la aplicación web, quien expuso las necesidades mediante la siguiente tabla.

Tabla 42: Requerimientos de la Software Odontológico

NOMBRE	DESCRIPCIÓN	PRIORIDAD	NIVEL DE RIESGO
CONTROL DE USUARIOS	Control de acceso Ingresar, Actualizar, Eliminar, Consultar,	ALTA	Crítico
CONTROL DE PACIENTES	Ingresar, Actualizar, Eliminar, Consultar, Control odontológico	ALTA	Crítico
CONTROL DE MÉDICOS (OPCIONAL)	Ingresar, Actualizar, Eliminar, Consultar	BAJA	Baja
CONTROL DE PROCEDIMIENTOS	Ingresar, Actualizar, Eliminar, Consultar,	ALTA	Crítico
REPORTES	Visualizar	BAJA	Baja

Fuente: propia

De acuerdo con los requerimientos se estableció la existencia de un usuario del sistema identificado como **Administrador** quién tendrá acceso total a la aplicación informática, a continuación, se detalla las funciones del usuario:

Control de Usuarios

Tendrá el acceso al sistema mediante la autenticación de la aplicación y podrá realizar las siguientes tareas:

- Control de acceso.
- Ingreso de usuarios.
- Actualización de usuarios.
- Eliminación de usuarios.
- Lectura de usuarios.

Control de Pacientes.

Podrá controlar la información de los pacientes realizando las siguientes tareas:

- Ingreso de pacientes.
 - Control odontológico
- Actualización de pacientes.
- Eliminación de pacientes.
- Lectura de pacientes.

Control de Médicos (Opcional).

Podrá controlar la información de los médicos realizando las siguientes tareas:

- Ingreso de médicos.
- Actualización de médicos.
- Eliminación de médicos.
- Lectura de médicos.

Control de Procedimientos.

Podrá controlar la información de los procedimientos que realiza la entidad, ejecutando las siguientes tareas:

- Ingreso de procedimientos.
- Actualización de procedimientos.
- Eliminación de procedimientos.
- Lectura de procedimientos.

Posterior a la identificación de requisitos se desarrolla las siguientes fases del modelo UWE para la construcción de la aplicación odontológica.

4.2 Modelo de Casos de Uso.

La fase Modelo de casos de uso establece los requerimientos del sistema mediante el modelado de casos de uso de UML, donde el actor principal es el usuario Administrador, que tendrá acceso a la aplicación odontológica y a su vez realizará el control de usuarios, pacientes, médicos, procedimientos y reportes, en la Figura 40 se muestra el diagrama de casos de uso donde se identifica al actor y sus funciones.

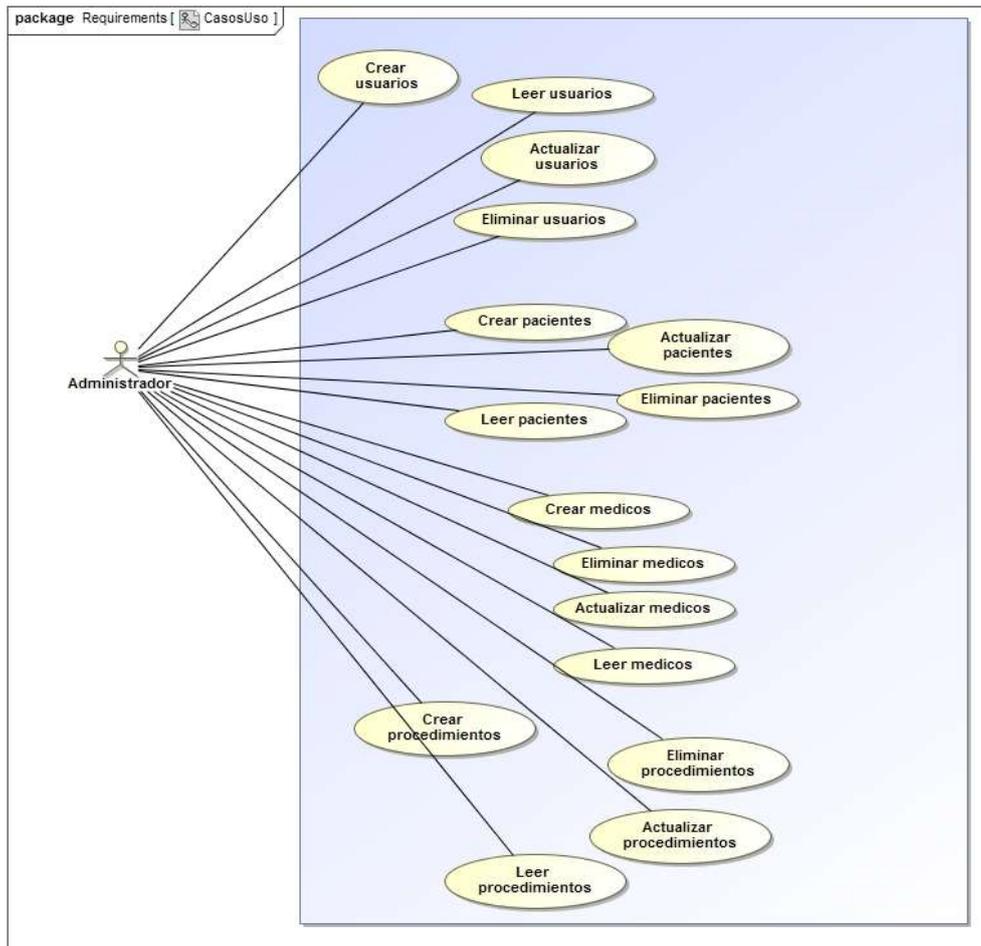


Fig. 40: Diagrama de casos de uso
Fuente: Propia

Usuarios

La Figura 41 presenta el diagrama de casos de uso que identifica las funciones que cumple el Administrador para el acceso y control de usuarios del sistema.

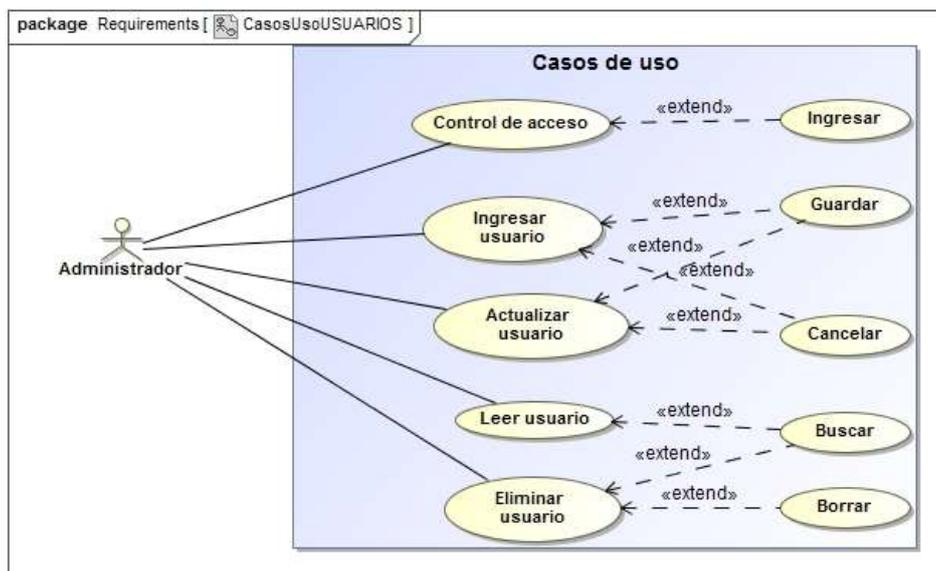


Fig. 41: Casos de uso Usuario
Fuente: Propia

Pacientes

La Figura 42 presenta el diagrama de casos de uso que identifica las funciones que cumple el Administrador, para el control de pacientes.

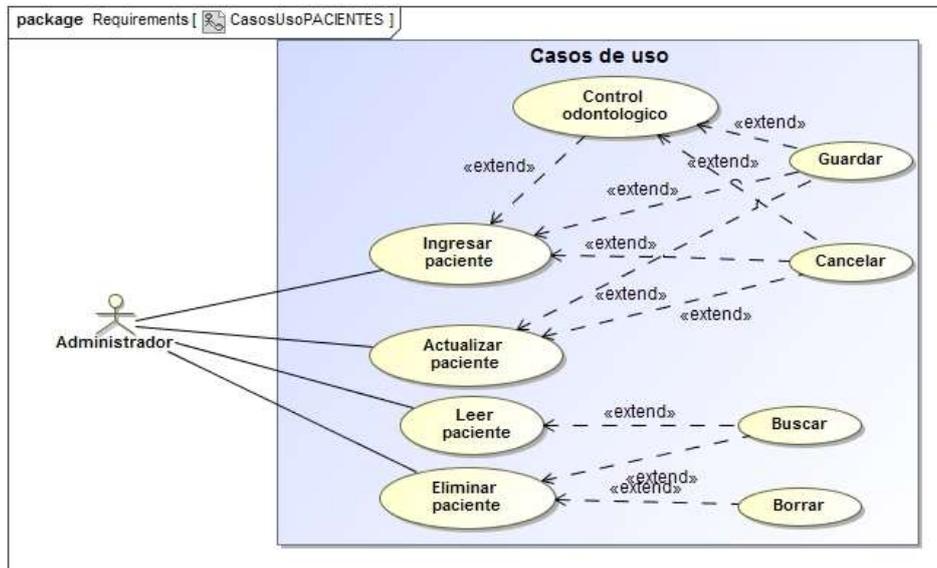


Fig. 42: Casos de uso Paciente
Fuente: Propia

Médicos

La Figura 43 presenta el diagrama de casos de uso que identifica las funciones que cumple el Administrador, para el control de médicos.

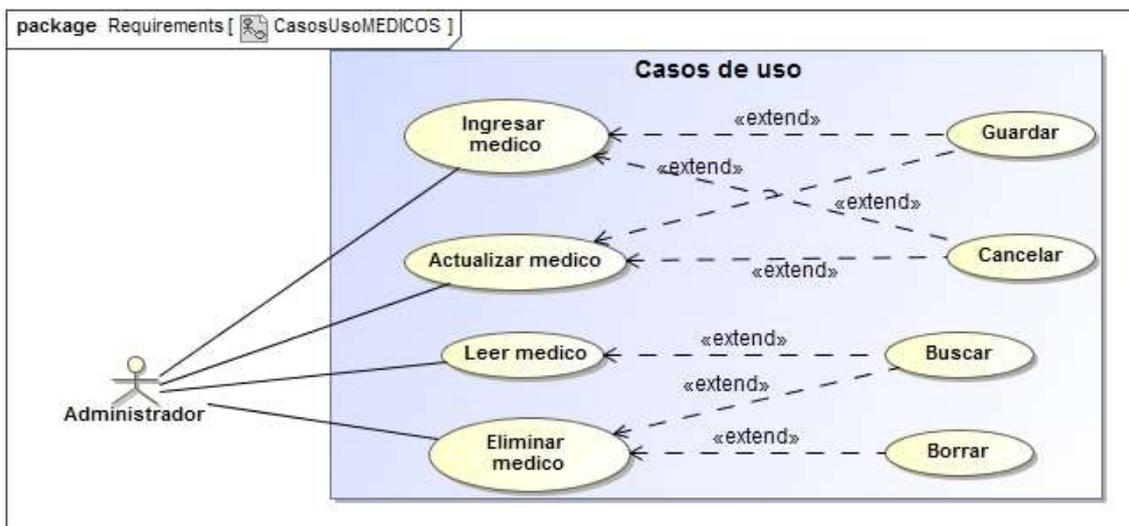


Fig. 43: Casos de uso Medico
Fuente: Propia

Procedimientos

La Figura 44 presenta el diagrama de casos de uso que identifica las funciones que cumple el Administrador, para el control de procedimientos.

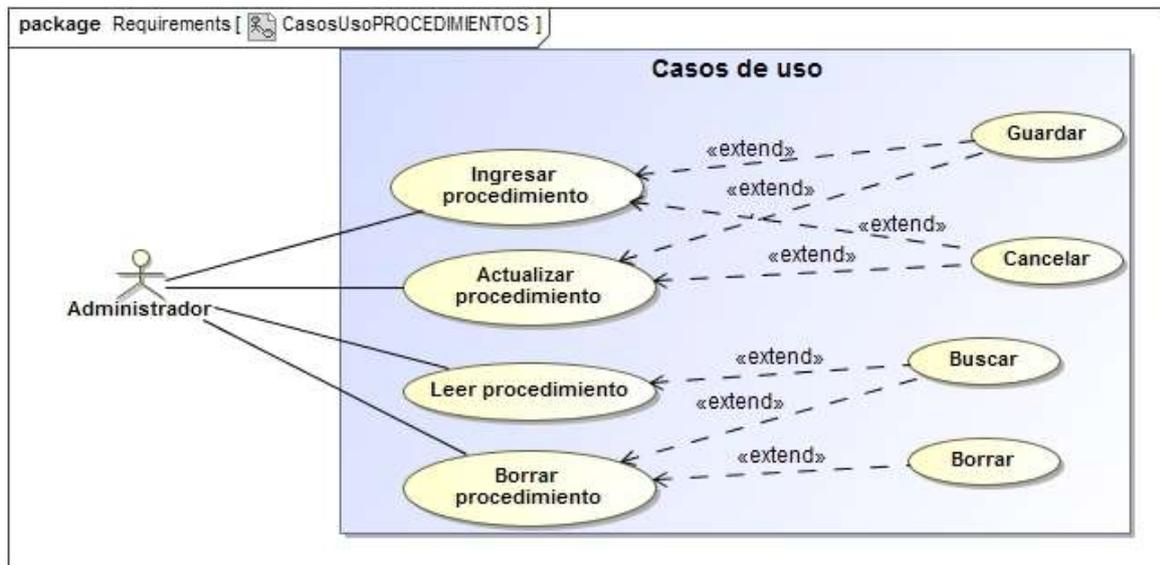


Fig. 44: Casos de uso Procedimientos
Fuente: Propia

4.3 Modelo de Contenido.

Posterior a la identificación de requisitos, se desarrolla la fase del modelo de contenido que se identifica con la elaboración del diagrama de clases de UML sin necesidad de detallar los elementos y notaciones adicionales o específicas como se realizó en la fase anterior, a continuación, se presenta la Figura 45 que detalla el modelo mencionado.

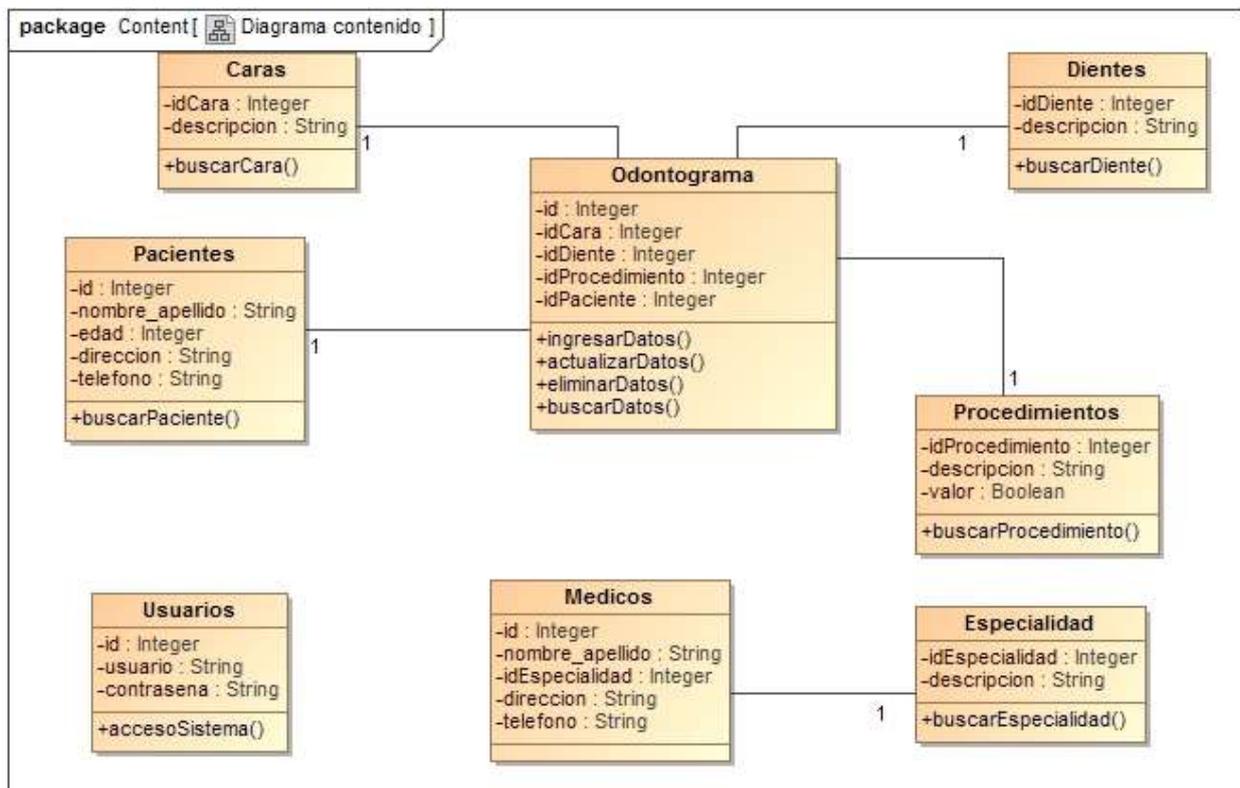


Fig. 45: Diagrama de clases
Fuente: Propia

4.4 Modelo de Navegación.

La fase de navegación utiliza el diagrama de navegación de UML, el cual permite especificar la exploración que el usuario efectúa en la aplicación web, los menús, link de dirección y procesos que realiza el Administrador al moverse en el software desarrollado.

A continuación, se presenta la Figura 46 que identifica el modelo de navegación de la tecnología informática en construcción.

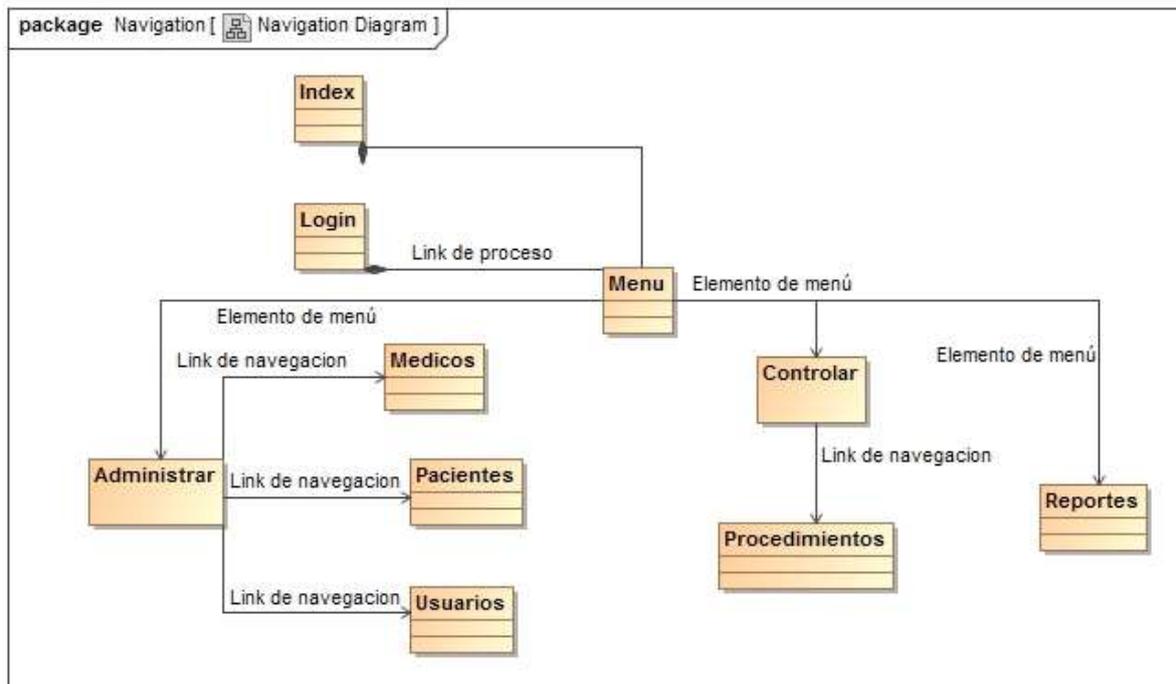


Fig. 46: Diagrama de navegación
Fuente: Propia

4.5 Modelo de Presentación.

La fase de presentación permite presentar una interfaz abstracta de la interfaz de usuario, mediante el diagrama de presentación de UML se podrá diseñar las pantallas principales y los elementos que tendrán las interfaces de la aplicación web.

En la Figura 47 se presenta el diseño de la página principal del software a desarrollarse.

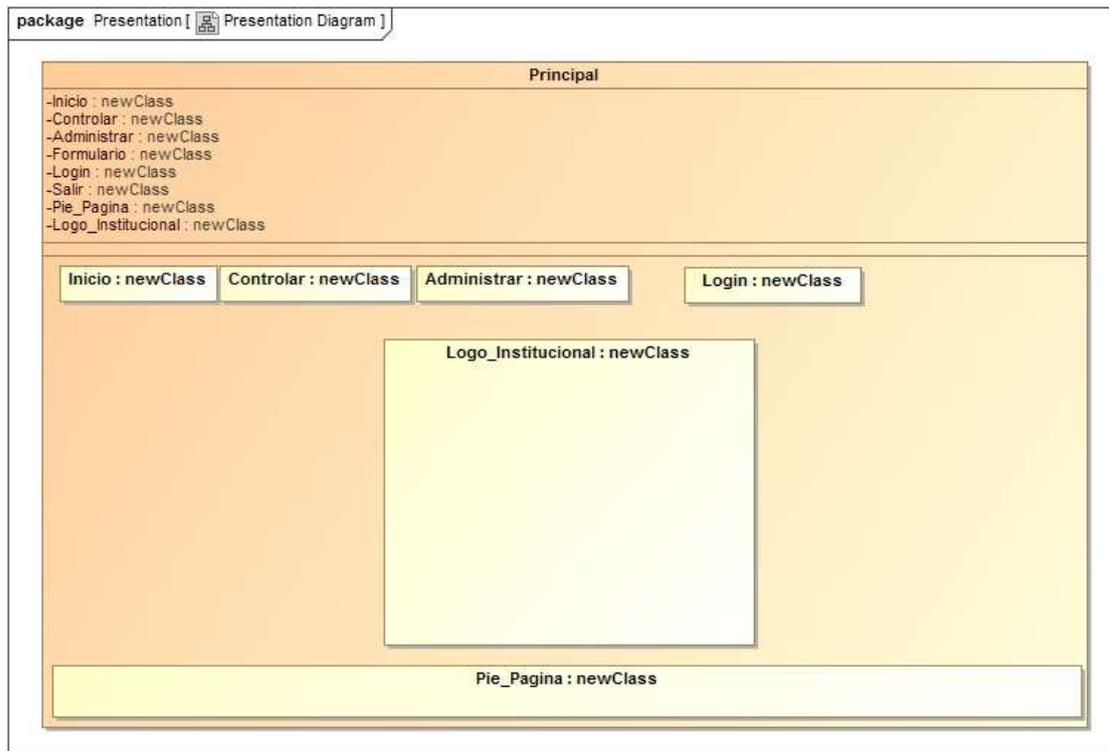


Fig. 47: Diagrama de presentación Principal
Fuente: Propia

La Figura 48 presenta los elementos que contendrá el formulario de acceso a la aplicación informática.

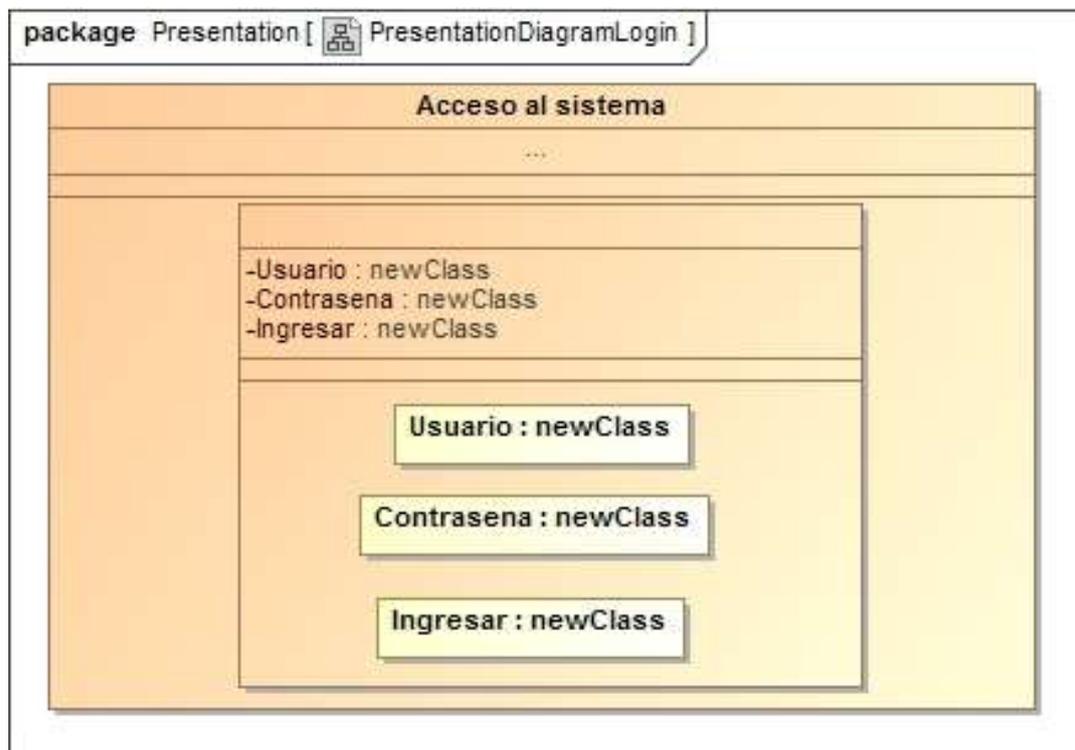


Fig. 48: Diagrama de presentación Login
Fuente: Propia

La Figura 49 presenta el diseño estándar de todas las páginas de la aplicación web, posterior al acceso correcto del sistema.

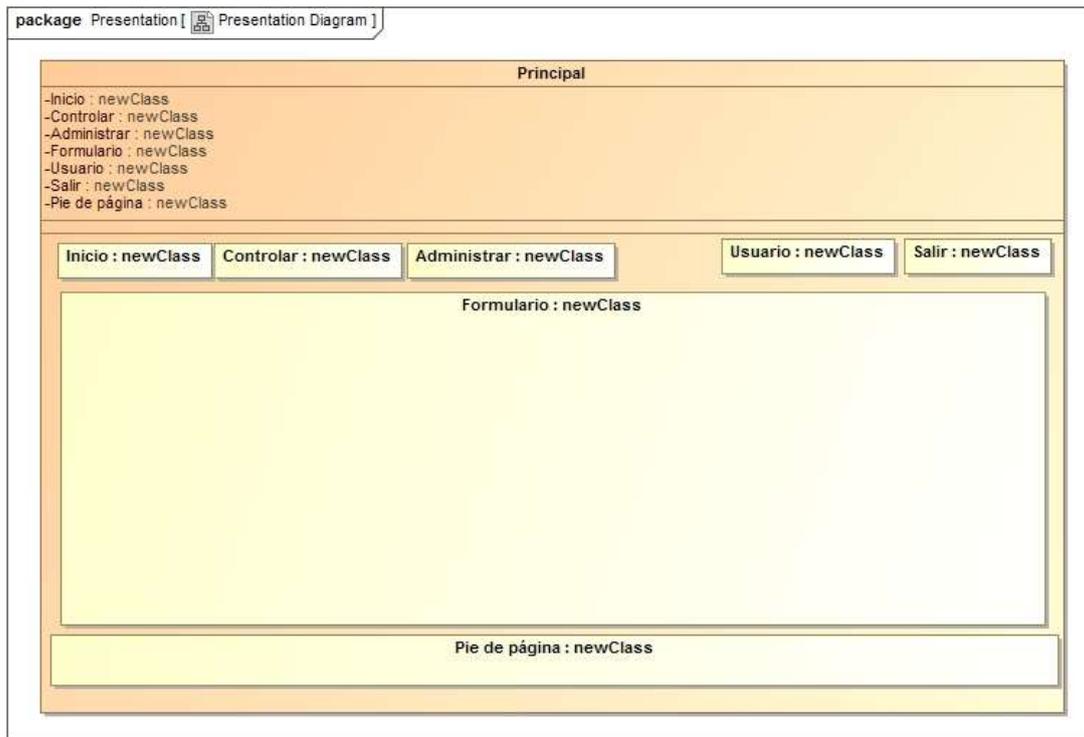


Fig. 49: Diagrama de presentación Estándar
Fuente: Propia

En la Figura 50 presenta la estructura de la página de usuarios mediante el diagrama de presentación de UML.

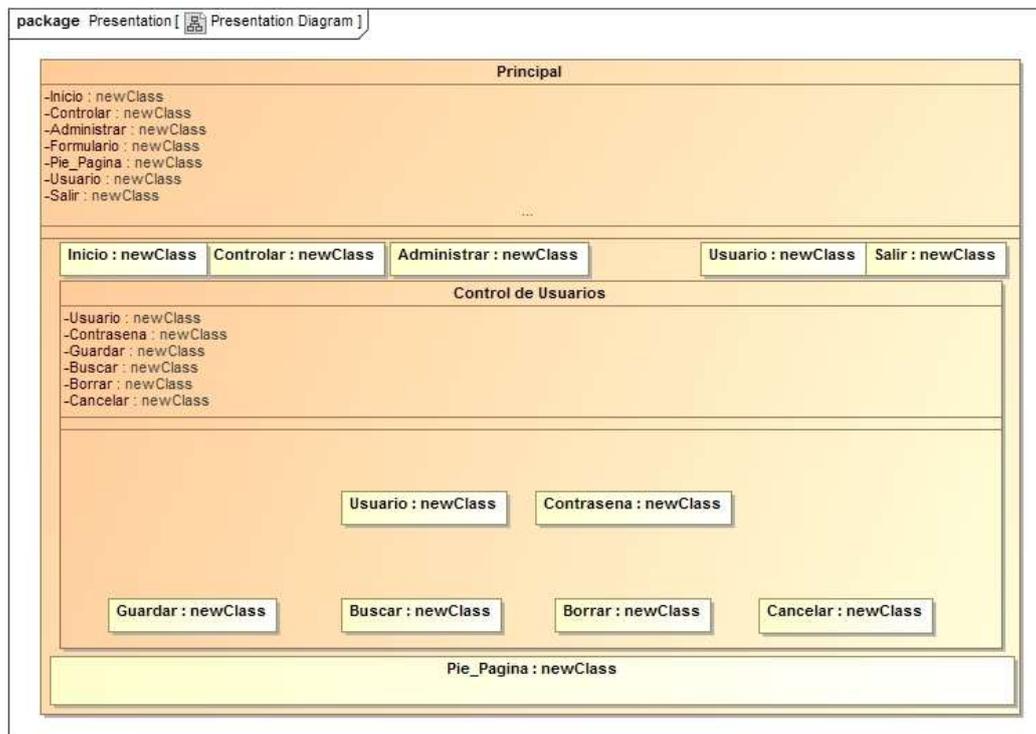


Fig. 50: Diagrama de presentación Usuarios
Fuente: Propia

En la Figura 51 se presenta la estructura de la página de pacientes mediante el diagrama de presentación de UML.

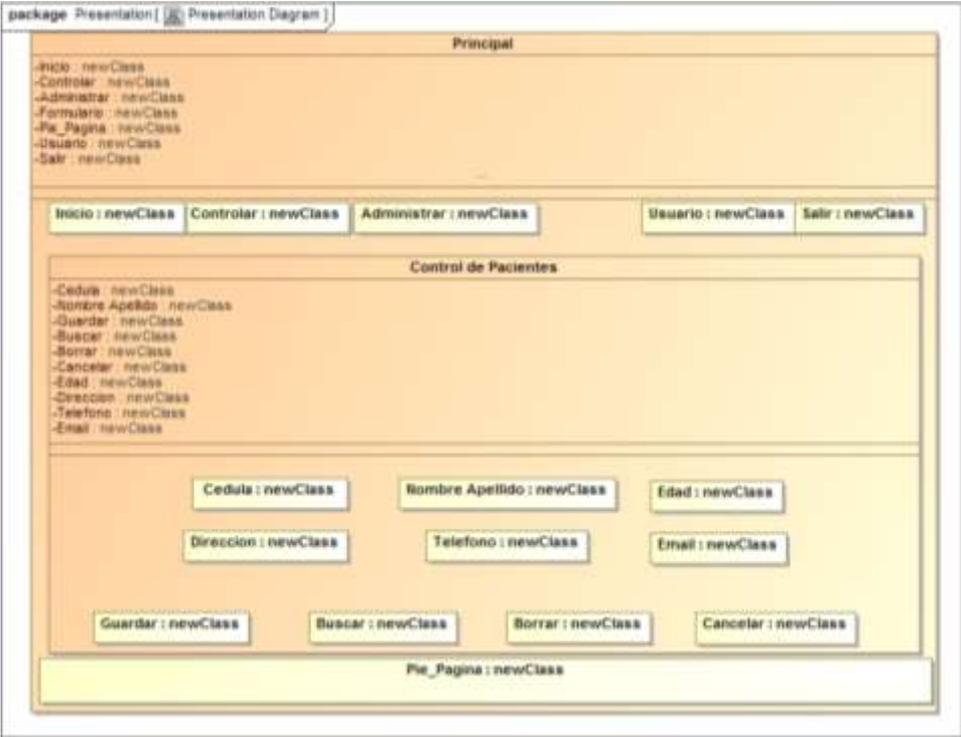


Fig. 51: Diagrama de presentación Pacientes
Fuente: Propia

En la Figura 52 se presenta la estructura de la página de médicos mediante el diagrama de presentación de UML.



Fig. 52: Diagrama de presentación Médicos
Fuente: Propia

En la Figura 53 se presenta la estructura de la página de procedimientos mediante el diagrama de presentación de UML.

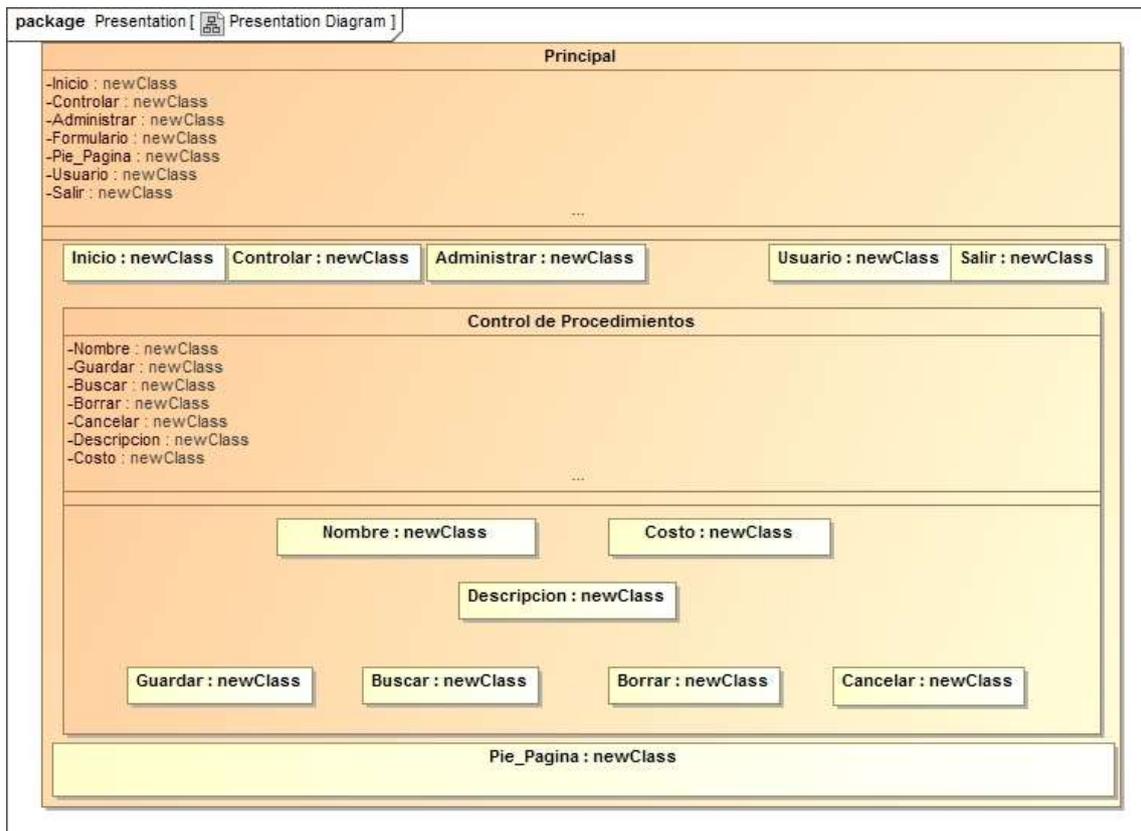


Fig. 53: Diagrama de presentación Procedimientos
Fuente: Propia

4.6 Modelo de Proceso.

En esta fase se detalla a profundidad las acciones y funcionalidades que debe cumplir el software. Para el desarrollo del modelo se utiliza los diagramas UML de estructura y de flujo de proceso.

4.6.1 Modelo de Estructura de proceso.

En esta sub fase se describe las relaciones entre las clases de proceso, en la Figura 54 se describe el modelo estructura de proceso de la tecnología a desarrollarse.

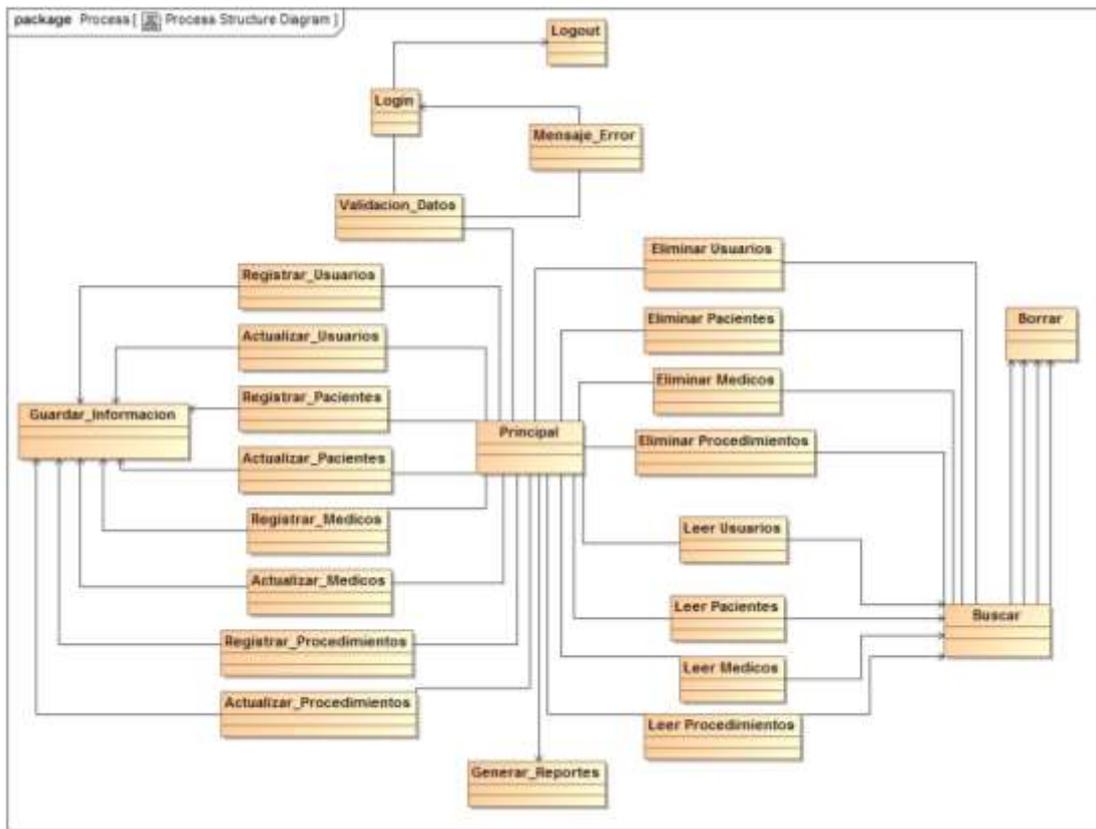


Fig. 54: Diagrama de Estructura de Proceso
Fuente: Propia

4.6.2 Modelo de Flujo de proceso.

En esta sub fase se describe las actividades de la aplicación web, en la Figura 55 se especifica el modelo de flujo de proceso mediante el diagrama de flujo de UML.

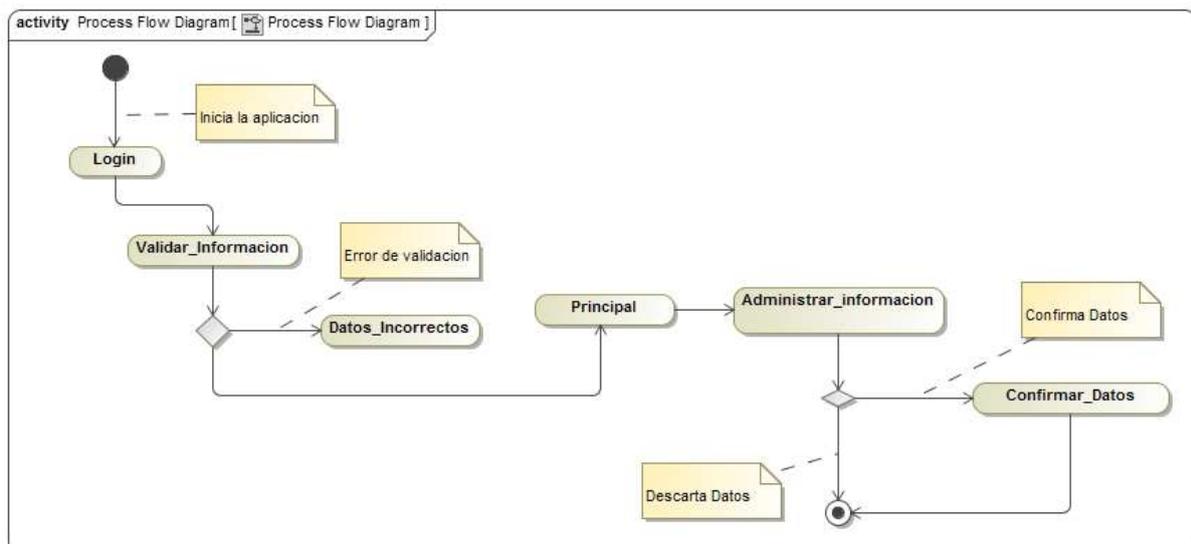


Fig. 55: Diagrama de Flujo de Proceso
Fuente: Propia

CAPÍTULO 5

Construcción de la aplicación web odontológica

5.1 Introducción.

Al finalizar el análisis de las etapas desarrolladas, se considera haber obtenido un completo conocimiento del problema, teniendo una idea clara del diseño, organización de la información y funciones de la aplicación informática.

De acuerdo con esto se puede argumentar que la utilización de la metodología UWE tiene como objetivo simplificar y desarrollar una aplicación eficaz mediante una separación entre las características de diseño y características de navegación, integrando al usuario como eje principal en la validación del sistema, a su vez generando una documentación clara y concisa que permita la corrección y detección oportuna de fallos. De acuerdo con estos parámetros se desarrolla la aplicación web, presentando la arquitectura del sistema, diseño de la base de datos, pruebas, software desarrollado y el ambiente de implementación.

El sistema de control odontológico para el consultorio dental auspiciante ha sido desarrollado mediante el lenguaje Java EE utilizando el framework JSF junto a la librería Primefaces y el gestor de base de datos de código abierto PostgreSQL. En la Tabla 43, se detallan las herramientas utilizadas para su construcción:

Tabla 43: Herramientas para el desarrollo de la aplicación web

HERRAMIENTA	DESCRIPCIÓN
ECLIPSE NEON	Es una plataforma de software compuesta de herramientas que permiten el desarrollo de aplicaciones Java.
JAVA SERVER FACES	Es un Framework para aplicaciones Java basadas en la web, que simplifica el desarrollo de interfaces de usuario en aplicaciones JavaEE.
POSTGRESQL	Es un sistema de gestión de Bases de Datos que permite el alojamiento de la información de las aplicaciones web.
APACHE TOMCAT	Es un contenedor Open Source de tecnologías Java.

Fuente: propia

5.2 Arquitectura de software.

El sistema odontológico ha sido construido mediante el patrón MVC (Model, View, Controller) separando de esta manera las capas de la construcción del software, la arquitectura se detalla en la Figura 56.

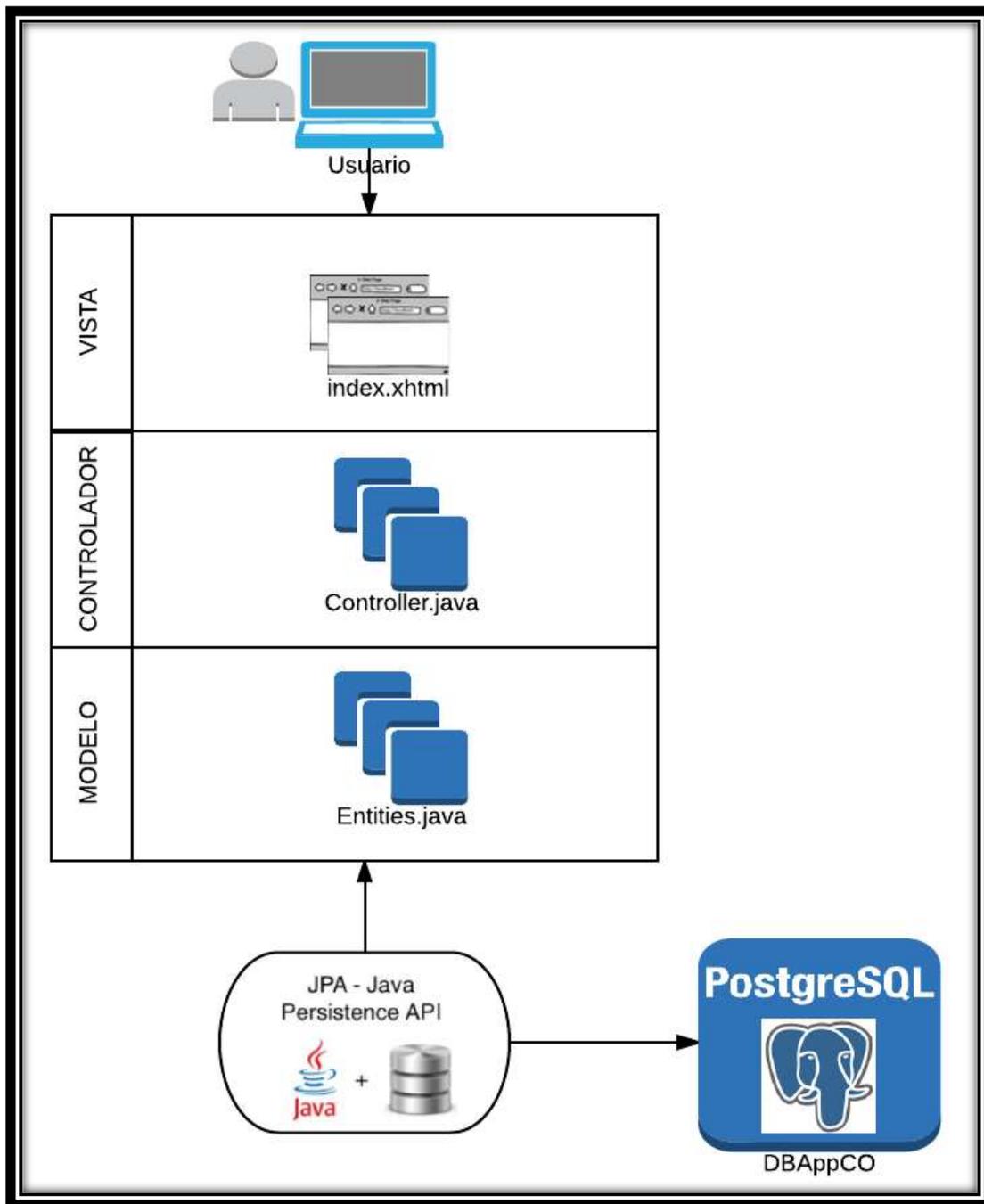


Fig. 56: Arquitectura de software
Fuente: Propia

5.3 Ambiente de implementación.

Para la implementación de la aplicación web odontológica y el soporte de las herramientas necesarias, se utilizó un equipo informático personal, la Tabla 44 presenta las características del PC.

Tabla 44: Ambiente de implementación

HERRAMIENTA	RECURSOS	CARACTERISTICAS
COMPUTADOR PERSONAL	Hardware	- Procesador Intel® Core™ i5-7200U 2.4Hz - Memoria RAM 8GB, DDR4 - Disco duro-SATA de 1TB - Impresora EPSON L210 series
	Software	- Sistema Operativo Windows 10 - Microsoft Office 2016 - Eclipse Neon - DBMS PostgreSQL 9.3 - Navegador web Mozilla Firefox o Google Chrome, etc. - Adobe Reader

Fuente: propia

5.4 Diseño de la base de datos.

En este ítem se presenta el diseño físico de tablas y relaciones que tendrá la base de datos para el almacenamiento de información, se puede argumentar que el desarrollo del modelo se realizó mediante la asistencia grafica del DBMS PostgreSQL 9.3.

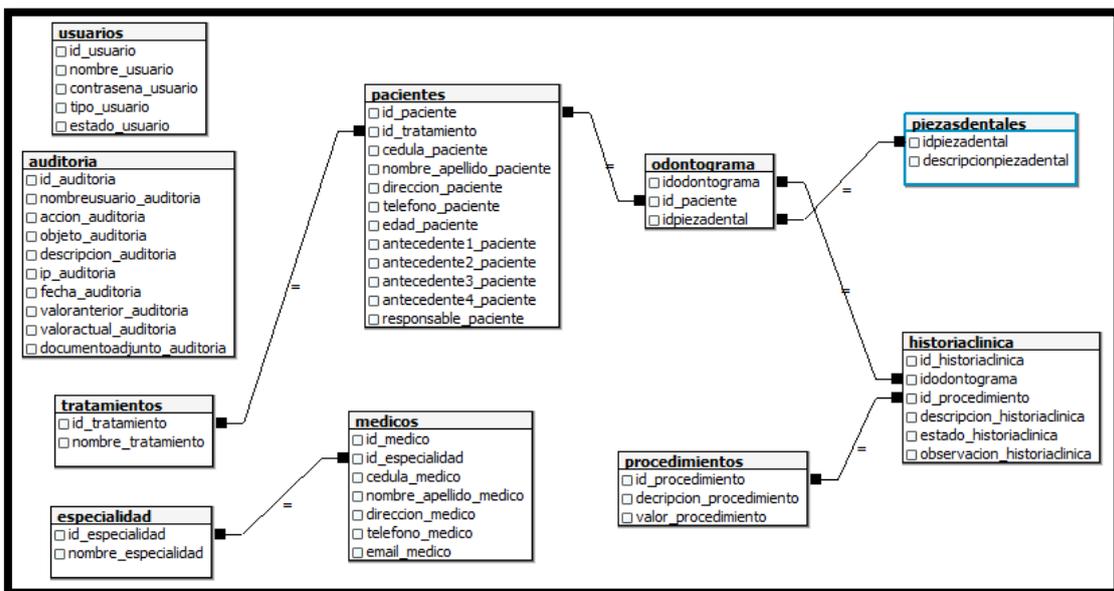


Fig. 57: Diseño físico de la base de datos

Fuente: Propia

5.5 Software Odontológico.

A continuación, se presenta el software de control odontológico. Para el funcionamiento de la aplicación web será necesario el uso de un navegador web el cual permitirá el acceso al sistema mediante la URL donde se ha alojado la aplicación.

La página principal de la aplicación odontológica se puede visualizar en la Figura 58.

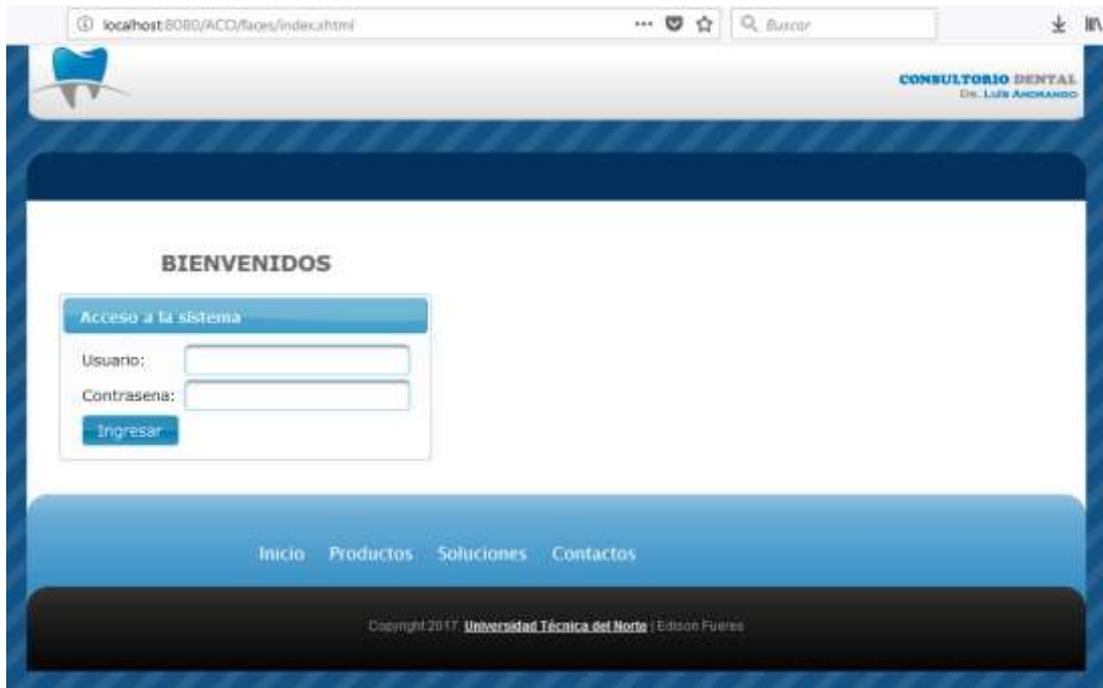


Fig. 58: Pantalla principal de la aplicación odontológica
Fuente: Propia

- **Inicio de sesión.**

Para ingresar a la aplicación el usuario debe encontrarse registrado en la base de datos, pues los datos de usuario y contraseña serán necesarios para el acceso al sistema.

Fig. 59: Pantalla de control de acceso al sistema
Fuente: Propia

Los datos del usuario y contraseña serán los que se encuentren registrados en la aplicación web.

Posterior al ingreso al sistema y de acuerdo con el rol del usuario, se podrá interactuar con los menús de acceso que permitirán la navegación en la aplicación web, en la Figura 60, se presenta los menús de navegación del software:



Fig. 60: Menús de acceso
Fuente: Propia

- **Pacientes**

La página de Pacientes permitirá al Administrador del sistema interactuar con la información del paciente y su historial clínico odontológico.

Fig. 61: Pantalla de pacientes
Fuente: Propia

Control odontológico

Cada paciente tendrá su historial clínico dental, donde el medico podrá controlar la administración de los tratamientos que se aplican a los clientes.



Fig. 62: Pantalla de control odontológico

Fuente: Propia

Los tratamientos que puede ser aplicados en las piezas dentales de cada paciente pueden se definirán de acuerdo con los procedimientos que brinda la entidad odontológica.



Fig. 63: Tratamientos en el control odontológico

Fuente: Propia

- **Especialidades**

La página de Especialidades permitirá al Administrador del sistema controlar las especialidades de los médicos que laboren en la institución odontológica.



Fig. 64: Pantalla de especialidades
Fuente: Propia

- **Médicos**

La página de Médicos permitirá al Administrador del sistema realizar el control de los médicos que son parte de la entidad privada.

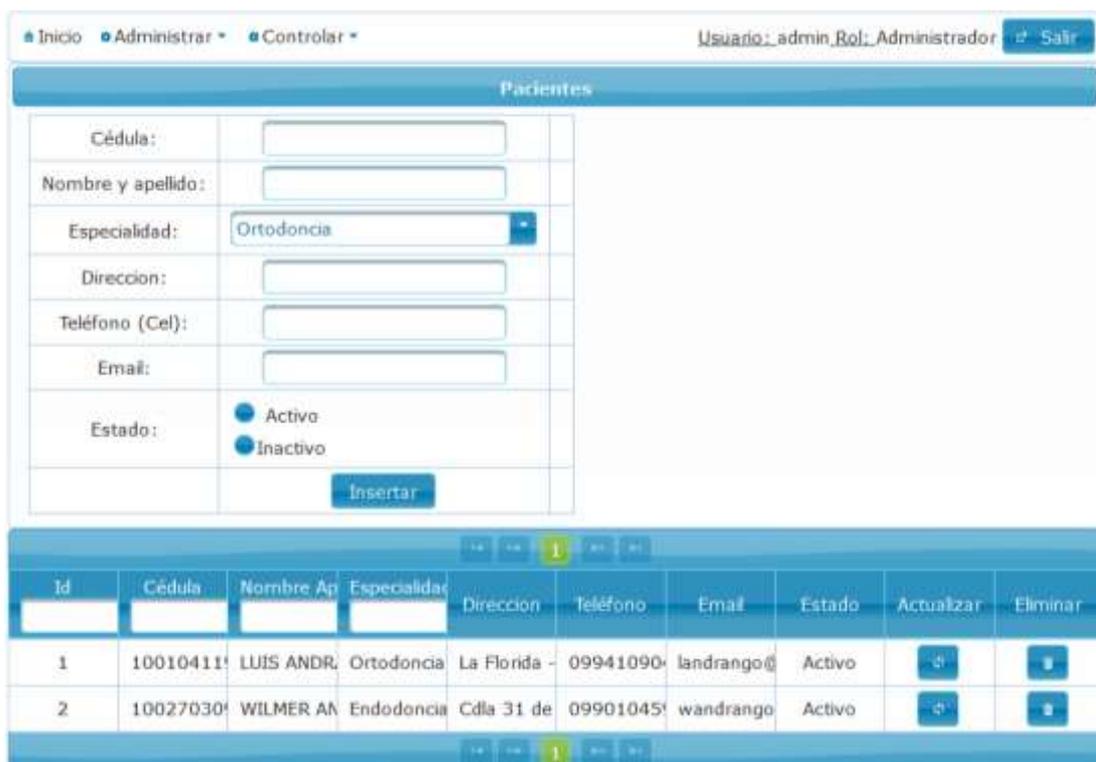


Fig. 65: Pantalla de médicos
Fuente: Propia

- **Usuarios**

La página de Usuarios permitirá al Administrador del sistema, asignar usuarios que controlen la aplicación de acuerdo con su determinada función.



Fig. 66: Pantalla de usuarios

Fuente: Propia

- **Procedimientos**

La página de Procedimientos permite al Administrador del sistema realizar el control de costos de los procedimientos disponibles en la entidad médica.



Fig. 67: Pantalla de usuarios

Fuente: Propia

- **Reportes**

La página de Reportes permite generar la documentación requerida por el usuario de la aplicación odontológica.

PACIENTES

Cedula	Nombre	Edad	Direccion	Telefono
1003149737	EDISON ANDRES FUERES QUILUMBANGO	28	Arahualpa Valle del Amanecer	0988212966
1002004461	LUIS ANGEL FUERES ANRANGO	47	Arizaga Toral	0997397667
1003223722	PATRICIA LORENA CHANDI ANDRANGO	28	La Victoria	0992342078
	SOFIA ELIZABETH AVILA MUNOZ	13	Cdla Ruminahui	0998735906
	JEFFERSON ALEXANDER FUERES QUILUMBANGO	19	Arahualpa Valle del Amanecer	0998999898
1000735827	MANUEL GERSON ANDRANGO	20	La Joya	0999999999

Fig. 68: Reporte de pacientes

Fuente: Propia

CAPÍTULO 6

Resultados

Al concluir el desarrollo de la aplicación de control odontológica mediante el uso de la metodología web UWE (UML-Based Web Engineering) y de acuerdo con la escala de ponderación de Likert se calificará los aspectos técnicos y funcionales detallados en el control de calidad.

6.3 Control de calidad

Para el control de calidad se validó los siguientes aspectos técnicos y funcionales en función del uso de la metodología UWE en el desarrollo de la aplicación odontológica.

6.3.1 Aspecto Funcional

Eficacia. Con relación al cumplimiento de los objetivos establecidos en la definición de requerimientos de la metodología UWE, para el correcto desarrollo de la aplicación web.

Facilidad. Con relación a la facilidad de comprensión de la metodología UWE en el desarrollo del software y sencilla utilización de la aplicación web.

La Tabla 45 detalla la ponderación que obtuvieron los aspectos funcionales evaluados en la aplicación web.

Tabla 45: Resultados de los aspectos funcionales

Aspectos Funcionales				
Aspecto	Excelente	Bueno	Regular	Malo
Eficacia	5			
Facilidad	5			
Total	10			

Fuente: Propia

Aspectos Funcionales= Excelente

6.3.2 Aspecto Técnico

Simplicidad. De acuerdo con la aplicación de la metodología UWE el software odontológico atiende a principios estéticos, es decir, amigable ya que facilita su uso, evitando que el administrador se desoriente de su ubicación en el sistema.

Navegación. Permite una navegación y utilización sencilla por los usuarios de la aplicación, gracias a la descripción detallada de procesos, navegación y funciones del sistema.

La Tabla 46 detalla la ponderación que obtuvieron los aspectos técnicos evaluados en la aplicación web.

Tabla 46: Resultados de los aspectos técnicos

Aspectos Técnicos				
Aspecto	Excelente	Bueno	Regular	Malo
Simplicidad	5			
Navegación	5			
Total	10			

Fuente: Propia

Aspectos Técnicos= Excelente

6.4 Análisis de impactos

El análisis de impactos se sustenta con información estadística, cambios y mejoras de procesos, diseño, tiempos, rendimiento, etc. Además, permite conocer el beneficio que obtuvo la aplicación odontológica al utilizar la metodología UWE.

En la Tabla 47 se detalla los resultados del análisis de impactos que obtuvo la utilización de la metodología UWE en la construcción de la aplicación web.

Tabla 47: Resultados del análisis de impactos

Análisis (cambios y mejoras)	Evaluación Anterior	Evaluación actual	Explicación (Evaluación actual)
Procesos	6	10	Se mejoró la comprensión de los procesos detallados en la metodología UWE, ayudando positivamente la construcción de la aplicación web.
Diseño	7	10	Al aplicar la metodología UWE se pudo desarrollar una aplicación que contenga una presentación atractiva al usuario final.
Rendimiento	7	10	Al usar la metodología UWE el sistema funcionará eficazmente.
Tiempos	5	10	La construcción de la aplicación web se desarrollará en menor tiempo gracias al uso de la metodología UWE.

Fuente: Propia

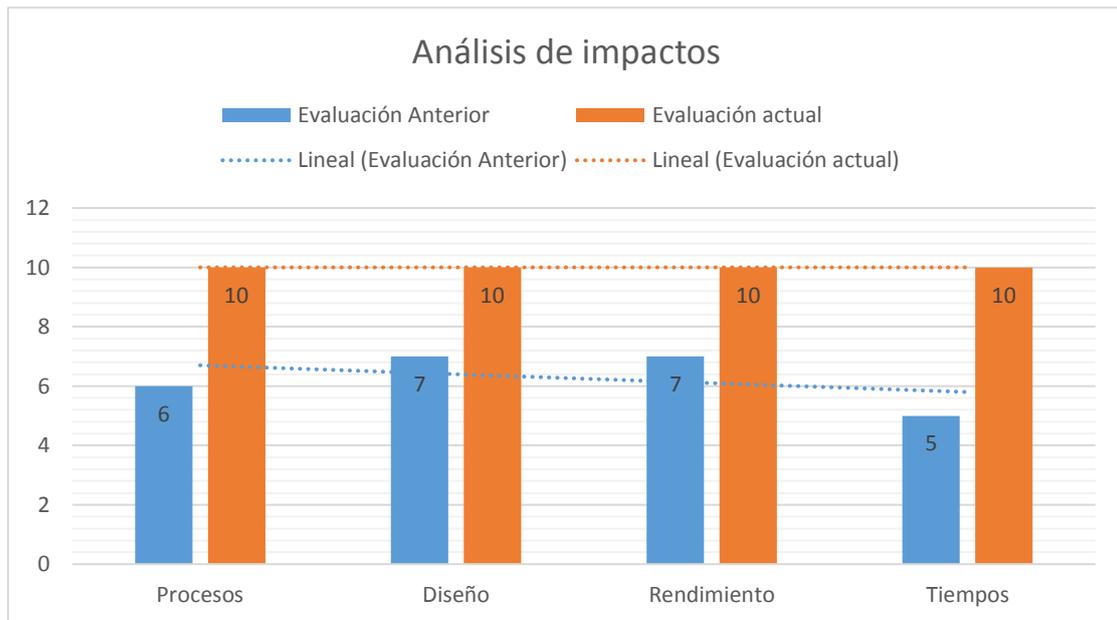


Fig. 69: Representación gráfica del análisis de impactos
Fuente: Propia

Al utilizar la metodología web UWE se mejoró el desarrollo de la aplicación odontológica, la información se confirma en la Figura 70.

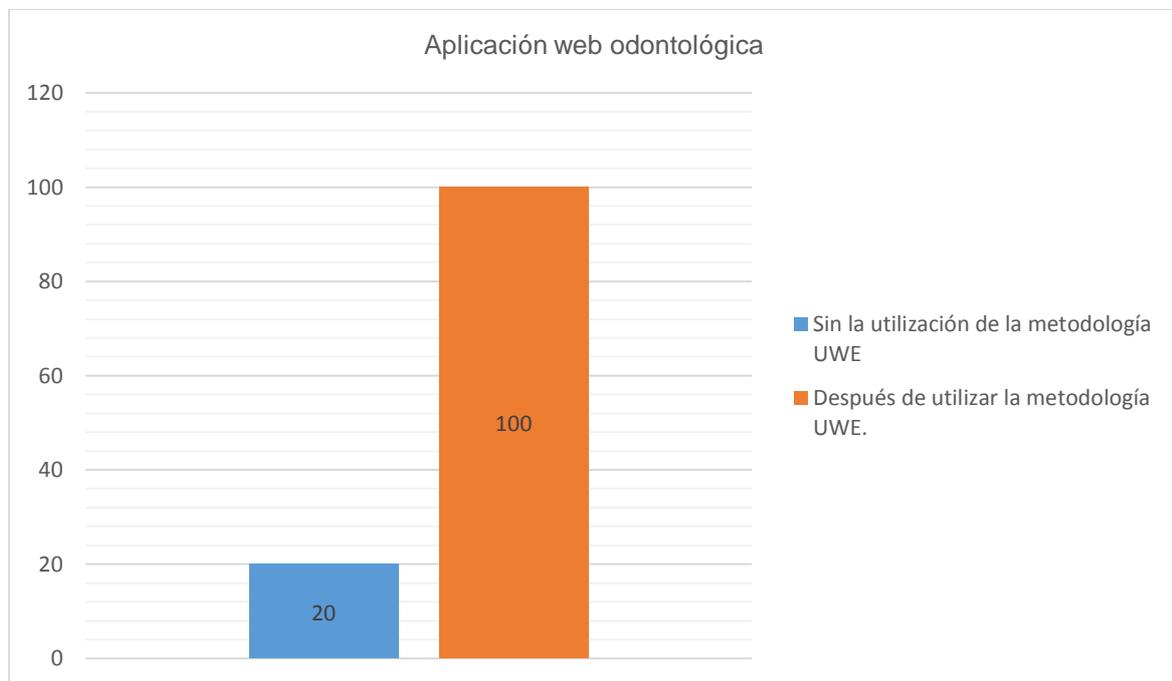


Fig. 70: Representación gráfica del uso de la metodología web
Fuente: Propia

Posterior a la utilización de UWE (UML-Based Web Engineering) se obtuvo excelentes resultados, debido a que la metodología aplicada ayudó en la construcción de una aplicación

web de alta calidad, pues considera el análisis detallado de los requerimientos, claridad en la representación del diseño conceptual y navegacional, y particularmente desarrolla una interfaz gráfica muy atractiva y aceptada por el usuario final.

De acuerdo con los resultados obtenidos se considera imprescindible la utilización de la metodología UWE (UML-Based Web Engineering) en el desarrollo de aplicaciones basadas en la web.

CONCLUSIONES Y RECOMENDACIONES

Conclusiones.

Se puede argumentar lo siguiente:

- Se investigó las principales metodologías, lo cual permitió conocer que OOHDM, WSDM y UWE son los modelos más preponderantes para el desarrollo de aplicaciones web.
- Se realizó el análisis comparativo de las metodologías OOHDM, WSDM y UWE, mediante el desarrollo de prototipos, modelo de evaluación y varios criterios de la norma de calidad de software ISO/IEC 9126.
- Mediante los resultados obtenidos en el estudio comparativo se llegó a determinar que, la metodología UWE es el modelo mejor adaptable en el desarrollo de aplicaciones web.
- Se creó el software de control odontológico mediante el uso de la metodología UWE, esto permitió el desarrollo de una aplicación web buena y funcional.
- Se ha llegado a concluir que no existe una metodología perfecta para el desarrollo de software.

Recomendaciones.

Se recomienda:

- Realizar un análisis previo de los modelos web, para la correcta selección de metodologías de desarrollo del software.
- Investigar modelos de evaluación de metodologías y normas de calidad para la construcción de aplicaciones informáticas.
- Utilizar la metodología UWE por su adaptabilidad a la construcción de aplicaciones web.
- Realizar una integración de los procesos analizados en los modelos OOHDM y UWE, para la creación de una metodología web híbrida más robusta.
- Hacer uso de software libre para la construcción de aplicaciones web, ya que esto reducirá costos a la entidad beneficiaria.

GLOSARIO DE TÉRMINOS

E-R: Entity relationship o Entidad Relación.

HDM: Hypertext Design Model o Modelo de Diseño Hipermedia.

HTML: HyperText Markup Language o Lenguaje de Marcas de Hipertexto.

ISO: International Organization for Standardization u Organizacion Internacional de Normalizacion.

JAVA EE: Java Platform Enterprise Edition.

JSF: JavaServer Faces.

MDD: Desarrollo de Software Dirigido por Modelos, paradigma basado en modelos simplificados del sistema.

OMG: Object Management Group.

OOHDM: Object Oriented Hypermedia Design Methodology o Método de Diseño de Hipermedia Orientado a Objetos.

PAAS: Platform as a Service o Plataforma como Servicio.

RMM: Relationship Management Methodology o Metodología de Administración de relaciones.

SQL: Structured Query Language o Lenguaje de consulta estructurada.

UID: Identificador de Usuario.

UML: Unified Modeling Language o Lenguaje Unificado de Modelado.

UTN: Universidad Técnica del Norte.

UWE: UML-Based Web Engineering o Ingeniería Web basado en UML.

WEBML: Web Modeling Language.

WSDM: Web Site Design Method o Método de Diseño para Sitios Web.

WWW: World Wide Web.

REFERENCIAS BIBLIOGRÁFICAS

- Calva, M., & Romero, K. (2010). *Definición de un ciclo de producción para web y multimedia para el grupo "Sociedad Electronica"*. Universidad Tecnica Particular de Loja, Loja. Obtenido de <http://dspace.utpl.edu.ec/bitstream/123456789/14624/1/1015020.pdf>
- Cardenas, J. (2011). *La utilizacion de la Ingeniería de Software en Hipermedia*. Obtenido de <http://ojs.unemi.edu.ec/index.php/cienciaunemi/article/view/31/28>
- Collaguazo, R., & Pulloquina, Y. (2014). *Implementación de un sistema web de gestión para la clinica veterinaria Reprocerdo aplicando la metodología OOHDM (Metodo de Diseño de Hipermedia Orientado a Objetos) utilizando herramientas software libre*. Latacunga. Obtenido de <http://repositorio.utc.edu.ec/handle/27000/1900>
- Computer Hope. (10 de 02 de 2017). *Computer Hope*. Obtenido de Computer Hope Web Site: <https://www.computerhope.com/>
- Coro, J. (2014). *Estudio comparativo de las metodologías UWE y OOWS para mejorar la productividad en el desarrollo de aplicaciones web. Caso práctico: Centro Parvulario Politécnico*. Escuela Superior Politécnica del Chimborazo, Riobamba. Obtenido de <http://dspace.espech.edu.ec/handle/123456789/3612>
- Garcés, R., & Hinojosa, C. (2011). Sistema Cero Papeles para la Revision de Correspondencia del Banco COFIEC S.A. *Revista DECC Report, Tendencias en Computación*, 42-52. Obtenido de <http://journal.espe.edu.ec/index.php/geeks/article/view/258/235>
- García, G., Gutiérrez, F., Camacho, E., Lorenzo, J., & Escalamilla, J. (2015). Implementación de un sistema MOOC para el aprendizaje de la Geometría. *Pistas Educativas Año XXXVI*, 443-455.
- IFML. (2017). Obtenido de <http://www.ifml.org/>
- Llauradó, O. (12 de Diciembre de 2014). *Netquest*. Obtenido de <https://www.netquest.com/blog/es/la-escala-de-likert-que-es-y-como-utilizarla>
- Méndez, E. (2010). *Modelo de evaluación de metodologías para el desarrollo de software*. Tesis especialidad, Universidad Católica Andrés Bello, Caracas. Obtenido de <http://biblioteca2.ucab.edu.ve/anexos/biblioteca/marc/texto/AAQ7365.pdf>

- Moreno, J. C., & Marciszack. (2014). *La Usabilidad web en el Desarrollo de Software dirigido por modelos*. Cordoba. Obtenido de <http://hdl.handle.net/10915/41611>
- Oliveros, A., Danyans, F., & Mastropietro, M. (2014). *Práctica de Ingeniería de Requerimientos en el desarrollo de aplicaciones Web*. Buenos Aires. Obtenido de http://wer.inf.puc-rio.br/WERpapers/artigos/artigos_WER14/paper9.pdf
- Oracle. (2017). Obtenido de <http://www.oracle.com/technetwork/java/javaee/overview/index.html>
- PostgreSQL. (2017). Obtenido de <https://www.postgresql.org/>
- Primefaces. (2017). Obtenido de <https://www.primefaces.org/>
- Red Hat. (2017). *Openshift*. Obtenido de <https://www.openshift.com/>
- Rodriguez, A. (2010). *Metodologías de diseño usadas en Ingeniería Web, su vinculación con las NTICS*. Universidad Nacional de La Plata, Buenos Aires. Obtenido de <http://hdl.handle.net/10915/4172>
- Rojo, S. d. (2013). *Elicitación y Especificación de requerimientos no funcionales en aplicaciones web*. Tesis maestría, Universidad Nacional de la Plata, Buenos Aires. Obtenido de <http://hdl.handle.net/10915/33039>
- Secretaria Nacional de la Administracion Publica. (2017). *Plan de Gobierno Electrónico*. Quito. Obtenido de <http://www.gobiernoelectronico.gob.ec/wp-content/uploads/downloads/2016/11/Plan-Gobierno-Electro%CC%81nico-Final.pdf>
- SENA. (2015). *Modelos de calidad en el desarrollo de software*. Bogotá. Obtenido de https://senaintro.blackboard.com/bbcswebdav/institution/semillas/228106_2_VIRTUAL-2015/contenido/oaaps/oaap10/aa2/oa_calidad/oa.pdf
- Silva, L. (2012). *Sistema web de Solicitud de Servicios para la empresa Core Business Consulting*. Universidad Nacional Abierta, Caracas. Obtenido de <http://biblo.una.edu.ve/docu.7/bases/marc/texto/t38841.pdf>
- Silvera, J., Arias, D., Gil, G., Díaz, M., Villanueva, G., González, V., . . . Chauque, E. (2015). *Metodologías WebML aplicada a un Sistema de Gestión de Calidad en Centros de Investigación*. Buenos Aires. Obtenido de <http://hdl.handle.net/10915/45929>
- UWE. (2017). *UML-Based Web Engineering*. Obtenido de <http://uwe.pst.ifi.lmu.de/index.html>

- Velarde, G. D., & Pilco, M. (2014). *Análisis comparativo de metodologías para el desarrollo de la aplicación web del control de las practicas pre-profesionales de la EIS de la ESPOCH*. Tesis pregrado, Riobamba. Obtenido de <http://dspace.esPOCH.edu.ec/handle/123456789/3576>
- Vera, M. (2015). *Desarrollo dirigido por modelos basado en componentes de interfaz de usuario*. Universidad Nacional de la Plata, Buenos Aires. Obtenido de <http://hdl.handle.net/10915/48475>
- Vera, P., Pons, C., González, C., Guilianelli, D., & Rodríguez, R. (2013). Metodología de Modelado de Aplicaciones Web Móviles Basada en Componentes. *XV Workshop de Investigadores en Ciencias de la Computación*, 451-455.
- Vilariño, J. (2010). *Modelo para la seleccion de la metodologia de desarrollo web de una aplicacion segun sus características funcionales*. Universidad Catolica Andrés Bello. Obtenido de <http://biblioteca2.ucab.edu.ve/anexos/biblioteca/marc/texto/AAS2255.pdf>
- Villarroel, R., & Rioseco, C. (2011). Una comparación de metodologías para el modelado de aplicaciones web. *Revista Cubana de Ciencias Informáticas*. Obtenido de <http://www.redalyc.org/articulo.oa?id=378343672004>