



UNIVERSIDAD TÉCNICA DEL NORTE

FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CARRERA DE INGENIERÍA EN SISTEMAS COMPUTACIONALES

**TRABAJO DE GRADO PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO EN SISTEMAS COMPUTACIONALES**

TEMA:

**“ANÁLISIS COMPARATIVO ENTRE LA BASE DE DATOS NO
RELACIONAL MONGODB CON LA BASE DE DATOS POSTGRESQL,
SISTEMA PARA LA GESTIÓN DE CLIENTES Y REGISTRO DE PAGOS
DE LA CLÍNICA ODONTOLÓGICA ORTHO DENT.”**

AUTOR: LUIS GERMAN CORREA REAL

DIRECTOR: ING. MAURICIO REA

IBARRA – ECUADOR

2015



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 información:

DATOS DE CONTACTO			
CÉDULA DE IDENTIDAD:	100264665-9		
APELLIDOS Y NOMBRES:	CORREA REAL LUIS GERMAN		
DIRECCIÓN:	IBARRA, AVDA. CRISTÓBAL DE TROYA 2-41.		
EMAIL:	luiscorrea1988@hotmail.com		
TELÉFONO FIJO:	06 2547 123	TELÉFONO MOVIL:	0939407204
DATOS DE LA OBRA			
TÍTULO:	"ANÁLISIS COMPARATIVO ENTRE LA BASE DE DATOS NO RELACIONAL MONGODB Y LA BASE DE DATOS RELACIONAL POSTGRESQL. SISTEMA PARA LA GESTIÓN DE CLIENTES Y REGISTRO DE PAGOS DE LA CLÍNICA ODONTOLÓGICA ORTHO DENT."		
AUTOR:	CORREA REAL LUIS GERMAN		
FECHA:	ABRIL DEL 2015		
PROGRAMA:	<input type="checkbox"/> PREGRADO <input type="checkbox"/> POSTGRADO		
TÍTULO POR EL QUE OPTA:	INGENIERO EN SISTEMAS COMPUTACIONALES		
DIRECTOR:	ING. MAURICIO REA		

2 AUTORIZACIÓN DE USO A FAVOR DE LA UNIVERSIDAD

Yo, Luis Germán Correa Real, portador de la cédula de ciudadanía N° 100264665-9, en calidad de autor y titular de los derechos patrimoniales de la obra o Trabajo 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.



Firma

Nombre: Luis Germán Correa Real

Cédula: 100264665-9

Ibarra, Abril del 2015



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CONSTANCIA

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

Firma

Nombre: Luis Germán Correa Real

Cédula: 100264665-9

Ibarra, Abril del 2015



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

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

Yo, Luis Germán Correa Real, portador de la cédula de ciudadanía N° 100264665-9 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 Trabajo de Grado denominado: "Análisis comparativo entre la base de datos no relacional MongoDB y la base de datos relacional PostgreSQL", con el aplicativo: "Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent", que ha sido desarrollada para optar por el título de Ingeniería en Sistemas Computacionales, 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 mencionada, aclarando que el trabajo aquí descrito es de mi autoría y que no ha sido previamente presentado para ningún grado o calificación profesional.

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 de la ciudad de Ibarra.


Firma

Nombre: Luis Germán Correa Real

Cédula: 100264665-9

Ibarra, Abril del 2015



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

CERTIFICACIÓN

El señor egresado Luis Germán Correa Real, ha trabajado en el desarrollo del proyecto de Tesis "Análisis Comparativo Entre la Base de Datos no Relacional MONGODB con la Base de Datos Relacional POSTGRESQL.", con el aplicativo "Sistema para la Gestión de Clientes y Registro de Pagos de la Clínica Odontológica Ortho Dent.", previo a la obtención del Título de Ingeniero en Sistemas Computacionales, realizándola con interés profesional y responsabilidad, lo cual certifico en honor a la verdad.

A handwritten signature in blue ink, which appears to read "Mauricio Rea".

ING. MAURICIO REA
DIRECTOR DE TESIS

CERTIFICACIÓN DE ORTHO DENT



Ibarra, 15 de abril del 2015

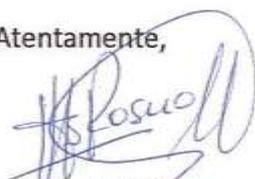
Señores
UNIVERSIDAD TÉCNICA DEL NORTE
Presente

De mis consideración.-

Siendo auspiciantes del proyecto de Tesis del Sr. LUIS GERMAN CORREA REAL, con CI: 1002646659 quien desarrollo su trabajo con el tema "SISTEMA PARA LA GESTIÓN DE USUARIOS Y REGISTRO DE PAGOS DE LA CLÍNICA ODONTOLÓGICA ORTHO DENT" me es grato informarle que se han superado a satisfacción las pruebas técnicas y la revisión de cumplimiento de los requerimientos funcionales, por lo que se recibe el proyecto como culminado y realizado por parte del Sr. LUIS GERMAN CORREA REAL.

El Sr. LUIS GERMAN CORREA REAL, puede hacer uso de este documento para los fines pertinentes en la Universidad Técnica del Norte.

Atentamente,



Dra. Beatriz Rosero
C.C.: 1102741756

Orlós Ray
Centro Radiológico Odontológico

Dr. Fernando Alvarado Hermida
ODONTÓLOGO-ORTODONCISTA
REG. SENEYC # 1005-07-663396
REG.MSP.#17



Ortho Dent
CLINICAS ODONTOLOGICAS ESPECIALIZADAS
RUC - 0400720470001
Av. Mariano Acosta 17- 42 y E. Ayala
Telf: 2606 440



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

DEDICATORIA

Esta tesis se la dedico a mi Dios quién supo guiarme por el buen camino, darme fuerzas para seguir adelante y no desmayar en los problemas que se presentaban, enseñándome a encarar las adversidades sin perder nunca la dignidad ni desfallecer en el intento.

A mi familia quienes por ellos soy lo que soy.

Para mis padres por su apoyo, consejos, comprensión, amor, ayuda en los momentos difíciles, y por ayudarme con los recursos necesarios para estudiar. Me han dado todo lo que soy como persona, mis valores, mis principios, mi carácter, mi empeño, mi perseverancia, mi coraje para conseguir mis objetivos.

A mis hermanas por estar siempre presentes, acompañándome para poderme realizar. A mi sobrino Sebastián quien ha sido y es una mi motivación, inspiración y felicidad.

Luis Germán Correa Real



UNIVERSIDAD TÉCNICA DEL NORTE
FACULTAD DE INGENIERÍA EN CIENCIAS APLICADAS

AGRADECIMIENTO

El eterno sentimiento de gratitud para la Universidad Técnica del Norte y a la Facultad de Ingeniería en Ciencias Aplicadas en especial a los profesores de la Escuela de Ingeniería en Sistemas Computacionales, quienes con sus conocimientos me han brindado la oportunidad de superarme personal, académica e intelectualmente.

Debo agradecer de manera especial y sincera al Ingeniero Mauricio Rea por aceptarme para realizar esta tesis bajo su dirección. Su apoyo y su capacidad para guiar mis ideas ha sido un aporte invaluable, no solamente en el desarrollo de esta tesis, sino también en mi formación como investigador. Las ideas propias, siempre enmarcadas en su orientación y rigurosidad, han sido la clave del buen trabajo.

Y, por supuesto, el agradecimiento más profundo y sentido va para mi familia. Sin su apoyo, colaboración e inspiración habría sido imposible llevar a cabo esta tesis.

A mis padres, German y Susana, por su ejemplo de lucha y honestidad; a mis hermanas por su tenacidad y superación; a mis amigos y amigas por su paciencia y capacidad de superación...por ellos y para ellos

Luis Germán Correa Real

RESUMEN

El presente proyecto está elaborado con la finalidad de conocer las ventajas y limitaciones de estas nuevas herramientas y su potencialidad para construir sistemas informáticos de uso comercial que complementará el conocimiento sobre bases de datos que deben tener los profesionales en esta área. Además de esto permitirá implementar un sistema para la Clínica Odontológica Ortho Dent.

Posteriormente se realizó un estudio de bases teóricas que acompaña y fundamenta el proyecto, y como está, define el proceso de las bases de datos no relacional como relacional.

El proceso de Soporte técnico estaba definido de manera simple y sencilla, esto se debía mejorar por lo que se redefinió el proceso y se documentó en el formato proporcionado por la Institución.

Luego de definir el proceso se procedió a la creación del sistema de software para la Clínica Odontológica Ortho Dent.

El software está diseñado siguiendo la metodología propia del RUP, la cual dicta que se desarrollará utilizando el framework de NetBeans, Base de Datos (PostgreSQL), JasperReports para la Vista de la Aplicación, JSP como Mapeador de Objetos Relacional.

Todo este proceso se lo realizó a través en un marco conversacional y reuniones con el personal de la Institución, lo cual nos lleva a magníficas conclusiones y competentes objetivos en cuanto a la prestación del servicio de la Clínica Odontológica Ortho Dent, lo que procede en el beneficio de tiempo y esfuerzo del propio.

SUMMARY

This project is developed in order to know the advantages and limitations of these new tools and their potential to build computer systems that complement commercial use knowledge databases must be professionals in this area. Besides this allow you to implement a system for Ortho Dental Clinic Dent.

Subsequently, a study and accompanying theoretical basis underlying the project was done, and how it defines the process of non-relational databases as relational data.

Support process was defined simply and easily, this should be improved so that the process is redefined and documented in the format provided by the institution.

After defining the process, I proceeded to the creation of the software system for Ortho Dental Clinic Dent.

The software is designed following the methodology of RUP, which dictates that the framework will be developed using NetBeans, Database (PostgreSQL), JasperReports for Vista Application, and JSP Objects as Relational Mapper.

This whole process was made through a conversational framework and meetings with staff of the institution, which leads to great conclusions and relevant targets for the provision of Ortho Dent Dental Clinic, which comes in benefit of their own time and effort.

ÍNDICE DE CONTENIDO

AUTORIZACIÓN DE USO Y PUBLICACIÓN	II
CONSTANCIA.....	IV
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	V
CERTIFICACIÓN	VI
DEDICATORIA.....	VIII
AGRADECIMIENTO.....	IX
RESUMEN	X
SUMMARY	XI
INDICE DE CONTENIDO.....	XII
ÍNDICE DE TABLAS	XVII
ÍNDICE DE ILUSTRACIONES	XX
CAPÍTULO I	1
1 INTRODUCCIÓN	1
1.1 TEMA	1
1.2 PROBLEMA	1
1.2.1 ANTECEDENTES	1
1.2.2 SITUACIÓN ACTUAL.....	2
1.2.3 PROSPECTIVA.....	3
1.2.4 DESCRIPCIÓN DEL PROBLEMA	4
1.3 OBJETIVOS	4
1.3.1 OBJETIVO GENERAL	4
1.3.2 OBJETIVOS ESPECÍFICOS.....	4
1.4 ALCANCE	5
1.5 JUSTIFICACIÓN	7
CAPÍTULO II	9
2 MARCO TEÓRICO.....	9
2.1 LAS BASES DE DATOS SQL.....	9

2.1.1	DEFINICIÓN.....	9
2.1.2	MODELAMIENTO DE BASES DE DATOS	9
2.1.3	ESTRUCTURA DE LAS BASES DE DATOS	10
2.1.4	TIPOS DE USUARIOS DE BASES DE DATOS	11
2.1.5	OBJETIVOS DE LOS SISTEMAS DE BASES DE DATOS	11
2.1.6	MODELO ENTIDAD – RELACIÓN	12
2.1.7	ENTIDADES.....	12
2.1.8	RELACIONES	13
2.1.9	ATRIBUTOS.....	13
2.1.10	CLAVES	14
2.1.11	LENGUAJE SQL	14
2.1.12	COMPONENTES DEL SQL.....	15
2.1.13	NIVELES DE ABSTRACCIÓN EN BASE DE DATOS.....	16
2.2	POSTGRESQL.....	17
2.2.1	DEFINICIÓN.....	17
2.2.2	HISTORIA	18
2.2.3	CARACTERÍSTICAS DE POSTGRESQL.....	19
2.2.4	ARQUITECTURA DE POSTGRESQL.....	23
2.2.5	COMPONENTES DEL GESTOR DE LA BASE DE DATOS POSTRESQL.....	23
2.2.6	RECURSOS PRINCIPALES DE HARDWARE Y SOFTWARE.....	24
2.2.7	ESQUEMA DE BÚSQUEDA DE LAS BASES DE DATOS RELACIONALES.....	25
2.2.8	HERRAMIENTAS DE DISEÑO DE POSTGRESQL.....	26
2.2.9	REQUISITOS DE INSTALACIÓN DE PGADMINIII	26
2.3	ANÁLISIS DE CARACTERÍSTICAS	27
2.3.1	TABLA COMPARATIVA.....	27
2.3.2	CRITERIOS DE EVALUACIÓN CONSIDERADOS	28
2.3.3	VENTAJAS DE POSTGRESQL.....	33
2.4	LAS BASES DE DATOS NOSQL	33
2.4.1	DEFINICIÓN.....	33
2.4.2	ESTRUCTURA DE LAS BASES DE DATOS NOSQL.....	34

2.4.3	ESQUEMAS Y VALIDACIÓN.....	35
2.4.4	TEOREMA CAP	36
2.4.5	CARACTERÍSTICAS DE LAS BASES DE DATOS NOSQL	37
2.4.6	TIPOS DE MODELOS DE BASES DE DATOS NOSQL	38
2.4.7	VENTAJAS DE LAS BASES DE DATOS NOSQL.....	41
2.4.8	DIFERENCIAS DE LAS BASES DE DATOS NOSQL	41
2.5	MONGODB	42
2.5.1	DEFINICIÓN.....	42
2.5.2	HISTORIA	43
2.5.3	CARACTERÍSTICAS DE MONGODB	44
2.5.4	ARQUITECTURA DE MONGODB.....	47
2.5.5	COMO FUNCIONA MONGODB	47
2.5.6	RECURSOS PRINCIPALES DE HARDWARE Y SOFTWARE.....	49
2.5.7	ESQUEMA DE BÚSQUEDA DE LAS BASES DE DATOS RELACIONALES.....	49
2.5.8	HERRAMIENTAS DE DISEÑO DE MONGODB.....	50
2.5.9	REQUISITOS DE INSTALACIÓN PARA MONGODB	50
2.6	ANÁLISIS DE CARACTERÍSTICAS	51
2.6.1	TABLA COMPARATIVA.....	51
2.6.2	CRITERIOS DE EVALUACIÓN CONSIDERADOS	52
2.6.3	VENTAJAS DE MONGODB.....	56
CAPÍTULO III	58
3	DESARROLLO DEL ESTUDIO COMPARATIVO.....	58
3.1	PARÁMETROS DE COMPARACIÓN.....	58
3.1.1	INTRODUCCIÓN	58
3.1.2	FASE DE PLANEACIÓN.....	58
3.1.3	PARÁMETROS ENTRE BASES DE DATOS	58
3.1.4	ESCENARIO DEL ANÁLISIS COMPARATIVO	59
3.1.5	FASE DE ANÁLISIS COMPARATIVO	59
3.1.6	INSTALACIÓN DE LAS BASES DE DATOS	60
3.1.7	SEGURIDAD DE LAS BASES DE DATOS	61

3.1.8 ADMINISTRACIÓN DE LAS BASES DE DATOS.....	62
3.1.9 TIEMPOS DE INSERCIÓN Y RECUPERACIÓN	63
3.1.10 USO DE RECURSOS	64
3.1.11 RESPUESTA FALLOS.....	65
3.1.12 COSTO DE LAS BASES DE DATOS	66
3.2 ANÁLISIS COMPARATIVO	67
3.3 ANÁLISIS FINAL.....	67
3.3.1 CONCLUSIÓN MONGODB	71
3.4 METODOLOGÍA DE DESARROLLO.....	72
3.4.1 INTRODUCCIÓN A RUP	72
3.4.2 HISTORIA DE RUP.....	73
3.4.3 RUP (PROCESO UNIFICADO RACIONAL).....	73
3.4.4 CARACTERÍSTICAS RUP	74
3.4.5 ESTRUCTURA DE LA METODOLOGÍA RUP.....	74
3.4.6 ESTRUCTURA DE LOS PROCESOS, ROLES, ACTIVIDADES, ARTEFACTOS Y FLUJOS DE TRABAJO	76
CAPÍTULO IV	78
4 DESARROLLO DEL SISTEMA DE GESTIÓN DE CLIENTES Y REGISTRO DE PAGOS PARA LA EMPRESA.	78
4.1 FASE DE INICIO.....	78
4.1.1 DOCUMENTO DE LA VISIÓN	78
4.1.2 PLAN DE DESARROLLO DE SOFTWARE.....	94
4.1.3 ACTAS DE TRABAJO.....	102
4.2 FASE DE ELABORACIÓN.....	108
4.2.1 INTRODUCCIÓN	108
4.2.2 VISTA DE CASOS DE USO	109
4.2.3 DESCRIPCIÓN DE LOS CASOS DE USO	117
4.2.4 VISTA DE RESTRICCIONES	131
4.2.5 TECNOLOGÍA.....	131
4.3 FASE DE CONSTRUCCIÓN	133

4.3.1 INTRODUCCIÓN	133
4.3.2 PLAN DE PRUEBAS.....	134
4.3.3 ESTRATEGIA DE PRUEBAS	135
4.4 FASE DE TRANSICIÓN.....	141
4.4.1 PLAN DE DESPLIEGUE	141
4.4.2 PASO A PRODUCCIÓN	141
4.4.3 TRANSFERENCIA TECNOLÓGICA Y CAPACITACIÓN.....	142
CAPITULO V	143
5 CONCLUSIONES Y RECOMENDACIONES.....	143
5.1 CONCLUSIONES.....	143
5.2 RECOMENDACIONES	144
5.3 ANÁLISIS DE IMPACTOS	145
5.4 GLOSARIO.....	148
5.5 INTRODUCCIÓN.	148
5.6 PROPÓSITO	148
5.7 REFERENCIAS	148
5.8 GLOSARIO DE TÉRMINOS	148
5.9 BIBLIOGRAFÍA	152
5.10 ANEXOS	154
MANUAL DE USUARIO	155

ÍNDICE DE TABLAS

TABLA 1: Comandos DDL	15
TABLA 2: Comandos DML	16
TABLA 3: Clausulas	16
TABLA 4: Versiones de PostgreSQL	19
TABLA 5: Limitaciones técnicas de la base de datos PostgreSQL	22
TABLA 6: Comparativa de PostgreSQL	28
TABLA 7: Criterios de Evaluación de PostgreSQL	29
TABLA 8: Información general de PostgreSQL	29
TABLA 9: Soporte para Sistemas Operativos de PostgreSQL	29
TABLA 10: Características Fundamentales de PostgreSQL	30
TABLA 11: Límites de la Base de Datos de PostgreSQL	31
TABLA 12: Tablas y Vistas de PostgreSQL	31
TABLA 13: Índices de PostgreSQL	31
TABLA 14: Capacidades Principales de PostgreSQL	32
TABLA 15: Tipos de Datos de PostgreSQL	32
TABLA 16: Otros Objetos de PostgreSQL	32
TABLA 17: Ejemplo Columnas	41
TABLA 18: Versiones de MongoDB	43
TABLA 19: Limitaciones técnicas de la base de datos MongoDB	46
TABLA 20: Comparativa MongoDB	52
TABLA 21: Criterios de Evaluación de MongoDB	53
TABLA 22: Información general MongoDB	53
TABLA 23: Soporte para Sistemas Operativos MongoDB	53
TABLA 24: Características Fundamentales MongoDB	54
TABLA 25: Límites de la Base de Datos MongoDB	54
TABLA 26: Capacidades Principales MongoDB	55
TABLA 27: Tipos de Datos MongoDB	56
TABLA 28: Escenario para Análisis	59

TABLA 29: Tabla de Valoración	60
TABLA 30: Análisis instalación	60
TABLA 31: Análisis: Seguridad	61
TABLA 32: Análisis: Administración	62
TABLA 33: Análisis: Tiempos de Peticiones y Recuperación	64
TABLA 34: Análisis: Recursos	65
TABLA 35: Análisis: Respuesta a fallos	65
TABLA 36: Análisis: Costos de las Bases de Datos	66
TABLA 37: Análisis: Análisis Final	67
TABLA 38: Definición del Problema	81
TABLA 39: Involucrados Interesados	82
TABLA 40: Involucrados Usuarios	82
TABLA 41: Coordinador del Sistema	83
TABLA 42: Responsable del Sistema	84
TABLA 43: Responsable Funcional	84
TABLA 44: Administrador del Sistema	85
TABLA 45: Auditor del Sistema	86
TABLA 46: Usuario del Sistema	86
TABLA 47: Necesidades de Usuario del Sistema	87
TABLA 48: Módulo de Gestión de Cliente	89
TABLA 49: Módulo de Facturación	89
TABLA 50: Módulo de Auditoria	90
TABLA 51: Módulo de Gestión de Aplicación	90
TABLA 52: Módulo de Reportes	90
TABLA 53: Costos y Precios	91
TABLA 54: Roles y Responsabilidades	99
TABLA 55: Plan de Fases	99
TABLA 56: Objetivos de Iteraciones	100
TABLA 57: Descripción de Usuarios	113
TABLA 58: Prioridad de los Casos de Uso	116

TABLA 59: Descripción Administrar Usuarios.....	117
TABLA 60: Descripción Administrar Sesiones.....	118
TABLA 61: Descripción Control de Acceso.....	119
TABLA 62: Descripción Gestionar Clientes	120
TABLA 63: Descripción Registrar Recibos	121
TABLA 64: Descripción Registrar Facturas	122
TABLA 65: Descripción Registrar Cotizaciones.....	123
TABLA 66: Descripción Registrar Consultas	124
TABLA 67: Descripción Analizar Pistas y Reportes en el Tiempo.....	125
TABLA 68: Descripción Mostrar Reportes	126
TABLA 69: Descripción Gestionar Parámetros.....	127
TABLA 70: Descripción Gestionar Roles	128
TABLA 71: Descripción Gestionar Productos	129
TABLA 72: Descripción Gestionar Servicios.....	130
TABLA 73: Tecnologías	132
TABLA 74: Pruebas de Integridad de Datos	136
TABLA 75: Pruebas del Sistema.....	136
TABLA 76: Pruebas del Ciclo del Negocio.....	137
TABLA 77: Pruebas de Interfaz de Usuario	137
TABLA 78: Pruebas de Seguridad y Control de Acceso.....	138
TABLA 79: Recursos.....	138
TABLA 80: Entregables.....	138
TABLA 81: Lista de Riesgos	139
TABLA 82: Actividades de Paso a Producción	142
TABLA 83: Análisis de impactos	146

ÍNDICE DE ILUSTRACIONES

ILUSTRACIÓN 1: Diagrama de Bloques del Módulo del Sistema	5
ILUSTRACIÓN 2: Implementación del Sistema a Realizar	6
ILUSTRACIÓN 3: Arquitectura del Sistema	7
ILUSTRACIÓN 4: Estructura de las bases de datos	10
ILUSTRACIÓN 5: Entidades y Relaciones	13
ILUSTRACIÓN 6: Atributos	13
ILUSTRACIÓN 7: Claves.....	14
ILUSTRACIÓN 8: Niveles de abstracción	17
ILUSTRACIÓN 9: Esquema de funcionamiento PostgreSQL	23
ILUSTRACIÓN 10: Esquema de Componentes de PostgreSQL.....	23
ILUSTRACIÓN 11: Esquema de búsqueda de PostgreSQL.....	25
ILUSTRACIÓN 12: Estructura de las bases de datos noSQL.....	35
ILUSTRACIÓN 13: Teorema CAP	36
ILUSTRACIÓN 14: Ejemplo Clave/Valor	38
ILUSTRACIÓN 15: Arquitectura de base de datos MongoDB	47
ILUSTRACIÓN 16: Esquema de búsqueda de PostgreSQL.....	49
ILUSTRACIÓN 17: Porcentaje de éxito.....	70
ILUSTRACIÓN 18: Estructura de Procesos, Roles, Actividades, Artefactos	76
ILUSTRACIÓN 19: Prospectiva del producto	88
ILUSTRACIÓN 20: Calendario de las tareas de la Metodología de Desarrollo RUP	101
ILUSTRACIÓN 21: Arquitectura del Sistema	109
ILUSTRACIÓN 22: Procesos Módulo de Gestión de Usuarios.....	110
ILUSTRACION 23: Procesos Modulo de facturación.....	110
ILUSTRACIÓN 24: Procesos Módulo de Gestión de Aplicación.....	111
ILUSTRACIÓN 25: Procesos Módulo de Auditoría	111
ILUSTRACIÓN 26: Procesos Módulo de Reportes	112
ILUSTRACIÓN 27: Actores del Sistema	112
ILUSTRACIÓN 28: Modelos de Casos de Negocios	113

ILUSTRACIÓN 29: Diagrama de Flujo de Páginas	114
ILUSTRACIÓN 30: Modelo de Caso de Uso Módulo de Gestión de Usuarios	114
ILUSTRACIÓN 31: Modelo de Caso de Uso Módulo de Facturación	115
ILUSTRACIÓN 32: Modelo de Caso de Uso Módulo de Gestión de Aplicación.....	115
ILUSTRACIÓN 33: Modelo de Caso de Uso Módulo de Auditoría	115
ILUSTRACIÓN 34: Modelo de Caso de Uso Módulo de Reportes	116
ILUSTRACIÓN 35: Administrar Usuarios	118
ILUSTRACIÓN 36: Administrar Sesiones	119
ILUSTRACIÓN 37: Control de Acceso	120
ILUSTRACIÓN 38: Registrar Clientes.....	121
ILUSTRACIÓN 39: Registrar Recibos.....	122
ILUSTRACIÓN 40: Registrar Facturas.....	123
ILUSTRACIÓN 41: Registrar Cotizaciones	124
ILUSTRACIÓN 42: Realizar Consultas	125
ILUSTRACIÓN 43: Analizar Reportes en el Tiempo.....	126
ILUSTRACIÓN 44: Mostrar Reportes.....	127
ILUSTRACIÓN 45: Gestionar Parámetros	128
ILUSTRACIÓN 46: Gestionar Roles.....	129
ILUSTRACIÓN 47: Gestionar Productos.....	130
ILUSTRACIÓN 48: Gestionar Servicios	131
ILUSTRACIÓN 49: Diagrama de Entidades SYS-OD.....	134
ILUSTRACIÓN 50: Análisis de impactos.....	146

CAPÍTULO I

1 INTRODUCCIÓN

1.1 TEMA

El tema a desarrollar es el Análisis Comparativo entre la Base de Datos no Relacional MONGODB con la Base de Datos Relacional POSTGRESQL. Con el Aplicativo Sistema para la Gestión de Clientes y Registro de Pagos de la Clínica Odontológica Ortho Dent.

1.2 PROBLEMA

1.2.1 ANTECEDENTES

Desde hace 40 se han distribuido y manejado bases de datos relacionales las cuales generan un sin fin de información de nuevas aplicaciones y software que existen en nuestro medio y el mundo y que se producen de manera corpulenta en pocos años al mundo digital, es allí donde se ha visto la necesidad de que estos sistemas escalables se conviertan en un medio para afrontar estos volúmenes de información.

Las bases de datos NOSQL (not only SQL) o no relacionales aparecieron como una necesidad para el manejo de información en cantidades enormes a través de la web.

Las bases de datos no relacionales surgieron desde hace 50 años por lo que se ha desarrollado software para almacenamiento masivo, datos multivalor, se han creado datos del orden de exabytes¹ por año. Donde los datos son más entrelazados y conectados, y menos estructurados y datos a escala de la web, con mucha lectura y escritura, los esquemas² (“schemas”) cambian a menudo, por ejemplo las aplicaciones sociales no necesitan el mismo nivel de la prueba ACID y la orientación del software es hacia servicios (PaaS: programas como Servicios).

¹Exabyte: Es una unidad de medida de almacenamiento de Datos cuyo símbolo es el 'EB', equivale a 1018 bytes.

²Esquema: En una base de datos, el esquema define sus tablas, sus campos en cada tabla y las relaciones entre cada campo y cada tabla.

Es por esto el importante análisis y estudio de nuevas herramientas de bases de datos no relacionales tales como MONGODB, lo que facilitará su desarrollo, modificación y prueba de nuevas aplicaciones.

La Clínica Ortho Dent tiene sus inicios en 1994 con el nombre de Clínica Dental y Ortodoncia, ubicada en los altos de Davivienda. En 1999 luego de la especialización en ortodoncia del Dr. Alvarado se crea Ortho Dent como clínica Odontológica Especializada y Ortho Ray como centro radiológico. Esta ampliación garantiza un mejor servicio, satisfacción y comodidad del usuario. En el 2001 se incorpora la Dra. Beatriz Rosero prestando su servicio como Odontóloga General. Además, se inaugura una nueva sede en la ciudad de Tulcán (Ecuador). Esta empresa garantiza un servicio satisfacción para comodidad del cliente.

Ortho Dent es una Clínica Odontológica especializada que trabaja para beneficio las necesidades de sus pacientes, cuentan con tecnología de punta en las diferentes especialidades de la Odontología, lo que nos permite garantizar los procedimientos, el mejoramiento continuo y un excelente servicio.

La Clínica Odontológica Ortho Dent de la ciudad de Ibarra cuenta con un sistema manual de registro de cada uno de sus clientes, el cual almacena información personal detallada en ficheros, además tiene un control de pagos que se realiza con la anotación de cada uno en agendas. La forma de pago se la realiza en el momento de la consulta o en los días siguientes, del total de la cantidad o en varios plazos, es por ello que se ven en la necesidad de sistematizar estos procesos utilizando tecnología moderna y adecuada para esta institución.

1.2.2 SITUACIÓN ACTUAL

Actualmente en el desarrollo de sistemas informáticos se usan bases de datos relacionales en lenguaje SQL³(Lenguaje de Consulta Estructurado) para la creación y manipulación de estas, sin embargo en los últimos tiempos han aparecido nuevas arquitecturas de bases de datos, en respuesta a

³SQL: Es un lenguaje de acceso a bases de datos que explota la flexibilidad y eficacia de sistemas relacionales.

requerimientos de nuevos formatos de manejo de la información, por lo que es necesario que los profesionales conozcan este tipo de arquitecturas que no son conocidas y que están incorporando muy fuertemente en el mercado.

En la actualidad la Clínica Odontológica Ortho Dent brinda a la ciudadanía una clínica a disposición. Además siguiendo las exigencias y expectativas de los clientes, la clínica ha desarrollado un área especializada y exclusiva para niños llamada Ortho Dent Kids, donde personal capacitado en el tratamiento de salud oral de los niños.

La Clínica Odontológica Ortho Dent, tiene insuficiencia al momento de manejar la información, como el control y registro adecuado de sus clientes, contiene información muy limitada en sus fichas, de igual manera al momento de tener un paciente con cierta antigüedad su carpeta contiene un sin fin de hojas que pueden deteriorarse con el paso del tiempo. Otra falencia es el no contar con ningún tipo de generación de informes. Por último la Clínica no cuenta con un respaldo de la información en caso de la pérdida de la misma.

1.2.3 PROSPECTIVA

A través de este análisis a realizarse, se establecerá documentación disponible para estudiantes y profesionales interesados en perfeccionar e instituirse este framework de bases de datos no relacional lo que acortará el desconocimiento del mismo, aumentará el desarrollo de aplicaciones con la base de datos no relacional MONGODB en el mercado y garantizará sistemas de calidad.

Con el desarrollo del sistema en la Clínica Odontológica Ortho Dent se anhela emplear los conocimientos e indicar las ventajas de usar bases de datos no relacionales con la base de datos MONGODB.

Este proyecto permitirá a la Clínica Odontológica Ortho Dent obtener información real de sus clientes, alcanzando un valioso nivel de confiabilidad sobre sus usuarios y además de su importante estado financiero, y alcanzando el éxito de los objetivos que la empresa apremia.

1.2.4 DESCRIPCIÓN DEL PROBLEMA

En el Dominio de base de datos relacionales, se han establecido nuevas arquitecturas de base de datos genéricas no relacionales; en nuestro medio estas herramientas no son conocidas ya que a que a nivel internacional se está escuchando mucho sobre ellas, sin embargo, en la Carrera de Ingeniería en Sistemas Computacionales no constan en el Pensum de Estudios por lo que es importante conocer su potencialidad en el desarrollo de sistemas informáticos.

La falta de información en el medio genera ignorancia de la base de datos no relacional MONGODB en estudiantes y profesionales, lo que minimiza el uso de la base de datos no relacionales.

Además de las excesivas limitantes en la Clínica Odontológica Ortho Dent que dificultan la adecuada atención al cliente como es el caso de la seguridad y búsqueda de la información, en el control de pagos, debido a la falta de un sistema informático y teniendo en la actualidad un sin número de programas que facilitan estos tipo de problemas y están al alcance de nosotros.

1.3 OBJETIVOS

1.3.1 OBJETIVO GENERAL

Realizar un estudio comparativo entre el sistema de bases de datos no relacional MongoDB y el sistema de bases de datos relacional PostgreSQL en el desarrollo de sistemas informáticos.

1.3.2 OBJETIVOS ESPECÍFICOS

1. Realizar un estudio de las características de base de datos no relacional MongoDB.
2. Realizar un estudio de las características de base de datos relacional PostgreSQL.
3. Definir parámetros de comparación de las dos herramientas.
4. Desarrollar un sistema para la gestión de clientes en la Clínica Odontológica Ortho Dent.

5. Establecer ventajas y desventajas de la base de datos relacional PostgreSQL y de la base de datos no relacional MongoDB en el desarrollo de sistemas informáticos.

1.4 ALCANCE

El estudio cubrirá el análisis técnico de las características de la base de datos no relacional llamada MongoDB, para la construcción de bases de datos no relacionales. Además se estudiará su integración con la plataforma NetBeans en lenguaje Java. Asimismo se realizara el estudio de las características técnicas de las base de datos PostgreSQL para la construcción de bases de datos relacionales. También se estudiará su integración con la plataforma NetBeans en lenguaje Java.

Se investigará la librería de generación de informes JasperReport para el mejor manejo de la información y vistas, esta librería tiene la capacidad de generar contenido enriquecido al monitor, ya sea en formatos PDF, HTML, XLS, CSV y XML destinadas a facilitar las vistas para mostrar información.

Para aplicar los resultados obtenidos en el análisis, luego de haber elaborado un análisis de las ventajas y desventajas del uso de la base de datos no relacional MongoDB y de la base de datos relacional PostgreSQL, se desarrollará un sistema para la gestión de clientes y registro de pagos de la empresa, en este caso para la Clínica Odontológica Ortho Dent.

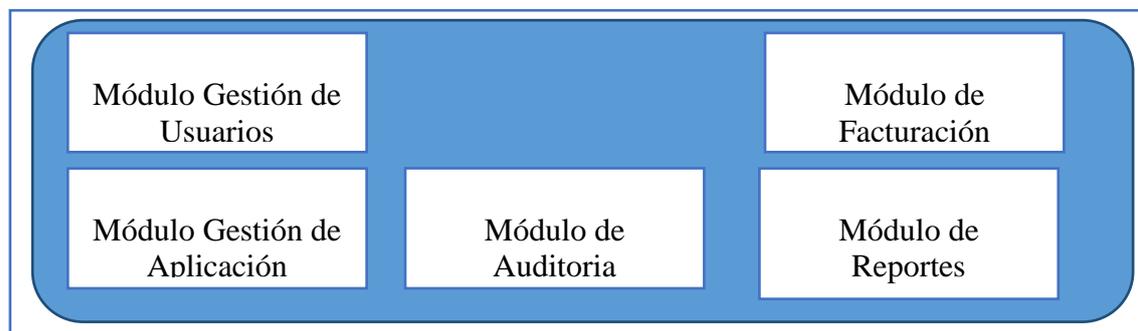


ILUSTRACIÓN 1: Diagrama de Bloques del Módulo del Sistema

Fuente: Diagrama de Bloques del Módulo del Sistema



ILUSTRACIÓN 2: Implementación del Sistema a Realizar

Fuente: Implementación del Sistema a realizar

El sistema brindará beneficios a la empresa antes mencionada, ya que permitirá registrar y generar información que será útil para optimizar ciertos procesos como:

- ✓ Realizar la gestión y el control de clientes de forma segura.
- ✓ Valorar costos de los clientes en el registro facturación.
- ✓ Totalizar tiempos y costos de cada cliente.

El sistema está estructurado por los siguientes Módulos:

MÓDULO DE GESTIÓN DE USUARIOS

Llevará la creación, eliminación, actualización de cada usuario que va a hacer uso del sistema en la empresa.

MÓDULO DE FACTURACIÓN

Se registrará la facturación y el seguimiento de clientes en sus pagos, además se generará un listado en el cual indique los diferentes pagos de clientes en el tiempo.

MÓDULO DE AUDITORIA

Se asegurara el sistema entre usuarios como los tiempos de ingresos como los campos que altera.

MÓDULO DE REPORTES

Se podrá desarrollar una lista de información del cliente para conocer sus pagos realizados y no abonados.

MÓDULO DE GESTIÓN DE APLICACIÓN

Se podrá controlar los diferentes parámetros que la Clínica Odontológica Ortho Dent está utilizando.

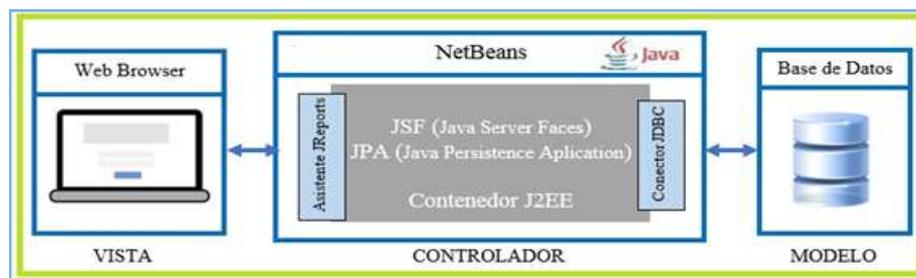


ILUSTRACIÓN 3: Arquitectura del Sistema

Fuente: Propia

1.5 JUSTIFICACIÓN

El análisis comparativo de la base de datos no relacional MONGODB frente a PostgreSQL permitirá conocer ventajas y limitaciones de estas nuevas herramientas y su potencialidad para construir sistemas informáticos. Complementará el conocimiento sobre bases de datos que deben tener los profesionales en esta área.

Brindará documentación suficiente que servirá de guía de estudio para estudiantes y profesionales de la informática que ampliarán sus conocimientos y garantizarán construcción de aplicaciones de calidad, Motivará a desarrolladores de software a utilizar estas bases de datos no relacionales y nuevas herramientas en el desarrollo de sus sistemas en el mercado.

Incentivará a estudiantes y profesionales a seguir realizando investigaciones de nuevas bases de datos no relacionales, ya que es la tendencia de los últimos tiempos.

También es importante destacar que el sistema beneficiará a la Clínica Odontológica Ortho Dent de tal manera que contribuirá a esta empresa en la gestión de clientes y pagos.

También permitirá a los propietarios conocer con precisión el estado de las cuentas, verificar su exactitud y confiabilidad de los datos obtenidos, y garantiza su estabilidad, permitiendo trabajar con mayor rapidez y eficacia para dar soluciones rápidas y eficaces.

CAPÍTULO II

2 MARCO TEÓRICO

2.1 LAS BASES DE DATOS SQL

2.1.1 DEFINICIÓN

Las bases de datos son una colección organizada de información. Los datos suelen mostrarse como imágenes, gráficos, texto y números entre otros y frecuentan estar organizadas para modelar los aspectos importantes de la realidad. Los sistemas de gestión DBMS ⁴están especialmente para interactuar con el usuario, existen otras aplicaciones que en sí capturan y analizan los datos obtenidos. Determinadamente, la base de datos se refiere a los datos en sí. Las bases de datos se crean para manipular grandes cantidades de información introduciendo, almacenamiento, recuperación y gestión de esa información.

Es un conjunto absoluto no redundante de datos estructurados organizados y su implementación en máquina sea accesible en tiempo real y compatible con usuarios concurrentes con necesidad de información diferente y no predicable en tiempo.

Las base de datos surgen desde mediados de los años sesenta la historia de las bases de datos, en 1970 Codd propuso el modelo relacional, este modelo es el que ha marcado la línea de investigación por muchos años, ahora se encuentran los modelos orientados a objetos⁵.

2.1.2 MODELAMIENTO DE BASES DE DATOS

En una modelación de bases de datos se refleja una representación de manera conceptual de los datos en una base de datos. Su estructura depende de los objetos de datos, las asociaciones entre datos objeto, y las reglas que rigen las operaciones en los objetos.

⁴DBMS: (Sistema Manejador de Bases de Datos). Es un software que controla la organización, almacenamiento, recuperación, seguridad e integridad de datos.

⁵Data Modeling, 2012, tomado de

<https://www.liberty.edu/media/1414/%5B6330%5DERDDataModeling.pdf>

El modelo de datos es una técnica para construir una base de datos, por lo que tiene que ser lo suficientemente simple para mostrarlo de manera detallada al usuario final y pueda entender sus funciones, características, estructuras de datos y después ser utilizada y seguir con una la estructura física⁶.

2.1.3 ESTRUCTURA DE LAS BASES DE DATOS

En la estructura de bases de datos se definen dos partes principales como son el hardware y el software, en Hardware se almacena la información en discos, tambores, cintas, entre otras; en Software se encuentra el SGBD o DBMS donde se administrara la información y las bases de datos.

Lo más importante de entre todas, los datos, los cuales serán almacenados en una estructura externa en este caso el hardware y luego procesados para crear información o software.

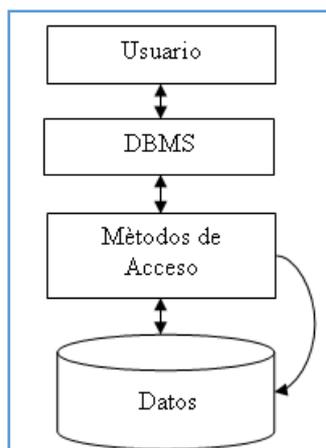


ILUSTRACIÓN 4: Estructura de las bases de datos

Fuente: Iriana Nadia Strycek 2010, Sistemas de Bases de Datos Multiplataforma para Aplicaciones Distribuidas pág. 10

Definiendo la estructura de las bases de datos, primero se necesita de un usuario para hacer el uso y manipulación del sistema gestor de bases de datos donde se encontraran la información a utilizar, a través de estos manejadores utilizaremos el método de acceso a los datos, me refiero a los diferentes comandos que tiene almacenada la DBMS para hacer uso de la información y presentarla al usuario.

⁶"DOMISOFT" Ing. Edgar Calderón Sánchez., 2014, tomado de http://www.domisoftonline.com/porta_web/modulo%20de%20bases%20de%20datos.pdf.

2.1.4 TIPOS DE USUARIOS DE BASES DE DATOS

Usuario Final: Es el sujeto que no entra en contacto con el código de la aplicación y solo utiliza los datos que puede observar en el dispositivo, esta persona ve datos convertidos en información, quiere decir que solo la utiliza sin saber qué es lo que pasa en el interior del sistema

Desarrollador: Es la persona que se desenvuelve en la codificación y programación de los sistemas que interactúan con la Base de Datos.

DBA (Data Base Administrador): Es el elemento que asegura integridad, consistencia, redundancia, seguridad de la información. Administra la Base de Datos y se encarga de realizar el mantenimiento de los datos.

2.1.5 OBJETIVOS DE LOS SISTEMAS DE BASES DE DATOS

De entre los más importantes tenemos:

- ✓ **Restringir la redundancia e inconsistencia de datos:** Es muy posible que en algún lugar de la base de datos se encuentre un mismo dato repetido, la redundancia⁷ aumenta el espacio en el almacenamiento, además de esto puede crear una inconsistencia⁸ de la información.
- ✓ **Reducir la dificultad para tener acceso a los datos:** Se deben considerar varias maneras de presentar la información por lo que necesitamos mostrar las diferentes formas de entregar los datos conveniente y eficiente a los usuarios finales si es el caso a un importante jefe de negocios.
- ✓ **Evitar el aislamiento de los datos:** Sucede cuando se quiere realizar un cambio a los datos en archivos, entonces estos cambios deben ser visibles para las demás operaciones que el DBMS ejecuta en la base de datos.
- ✓ **Disminuir los problemas de seguridad:** La seguridad es muy trascendente al momento de implementar una aplicación que contiene información de suma importancia como por ejemplo cuentas bancarias.

⁷Redundancia: Quiere decir que un dato almacenado se encuentren guardado varias veces en diferentes lugares.

⁸Inconsistencia: Cuando existe redundancia puede existir inconsistencia y nos dice que si un dato es actualizado en un sitio, puede que otro dato duplicado este no actualizado.

2.1.6 MODELO ENTIDAD – RELACIÓN

Para realizar la comparación entre la base de datos relacional POSTGRESQL con la base de datos no relacional MONGODB utilizare el modelo entidad relación.

El modelo Entidad - Relación es un modelo de datos conceptual que consta de un repertorio de objetos básicos entidades⁹, y de relaciones entre objetos que se presentan en el mundo real. El diagrama entidad-relación se utiliza para representar objetos de los datos de la base de datos.

Las ventajas del Modelo Entidad Relación:

- a) Que es simple y fácil de comprender. El modelo puede ser utilizado por un diseñador de bases de datos para indicar el diseño final al usuario.
- b) El modelo puede ser utilizado como un plan de diseño por el desarrollador de la base de datos a poner en práctica un modelo de datos en un software específico de gestión de base de datos¹⁰.
- c) Estos diagramas entidad relación, permiten conservar una punto de vista global del diseño y beneficia la comunicación de diseñadores.

Las desventajas del Modelo Entidad Relación:

- a) En los sistemas gestores de bases de datos no se lo pueden implementar directamente.
- b) Habitualmente se los debe transformar en un modelo de más bajo nivel.

2.1.7 ENTIDADES

La parte más significativa de las bases de datos son las entidades, pueden ser de forma concreta o abstracta, como persona, lugares, cosas o eventos que tienen importancia para la base de datos. Algunos ejemplos específicos de entidades son empleados, cuentas, tickets. Una entidad es semejante a una tabla en el modelo relacional.

⁹Entidades: Es un objeto donde se recoge información de interés de cara a la base de datos.

¹⁰Data Modeling, 2012, tomado de

<https://www.liberty.edu/media/1414/%5b6330%5derddatamodeling.pdf>

Las entidades se clasifican como independientes (fuertes) o dependientes (débiles). La entidad fuerte es aquel que no necesita depender de ninguna otra entidad, una entidad débil necesita depender de una entidad.

2.1.8 RELACIONES

Una Relación representa una asociación entre dos o más entidades, ejemplo: los clientes están asignados a una ficha que tiene sus datos personales con sus citas y pagos donde deberán acercarse a pagar.

La manera de mostrar las entidades y relaciones, es a través de un rectángulo en el cual se indicara el nombre de la entidad y además si existe relación entre ellas se debe unir con líneas, estas entidades dependiendo de la cordialidad¹¹.

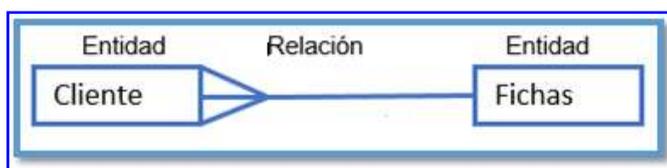


ILUSTRACIÓN 5: Entidades y Relaciones

Fuente: Propia

2.1.9 ATRIBUTOS

Los atributos describen la entidad de la que están asociados, son las propiedades de una entidad. Ej. Entidad cliente (nombre, apellido, dirección, teléfono).

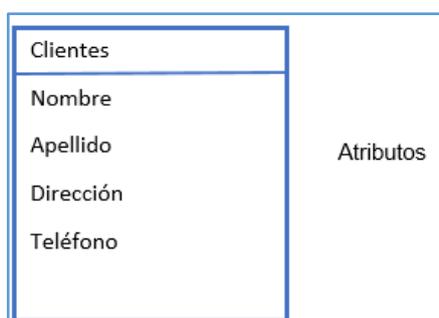


ILUSTRACIÓN 6: Atributos

Fuente: Propia

¹¹Cardinalidad: Indica el número de entidades con las que puede estar relacionada una entidad dada, estas pueden ser (uno a uno) (uno a varios) (varios a no) (varios a varios).

2.1.10 CLAVES

Toda entidad, posee un atributo esencial que se describe como clave primaria, este atributo se lo aprovecha para diferenciar a cada una de las demás entidades, las claves se dividen en:

- a) **Clave Primaria:** Esta clave, se determina para identificar a una entidad de las demás, además esta clave es elegida inequívocamente por el diseñador
- b) **Súper Clave:** Es un subconjunto de atributos que distingue a la entidad de diversas entidades. Si se agrega un atributo a la entidad, esta seguirá siendo una súper clave
- c) **Clave Candidata:** si se agrega una súper clave a la entidad, un atributo es quitado de esta entidad, de tal manera que la súper clave se convertirá en una clave candidata.

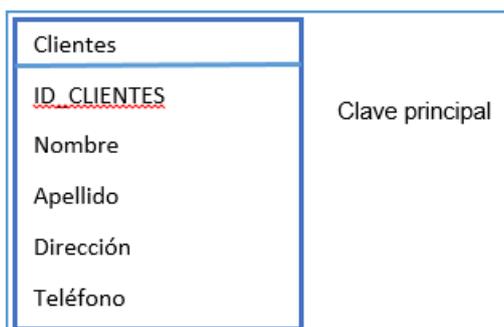


ILUSTRACIÓN 7: Claves

Fuente: Propia

2.1.11 LENGUAJE SQL

SQL (Structured Query Language) es un lenguaje de programación diseñado para almacenar, manipular y recuperar datos almacenados en bases de datos relacionales, la primera encarnación de SQL apareció en 1974, cuando un grupo de IBM desarrolló el primer prototipo de una base de datos relacional. Relational Software (luego se convirtió en Oracle) lanzó la primera base de datos relacional comercial¹².

¹²Tutorial de SQL: Curso de SQL, tomado de <http://www.1keydata.com/es/sql/>

El SQL es el lenguaje estándar ANSI/ISO¹³ de definición, manipulación y control de bases de datos relacionales; concretamente, se parece al inglés. El SQL es un lenguaje con el que se puede acceder a todos los sistemas relacionales comerciales¹⁴.

La creación de la base de datos consiste en la creación de las tablas que la componen. En realidad, antes de poder proceder a la creación de las tablas, normalmente hay que crear la base de datos. De esta manera, para una DBMS se pueden gestionar diferentes bases de datos independientes al mismo tiempo sin que se den conflictos con los nombres que se usan en cada una de ellas¹⁵.

2.1.12 COMPONENTES DEL SQL

El lenguaje SQL está compuesto por comandos, cláusulas, operadores. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos¹⁶.

Comandos

Existen varios tipos de comandos como los DDL¹⁷ (lenguaje de definición de datos), las DML¹⁸ (lenguaje de manipulación de datos), Cláusulas que son condiciones para seleccionar información de la tabla necesaria. Los cuales nos ayudarán a un mejor manejo de la información.

TABLA 1: Comandos DLL

DDL	
Create	Crea entidades en la base de datos
Alter	Edita o modifica los atributos de la base de datos
Drop	Elimina o borra los atributos de la base de datos
Delete	Elimina todos los registros de la base de datos

Fuente: El autor

¹³ANSI/ISO (American National Standards Institute): es una organización sin fines de lucro que supervisa el desarrollo de estándares para productos, servicios, procesos y sistemas.

¹⁴El lenguaje SQL, tomado de http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02149.pdf

¹⁵Crear la base de datos, tomado de http://www.htmlpoint.com/sql/sql_07.htm

¹⁶SQL, tomado de <http://www.monografias.com/trabajos11/manu/manu.shtml#ixzz37TDy0GU>

¹⁷DDL: permite la creación y modificación de la estructura de las entidades o tablas así como objetos.

¹⁸DML: Se las utiliza para gestionar la información a través de consultas, filtros y extracciones de datos.

TABLA 2: Comandos DML

DML	
Select	Selecciona registros de la base de datos
Insert	Ingresar nuevos datos a una tabla de la base de datos
Update	Actualiza los campos dentro de una tabla de la base de datos
Delete	Elimina tablas y la información que se contengan en ellas

Fuente: El autor

TABLA 3: Clausulas

Clausulas	
From	Se usa para especificar la tabla de la cual se va a obtener la información
Where	Se usa para definir los requisitos que la información debe tener para presentarse
Group by	Nos ayudara a separar registros en grupos
Order by	Se usa para ordenar registros mediante un orden específico

Fuente: El autor

2.1.13 NIVELES DE ABSTRACCIÓN EN BASE DE DATOS

Existen tres tipos de niveles:

Interno: Nos indica cómo los datos se almacenan en la realidad, es decir a un medio físico a través del manejo de hardware; por medio de un sistema operativo se facilitaría y agilizaría el desarrollo.

Externo: Es la parte más cercana al usuario final, en la cual se puede ver como se manejan datos y se están transformando en información para que sea entendida por el usuario. Otra manera de verla, es en la que el usuario puede apreciar dichos datos y sus relaciones entre ellos.

Conceptual: Es donde se definen estructuras lógicas y físicas, entre almacenamiento de datos y relaciones entre ellas. Indica la forma de estructuración de la base datos.



ILUSTRACIÓN 8: Niveles de abstracción

Fuente: Piñeiro Gómez, José Manuel, “Bases de Datos Relacionales y Modelado de Datos”
Segunda Edición 2013

2.2 POSTGRESQL

2.2.1 DEFINICIÓN

POSTGRESQL, es un sistema de gestión de base de datos relacional y libre, que agiliza la interacción de cliente, servidor y base de datos, donde PostgreSQL es el que realiza la mayoría del trabajo referente a bases de datos cuando se le hacen peticiones.

También es un gestor de bases de datos orientadas a objetos, usado en entornos de software libre, porque cumple los estándares (SQL92 y SQL99)¹⁹ y también un conjunto de funcionalidades avanzadas que soporta, fue desarrollada en la universidad de Berkeley, y se distribuye bajo licencia BSD Es el sistema de gestión de bases de datos de código abierto más potente del mercado.²⁰

Fue uno de los pioneros en conceptos existentes en el desarrollo objeto relacional, además de esto incluye características de orientación a objetos, tipos de datos, herencia, triggers, funciones, procedimientos almacenados, integridad transaccional, entre otras.

¹⁹SQL92 Y SQL99: Es un estándar o norma para la estandarización de sistemas gestores de bases de datos. Tomado de <http://savage.net.au/SQL/sql-92.bnf.html>

²⁰PostgreSQL, tomado de <http://postgresql-dbms.blogspot.com/p/importante-sobre-el-dbms-postgresql.html>.

PostgreSQL integra un modelo cliente servidor, usa multiprocesos para la buena estabilidad de la aplicación, si es que falla uno de estos proceso pues no afectara a otros procesos.

El proyecto PostgreSQL es manejado por miles de programadores y desarrolladores a nivel mundial y se sigue desarrollando hasta la fecha gracias al ser un sistema de gestor de bases de datos de código abierto, con la fuente disponible para seguir desplegándolo.

2.2.2 HISTORIA

Se inicia en 1986 como un proyecto realizado por el profesor Michael Stonebraker y su equipo de desarrolladores de la Universidad Berkeley (California), el nombre original del proyecto era “POSTGRES” donde indicaban conceptos de bases de datos orientadas a objetos.²¹

Postgres se comercializo por Illustra, que fue una empresa comercializadora de software y que ahora forma parte de Informix. Tiempo es lo que los absorbía para mantener este software de bases de datos por lo que los desarrolladores en 1993 sacaron la versión 4.5 al mercado por lo que dio por terminado el proyecto en esta universidad.

Por lo que en 1994 Yu yJolly Chen agregaron SQL a Postgres para posteriormente liberar su código en la web con el nombre de postgres95 en donde se mejora su legibilidad y rendimiento. En 1996 se cambia el nombre por PostgreSQL retomando la continuidad de las versiones.

²¹ Bases de datos en PostgreSQL, tomado de http://ocw.uoc.edu/computer-science-technology-and-multimedia/bases-de-datos/bases-de-datos/P06_M2109_02152.pdf

TABLA 4: Versiones de PostgreSQL

Versión	Versión menor	Soportada	Lanzamiento	Soporte
9.2	9.2.0	Si	Sep 2012	Sep 2017
9.1	9.1.5	Si	Sep 2011	Sep 2016
9.0	9.0.9	Si	Sep 2010	Sep 2015
8.4	8.4.13	Si	Jul 2009	Jul 2014
8.3	8.3.20	Si	Feb 2008	Feb 2013
8.2	8.2.23	No	Dic 2006	Dic 2011
8.1	8.1.23	No	Nov 2005	Nov 2010
8.0	8.0.26	No	Ene 2005	Oct 2010
7.4	7.4.30	No	Nov 2003	Oct 2010
7.3	7.3.21	No	Nov 2002	Nov 2007
7.2	7.2.8	No	Feb 2002	Feb 2007
7.1	7.1.3	No	Abr 2001	Abr 2006
7.0	7.0.3	No	May 2000	May 2005
6.5	6.5.3	No	Jun 1999	Jun 2004
6.4	6.4.2	No	Oct 1998	Oct 2003
6.3	6.3.2	No	Mar 1998	Mar 2003

Fuente: El autor

2.2.3 CARACTERÍSTICAS DE POSTGRESQL

Las características técnicas de la base de datos PostgreSQL la forman para hacer una de las más robustas y potentes del mercado, tales como su facilidad de administración, estabilidad, tiempo en transacciones menor, entre otras. PostgreSQL trabaja con grandes cantidades de información y asimismo de usuarios conectados indefinidamente en la aplicación.

- ✓ Soporta diferentes tipos de datos como: generales de todas las bases de datos además de, fecha, gráficos, cadena de bits, monetarios, datos de tipo (MAC, IP, entre otras).
- ✓ Tipo objeto relacional, trabaja a base de rutinas y reglas multiusuario, transacciones, optimización de consultas, herencia y arreglos.
- ✓ Maneja caracteres internacionales.
- ✓ Licencia BSD
- ✓ Manejo del lenguaje SQL comprensivo.
- ✓ Realiza copias de seguridad en caliente (Online/hot Backups)
- ✓ Capacidad de realizar integridad referencial²².
- ✓ Desencadenantes
- ✓ Vistas

²²Integridad Referencial: En todo momento dichos datos sean correctos, sin repeticiones innecesarias.

PostgreSQL provee nativamente soporte para:

- ✓ Números de precisión arbitraria.
- ✓ Bloques de direcciones estilo CIDR²³.
- ✓ Direcciones MAC²⁴.

Algunas otras de sus principales características son:

Alta concurrencia

PostgreSQL permite que mientras un proceso escribe en una tabla, otros accedan a la misma tabla sin necesidad de bloqueos. Cada usuario obtiene una visión consistente de lo último a lo que se le hizo commit²⁵.

Instalación.

PostgreSQL permite instalar de manera infinita ya que no se viola ningún acuerdo de licencia por lo que no hay costo de este software libre.

Soporte

Por ser libre, existen desarrolladores y profesionales los cuales manejan y contribuyen.

Mantenimiento

PostgreSQL se mantiene y conserva mucho mejor que otras bases de datos, manteniendo el ahorro de los costos de operación, estabilidad, características y rendimiento de la mejor manera.

Prolongable

El código fuente como se menciona está libre, por lo que si una empresa necesitara cambiar el código para bien, se lo puede realizar sin costo.

²³ClasslessInter-DomainRoutingoCIDR(enespañol«enrutamientoentredominiossinclases»).

²⁴Es un identificador de 48 bits (6 bloques hexadecimales) que corresponde de forma única a una tarjeta o dispositivo de red.

²⁵Una sentencia COMMIT en SQL finaliza una transacción de base de datos dentro de un sistema gestor de base de datos relacional y pone visibles todos los cambios a otros usuarios.

Multiplataforma

PostgreSQL está disponible para diferentes plataformas (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.

Datos predefinidos

PostgreSQL agrega múltiples tipos de datos definidos por el estándar SQL3²⁶, entre otros.

Lenguajes

PostgreSQL brinda una interfaz con diversos tipos de lenguajes como: Python, PHP, C, C++, Delphi, Bash, Ruby, Tcl, ODBC, PHP, Lisp, Scheme y muchos otros.

PostgreSQL es completamente ACID

Es un conjunto de características que una base de datos debe cumplir para que se garantice que las transacciones son fiables, las siglas ACID representan atomicidad consistencia, aislamiento y durabilidad.

Transacción

Permite que las órdenes en formato SQL transfiera el paso de información en forma indivisible o atómica, el conjunto de acciones que debe formar una transacción son:

1. BEGIN TRAN: Indica que va a comenzar una transacción.
2. COMMIT TRAN: Indica que la transacción se puede considerar completa.
3. ROLLBACK TRAN: Indica que existió un fallo y que se debe regresar a un punto fiable.

²⁶SQL3: Es un estándar que soportar dos tipos o grupos de tipos de datos: tipos de datos predefinidos y tipos abstractos de datos.

Atomicidad

Garantiza que todas sus transacciones deben ser realizadas o caso contrario ninguna ejecutada.

Consistencia

Garantiza que todas las reglas que han sido declaradas deben cumplirse, si es que no es posible, la transacción será abortada.

Aislación

Garantiza que las transacciones que se estén realizando en ese momento, no afecten a las demás transacciones hasta que estas estén completadas.

Durabilidad

Garantiza en el que al instante de que se finaliza una transacción, permanezca fiable hasta que falle el software.

TABLA 5: Limitaciones técnicas de la base de datos PostgreSQL

Descripción	Valor
Máximo base de datos	Ilimitado
Máximo tamaño de tabla	32 TB
Máximo tamaño de registro	1.6 TB
Máximo tamaño de campo	1 GB
Máximo registro por tabla	Ilimitado
Máximo campos por tabla	250 a 1600
Máximo índices por tabla	Ilimitado
Numero de lenguajes para funciones	PGSQL, Java, Python, TCL, PHP, C, C++, Ruby, entre otros

Fuente: El autor

2.2.4 ARQUITECTURA DE POSTGRESQL.

El gráfico siguiente muestra el esquema de funcionamiento del Motor de PostgreSQL, se puede observar que la Arquitectura básica es la llamada “Usuario - Servidor”, para cada usuario que solicite una conexión, se crea un proceso llamado “postgres”, quien organiza la creación y eliminación de los procesos para cada usuario, es el servidor llamado “postmaster”, el cual atiende las peticiones de los usuarios en primera instancia.

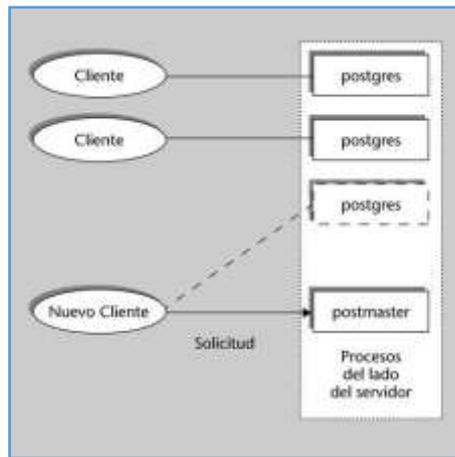


ILUSTRACIÓN 9: Esquema de funcionamiento PostgreSQL

Fuente: Bases de Datos en PostgreSQL, 2012, tomado de <http://ggomez.files.wordpress.com/2008/09/postgresql.pdf>

2.2.5 COMPONENTES DEL GESTOR DE LA BASE DE DATOS POSTRESQL

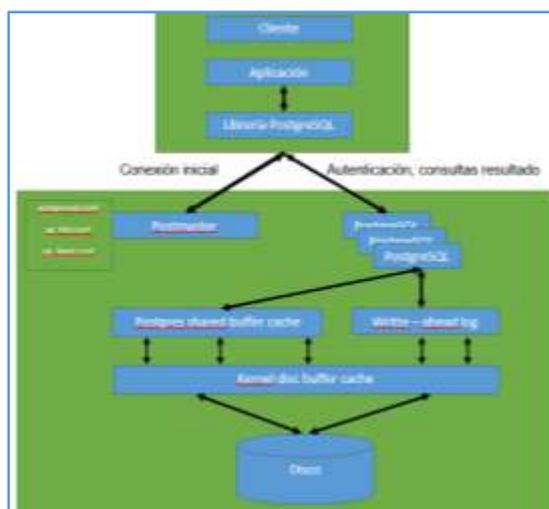


ILUSTRACIÓN 10: Esquema de Componentes de PostgreSQL

Fuente: El autor

Ciente: Crea una aplicación la cual va a necesitar de una base de datos en este caso de PostgreSQL, y sus librerías son útiles para hacer la conexión mediante sockets o a través de una conexión web usando TCP/IP.

Daemon Postmaster: Es la herramienta de PostgreSQL para escuchar sockets que están libres para los clientes, además se encarga de crear procesos hijos que son los tres ficheros principales que son postgresql.conf, pg_hba.conf y pg_ident.conf para que pueda autenticar peticiones, consultas y resultados de parte de los clientes.

Postgres shared buffer cache: es una memoria compartida por PostgreSQL para recopilar los datos que se almacenan en cache

Write – ahead log: este componente nos ayudara a asegurar la integridad de la información.

Kernel disc buffer cache: es el cache del disco que se almacena la información para que pueda funcionar el gestor de base de datos de PostgreSQL.

2.2.6 RECURSOS PRINCIPALES DE HARDWARE Y SOFTWARE.

Para hacer uso de este gestor de base de datos debemos utilizar los siguientes recursos físicos como lógicos.

Recursos Hardware

Con las siguientes características:

- ✓ Procesador: Intel(R) Core(TM) i5-337U CPU de 1.80GHz.
- ✓ Memoria RAM: 6 Giga Bytes.
- ✓ Disco duro de 1 Tera Byte.

Recursos Software

Con las siguientes características:

- ✓ Gestor de base de datos PostgreSQL.
- ✓ PostgreSQL GUI Tools.
- ✓ Sistema operativo Windows 8.1 de 64 bytes.

2.2.7 ESQUEMA DE BÚSQUEDA DE LAS BASES DE DATOS RELACIONALES

En el esquema de búsqueda de información, trabaja en forma organizada y la analiza adecuadamente, el esquema a continuación indica la forma de hacer una consulta.

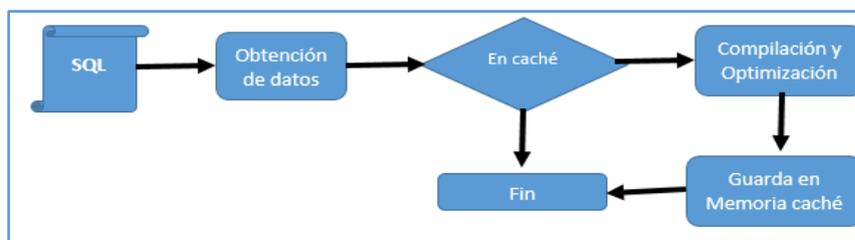


ILUSTRACIÓN 11: Esquema de búsqueda de PostgreSQL

Fuente: El autor

Como se puede observar en la ilustración de cómo hace un gestor de base de datos relacional en buscar y encontrar información que cuanta con los siguientes pasos:

SQL: se realiza una o varias consultas al gestor de base de datos en lenguaje Query analizando la expresión por si encuentra algún error.

Obtención de datos: una vez analizada la sentencia Query, continua con la forma de cómo se entregara la información a través de tablas, índices ordenadas adecuadamente.

Cache: permite buscar rápidamente la información si en algún momento se ha realizado anteriormente la misma consulta.

Compilación y optimización: se realiza el análisis de la sentencia y de cómo entregarla de manera rápida y segura la información de la sentencia que se está tratando de utilizar.

Memoria cache: se almacena la sentencia que realizamos para que en algún momento si es que necesitamos buscar la misma información, se pueda realizar de manera rápida y eficiente.

Fin: se realiza la sentencia SQL que indicamos al inicio.

2.2.8 HERRAMIENTAS DE DISEÑO DE POSTGRESQL

PgAdminIII

Se lo puede instalar bajo los diferentes sistemas operativos MS Windows, Mac OS X, Solaris, FreeBSD, GNU/Linux. Esta aplicación de código abierto, permite la gestión de bases de datos de PostgreSQL. Es una administración completa para Unicode²⁷, multihilos²⁸, herramientas para edición para todo tipo de datos, consulta rápida de la información. Se lo puede utilizar como un servidor remoto para otro sistema operativo.

PgAccess

Se lo puede instalar bajo los diferentes sistemas operativos MS Windows, Mac OS X, Solaris, FreeBSD, GNU/Linux, entre otras. Integra una interfaz gráfica para usuario de PostgreSQL y un navegador de Microsoft Access para la administración, edición manipulación, vistas y funciones de tablas de bases de datos e información.

PhpPgAdmin

Integra un navegador como servicio web para la administración de bases de datos, brinda edición, manipulación de todo tipo de objetos de una base de datos.

2.2.9 REQUISITOS DE INSTALACIÓN DE PGADMINIII

Software

- ✓ Para el uso del gestor de bases de datos PgAdminIII la cual vamos a hacer uso, es multiplataforma por lo cual se la puede instalar en Windows, Linux, FreeBSD, Solaris, así como en un sistema virtualizado
- ✓ En el sistema Linux es más práctico ya que se descarga automáticamente los archivos de manera rápida.
- ✓ Debemos tener en cuenta en las conexiones de red, debe ser estática.

²⁷Unicode es un estándar de codificación de caracteres diseñado para facilitar el tratamiento informático, transmisión y visualización de textos de múltiples lenguajes.

²⁸Multihilos contiene dos o más partes que se pueden ejecutar de manera concurrente o simultánea.

Hardware

- ✓ Se debe tener el disco duro en formato NTFS.
- ✓ Se necesita 8 Mega Bytes de memoria RAM.
- ✓ Un espacio requerido de 36 Mega Bytes en disco duro.

2.3 ANÁLISIS DE CARACTERÍSTICAS

Se debe conocer los requerimientos principales para que se logre escoger de manera adecuada la base de datos, los siguientes parámetros:

- ✓ El número máximo de datos y de tipo de datos ya sean textuales, binarios, imágenes u otro tipo.
- ✓ El número total de usuarios que va a hacer uso del sistema
- ✓ La disponibilidad que tiene la base de datos.
- ✓ La escalabilidad, crecimiento de información.
- ✓ La seguridad para el manejo de usuarios, roles.
- ✓ Administración de la base de datos dependiendo de cuan amigable sea el sistema informático.

2.3.1 TABLA COMPARATIVA

A continuación se presentara una matriz en la cual se mostrara las bases de datos relacionales principales donde se encontrara más información detallada.

TABLA 6: Comparativa de PostgreSQL

	Oracle	MySQL	SQL Server	PostgreSQL
Empresa	Oracle Corporation	Sun Microsystem	Microsoft	PostgreSQL Global Development Group
Licencia	Privada	Libre con condición de que si una empresa quiera agregarlo en productos deberá adquirir esta.	Privada	Libre
Ventajas	Oracle es más usado por las empresas a nivel mundial. Usar en cualquier plataforma	Puede ser ejecutado en cualquier maquina por su bajo consumo de memoria RAM. Tiene una conectividad muy segura.	Es muy fácil de instalar. Tiene conectividad a internet entre bases de datos.	Conexión segura entre máquinas. El código fuente es completamente disponible.
Desventajas	El precio es muy elevado para hacer la adquisición de este.	No es capaz de sincronizar los datos con otras bases de datos.	Consume demasiada memoria en la pc.	Tiempos de respuesta es lento.

Fuente: El autor

2.3.2 CRITERIOS DE EVALUACIÓN CONSIDERADOS

La sección en los criterios de evaluación de la base de datos PostgreSQL, depende de los resultados finales para la selección de la base de datos, que será utilizado en el desarrollo de la aplicación con la plataforma de Netbeans.

Por lo que se continuó con la búsqueda e investigación sobre las características de esta base de datos, para finalmente llegar a una conclusión de esta evaluación.

Los criterios de evaluación de la base de datos se resumen en la tabla siguiente:

TABLA 7: Criterios de Evaluación de PostgreSQL

Categorías	Criterios
Información general.	Toda la información de la base de datos PostgreSQL.
Soporte para sistemas operativos.	Cuales sistemas operativos se la puede instalar y manejar.
Características fundamentales.	Se tomara en cuenta las diferentes características principales.
Límites de la base de datos.	Capacidad de la base datos
Tablas y vistas.	Capacidades de manejar tipos de tablas.
Índices.	Capacidad de manejar tipos de índices.
Capacidades principales.	Información acerca del manejo de la información.
Tipos de datos.	Los diferentes tipos de datos que usa este gestor.

Fuente: El autor

TABLA 8: Información general de PostgreSQL

	Desarrollador	Fecha de lanzamiento	Última versión estable	Fecha última versión	Licencia
PostgreSQL	PostgreSQL Global Development Group	Marzo de 1998	9.3.4	Marzo 20 del 2014	PostgreSQL licencia comunidad

Fuente: El autor

TABLA 9: Soporte para Sistemas Operativos de PostgreSQL

	Windows	OS X	Linux	Unix	i OS	Android	OpenVMS
PostgreSQL	soporta	soporta	soporta	soporta	No soporta	soporta	No soporta

Fuente: El autor

TABLA 10: Características Fundamentales de PostgreSQL

Característica de PostgreSQL	
Objetos de gran tamaño de 64 bits	Sí
ALTER al objeto si existe	Sí
Cadena de replicación en cascada	Sí
CREATE TABLE tablas foráneas	Sí
Fondos editables	Sí
Fondo dinámicos de trabajo	Sí
Conjunto mejorado de funciones y operadores JSON	Sí
Vistas materializadas	Sí
Vistas materializadas con actualización simultánea	Sí
Slots de replicación	Sí
Índices para tipos de rango	Sí
Soporte anónimo de memoria compartida	Sí
Temporizadores	Sí
Vistas actualizables	Sí
Rendimiento	
Rendimiento Índice y mejoras de tamaño	Sí
Escaneo de índice	Sí
Escalabilidad multi-core para cargas de trabajo de sólo lectura	Sí
Niveles de bloqueo reducidos para los comandos ALTER TABLE	Sí
Red	Sí
Internacionalización	Sí
Tipo de datos JSONB	Sí
Tipo serial de tamaño pequeño	Sí
Funciones y disparadores	Sí
Disparadores de eventos	Sí
Cláusula FILTER para funciones agregadas	Sí

Fuente: El autor

TABLA 11: Límites de la Base de Datos de PostgreSQL

	PostgreSQL
Tamaño máximo de una base de datos	Ilimitadas
Tamaño máximo de un tabla	Hasta 32 Terabytes
Tamaño máximo de una fila	Hasta los 1.6 Terabytes
Columnas máximas por fila	Desde los 250 hasta las 1600 depende el tipo de dato
Tamaño máximo de un imagen Blob	1 GB de tipo texto o 4 TB almacenado en pg_largeobject
Tamaño máximo de tipo Char	Hasta 1 Gigabyte
Tamaño máximo de tipo numeral	Ilimitado
Valor mínimo de tipo fecha	Desde -4,713
Valor máximo de tipo fecha	Hasta los 5,874,897
Tamaño máximo de columnas	Hasta las 63 columnas

Fuente: El autor

TABLA 12: Tablas y Vistas de PostgreSQL

	Tablas temporales	Vistas Materializadas
PostgreSQL	Si permite	Si permite

Fuente: El autor

Las vistas materializadas dependen de la versión en que se las quiera usar desde la versión 9.3 en adelante.

TABLA 13: Índices de PostgreSQL

	Hash	Expresión	Parcial	Bitmap	Texto
PostgreSQL	Si permite				

Fuente: El autor

TABLA 14: Capacidades Principales de PostgreSQL

	PostgreSQL
Unión	Si permite
Excepciones	Si permite
Junturas internas, externas	Si permite
Unión de junturas	Si permite
Funciones de ventanas	Si permite
Sentencia paralela	Si permite

Fuente: El autor

TABLA 15: Tipos de Datos de PostgreSQL

	PostgreSQL
Tipo de sistema	Estático
Entero (Integer)	Smallint 16 bits (entero pequeño), integer 32 bits (entero), bigint 64 bits (entero grande)
Punto flotante (Float)	Real 32 bits, double (doble) 64 bits
Decimal	Decimal, numeric (numérico)
Cadena (String)	Char (texto pequeño), varchar (texto), text (texto)
Binario (Binary)	Bytea (tipo bite)
Tipo fecha (Date/Time)	Date (fecha), time (tiempo sin zona horaria), timestamp, interval
Booleano (Boolean)	Verdadero o falso

Fuente: El autor

TABLA 16: Otros Objetos de PostgreSQL

	PostgreSQL
Cursores	Si permite
Desencadenadores (Trigger)	Si permite
Funciones (Function)	Si permite
Procedimientos almacenados (Procedure)	Si permite
Rutinas externas (External Routine)	Si permite

Fuente: El autor

2.3.3 VENTAJAS DE POSTGRESQL

Un sistema estable de almacenamiento, control, administración como lo es PgAdminIII, que se caracteriza principalmente que la mayoría de gestores de bases de datos ya que se lo puede instalar y configurar en sistemas operativos tanto libres como privativos. Permite la administración, componer nuevos sistemas y aplicaciones con el buen manejo de este gestor.

Este gestor de bases de datos permite migrar y desarrollar aplicaciones desde Visual Basic, Access, Visual C/C++, entre otras. Es una gran alternativa de uso, este gestor es un sistema estable, con gran flexibilidad. PostgreSQL está agregando características acerca de las bases de datos NoSQL, como lo es el tipo de archivo JSON ²⁹ esta característica se la está añadiendo por el uso paulatino por parte de MongoDB.

2.4 LAS BASES DE DATOS NOSQL

2.4.1 DEFINICIÓN

Las bases de datos relacionales SQL han sido utilizables desde el año 40 como ORACLE, MYSQL, SQL Server, POSTGRESQL, DB2, INFORMIX, entre otras. Con una simple meta que es la solución de problemas para el almacenamiento de información en el mundo.

Las bases de datos NoSQL se crearon para resolver problemas de gestión y estructuración de datos de las empresas u organizaciones. Las bases de datos NoSQL son bases de datos que no se ligan a sus características y modelo de bases de datos relacionales.

Son sistemas gestores de bases de datos que reemplaza el modelo clásico de sistema de gestión de bases de datos donde se consigue almacenar grandes cantidades de información.

Estas bases de datos no relacionales no poseen esquemas, joins ³⁰ y tampoco usan SQL además no se garantiza ACID, no escalan verticalmente como lo hace

²⁹JSON: JavaScript Object Notation, es un formato ligero para el intercambio de datos.

³⁰Joins: Es una sentencia que permite combinar registros de datos en uno solo para presentarlos.

SQL; resuelven diferentes tipos de problemas para grandes cantidades de datos, además de su considerable capacidad de realizar transacciones y consultas de manera diaria.

De manera que en esta época se la está utilizando como una alternativa más fiable para el almacenamiento y estructuración de nuevas bases de datos, ya que estas bases de datos no relacionales pueden crecer de manera escalable procesando información de distintos tipos de datos, y pueden ser procesados instantáneamente.

La diferencia primordial entre las bases de datos SQL relacionales con las bases de datos NoSQL no relacionales es la cualidad de cómo almacenar los datos:

- ✓ En las bases de datos SQL se debe crear tablas y luego transformarlo a objeto de la vida real.
- ✓ En las bases de datos NoSQL es libre de esquemas por lo que se almacena la información eficazmente, no se crea tablas ni estructura, a menos que los datos sean relacionales.
- ✓ Las bases de datos SQL ofrecen poco rendimiento en aplicaciones como indexación de documentos, páginas web con alto tráfico web, datos; caso contrario con las bases de datos NoSQL.
- ✓ Las bases de datos SQL están estrechamente ligadas a pequeñas transacciones de información o a largas transacciones con poco acceso de escritura.
- ✓ Las bases de datos brindan servicio a grandes cantidades de escritura y lectura.

2.4.2 ESTRUCTURA DE LAS BASES DE DATOS NOSQL

La estructura nos indica que en la parte de hardware, lugar donde se almacenara la información ya sean discos, tambores, cintas magnéticas, entre otras.

Se encuentra también la parte software o capa de administración de bases de datos donde se administrara y gestionara las bases de datos NoSQL.

Por último se encuentra la interfaz de usuario donde se mostrara la información detallada, ya sea una aplicación web, móvil u otros sistemas.

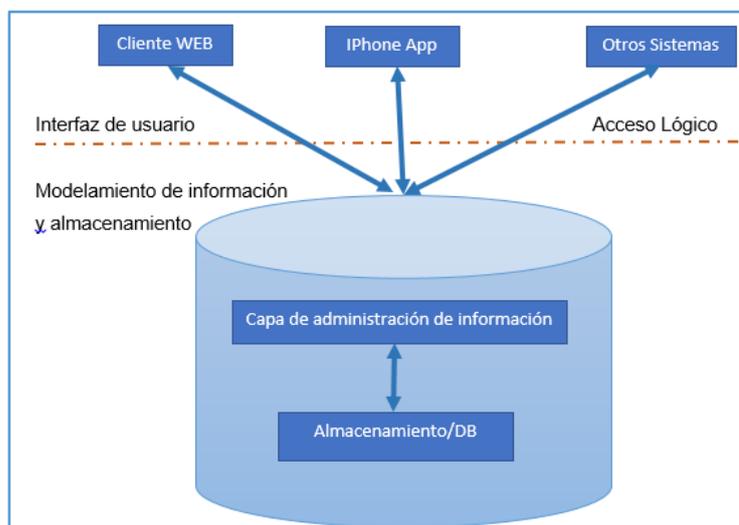


ILUSTRACIÓN 12: Estructura de las bases de datos noSQL

Fuente: El autor

Una ventaja importante de las bases de las bases de datos NoSQL, es la de brindar estructuras jerárquicas en entidades, estas estructuras son fáciles de escribir en lenguaje JSON de manera que simplifica en gran medida el espacio de información, por lo que puede almacenar grandes cantidades de información.

Muchas aplicaciones NoSQL hacen uso de la arquitectura distribuida, conservando la información redundante para almacenarlos en diferentes servidores por lo que si un servidor se inutilizase, otro lo reemplazara.

2.4.3 ESQUEMAS Y VALIDACIÓN

Las bases de datos NoSQL son llamadas bases de datos sin esquema ya que es una característica primordial de estas bases de datos, esa es su ventaja principal, quiere decir que, no se necesitaría reestructurar por lo que la actualización de estas bases de datos seria de manera rápida y flexible al igual que su almacenamiento de información.

En su ausencia de no tener que realizar esquemas, vendrían inconvenientes como las restricciones de validez de la información así como su integridad. De alguna manera si desearíamos que se solucionara este problema, nosotros podemos indicar nuestras propias restricciones y hacer cumplir las reglas de bases de datos relacionales a través de un lenguaje de programación.

2.4.4 TEOREMA CAP

El teorema CAP o teorema de Brewer, nos indica que un sistema distribuido no puede satisfacer a la vez consistencia, disponibilidad y tolerancia a particiones.

Consistencia: indica que todos los usuarios en ese momento puedan observar la misma información al momento de realizar una consulta.

Disponibilidad: Indica que el sistema debe seguir brindando servicio a pesar de que un nodo de servidor fallara a los usuarios.

Tolerancia a partición: Indica que el sistema debe seguir brindando el servicio independientemente del lugar (geográfica) donde están consultando. Las bases de datos

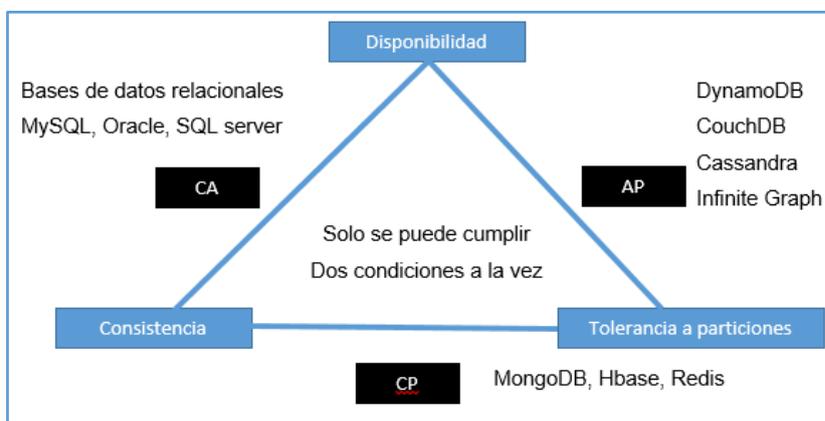


ILUSTRACIÓN 13: Teorema CAP

Fuente: El autor

NoSQL dependen de este teorema para ser escalables y distribuidas, dependiendo de las condiciones que tiene el teorema se dividen las bases de datos:

AP: indica disponibilidad y tolerancia a particiones, a excepción de la consistencia en su totalidad.

CP: Indica consistencia y tolerancia a particiones, a excepción de la disponibilidad.

CA: Indica consistencia y disponibilidad, a excepción de tolerancia a particiones que de alguna manera se lo puede solucionar replicando datos

Se debe considerar a MongoDB como CP pero a través de la configuración de agregación de nodos podemos replicar datos, de alguna manera si sacrificamos la consistencia podemos ganar disponibilidad.

2.4.5 CARACTERÍSTICAS DE LAS BASES DE DATOS NOSQL

El fin con el cual se desarrollaron las bases de datos NoSQL para satisfacer las necesidades de la gran cantidad de información o (Big Data) de manera eficaz.

✓ Tipos de lenguaje diferente

La mayoría de gestores de bases de datos NoSQL utilizan diferente tipo de lenguaje que el estándar o SQL sencillamente lo usan como apoyo. Estas bases de datos utilizan su propio lenguaje como MongoDB usa JSON, Cassandra utiliza CQL, entre otros.

✓ Estructuras variables

No poseen una estructura fija, sino que crean una propia dependiendo su utilidad ya sean en clave – valor, objetos documentos, grafos entre otros.

✓ No garantiza ACID

La propiedad ACID (Atomicidad, consistencia, aislamiento y durabilidad) no contemplan estas bases de datos NoSQL ya que causaría que el rendimiento de estas decaería. A cambio de esta propiedad se usa la llamada consistencia eventual o BASE, esta propiedad brinda la llamada consistencia en la que las transacciones deben cumplirse solo cuando no se haya modificado durante un largo tiempo.

✓ Operaciones Join no usadas

No se usa la operación Join puesto que las bases de datos son inmensamente extensas, por lo que puede dar como resultado una sobrecarga de

transacciones, disminuyendo la eficacia del gestor de base de datos y aplicación. La mejor manera de solucionar esto es a través de la des normalización de la base de datos, si deseamos realizar un Join pues debemos hacerlo en la aplicación.

✓ **Su arquitectura distribuida**

En las bases de datos NoSQL se usan fácilmente las tablas hash para compartir la información por lo que el procesamiento de estos datos es más rápido y eficaz.

✓ **Escalabilidad Horizontal**

La escalabilidad horizontal brinda un beneficio para las bases de datos NoSQL ya que permite la disminución de tráfico en las transacciones y búsquedas de la información, ayudando a Pc's con bajos recursos de memoria.

2.4.6 TIPOS DE MODELOS DE BASES DE DATOS NOSQL

Hasta el día de hoy existen 150 tipos de gestores de bases de datos NoSQL, dependiendo de las características que tengan estos gestores se podrá seleccionar al que mejor se acomode a la empresa. Los principales tipos bases de datos NoSQL son:

Bases de Datos Clave – Valor

Es un modelo sencillo donde se crea una clave o llave única para cada elemento por lo que la velocidad de búsqueda, lectura y escritura de datos es elevada, por ejemplo de las bases de datos de tipo clave valor son: DynamoDB, Redis.

Ejemplo del formato de lenguaje y estructuración:

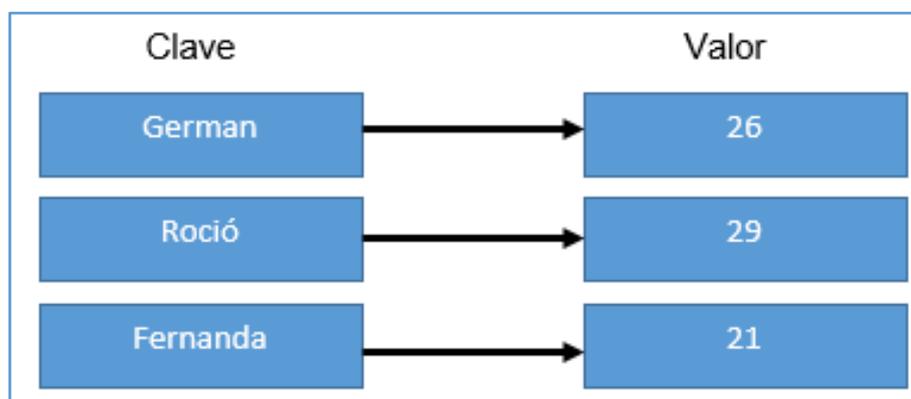


ILUSTRACIÓN 14: Ejemplo Clave/Valor

Fuente: El autor

Bases de Datos Orientadas a Documentos

Es el modelo más popular del mercado, gestiona datos semi estructurado, con un formato básico de almacenamiento como lo es XML, JSON y BSON de manera binaria donde hay una clave única para cada registro. Puede ejecutar consultas de clave - valor eficazmente. Se lo compara con las bases de datos relacionales por su versatilidad.

Ejemplo de las bases de datos de tipo Documentos son: MongoDB o CouchDB.

Características de la Base de Datos Orientadas a Documentos

- ✓ Modelado de datos.
- ✓ Apegadas al desarrollador.
- ✓ Desarrollo rápido.
- ✓ Orientas a la web: CRUD.

Ejemplo del formato de lenguaje y estructuración:

```
{
  Name: "German",
  Tipo: "Entero",
  Tipos:
  [
    {
      Título: "Estudiante",
      Artículos: 2
    },
    {
      Título: "Egresado",
      Artículos: 1
    }
  ]
}
```

Bases de Datos Orientadas a Grafos

Las bases de datos orientadas a grafos no son más que representaciones de información almacenada utilizando nodos, se hace uso de la teoría de grafos para comprenderla. Su estructura debe estar normalizada para realizar una búsqueda de información eficaz.

Características de la Base de Datos Orientadas a Grafos

- ✓ Infundidas por Euler y la teoría de grafos.
- ✓ Modelo de datos: nodos, relaciones con pares clave valor en ambos.

Bases de datos Orientadas a Objetos

La información que contiene esta base de datos se representa de la misma manera que en lenguajes de programación orientada a objetos como lo es con JAVA o C#; a través de relaciones jerárquicas usando un identificador único que brinda la base de datos. Ejemplo de las bases de datos de tipo Orientadas a Objetos son: Zope, Gemstone o Db4o.

Bases de datos Orientadas a Columnas

Esta base de datos NoSQL almacena la información en columnas con una apariencia a bases de datos de clave - valor, esta base de datos nos brinda velocidad de lectura de los datos pero decae en la escritura de estos, se lo usa en data warehouses ³¹y en business intelligence³². Ejemplo de esta base de datos son: HBase, Hypertable, Cassandra, Riak.

Características de la Base de Datos Orientadas a Columnas

- ✓ Gestión de tamaño.
- ✓ Alta disponibilidad.

Ejemplo del formato de Columnas:

³¹Warehouses: Los almacenes de datos almacenan los datos actuales e históricos y se utilizan para la creación de informes y presentación.

³²Business Intelligence: Conjunto de estrategias y aspectos relevantes enfocados a la administración y creación de conocimiento.

TABLA 17: Ejemplo Columnas

ID_AUTOMOVIL	NOMBRE AUTOMÓVIL	PLACA
10044921	Corsa	CHG-394
29993221	Vitara	BWQ-294

Fuente: El autor

2.4.7 VENTAJAS DE LAS BASES DE DATOS NOSQL

Se los puede instalar y manipular desde máquinas con recursos limitados, las bases de datos NoSQL no requieren mucha capacidad para poder ser utilizados.

Tiene escalabilidad horizontal añadiendo más nodos, con esto puede haber más recursos disponibles para la utilización y manejo.

Tiene una gran capacidad de almacenar información, estas bases de datos manejan una estructura distribucional, por lo que puede manipular cientos de miles de datos, otra manera es usando tablas hash.

El problema principal de las bases de datos que manejan SQL es que al momento de ejecutar una sentencia esta se la tiene que transcribir de manera que se complica cada vez que se la sigue utilizando; por lo que se puede ralentizar el momento de realizar estas sentencias en este tipo de bases de datos SQL, caso contrario en las bases de datos NoSQL no sucede esto.

2.4.8 DIFERENCIAS DE LAS BASES DE DATOS NOSQL

No se usa lenguaje SQL para realizar las consultas, rara vez se las utiliza como un apoyo en las consultas, en este caso de las bases de datos NoSQL se utiliza un lenguaje como es el caso de Cassandra que usa CQL o como es el caso de MongoDB que usa JSON.

Las bases de datos NoSQL permiten el almacenamiento de mucha información por lo que es mejor evitar hacer el uso de joins, ya que si es que se desea hacer una operación de un campo, la sobrecarga es costosa al momento de medir tiempos, por lo que de alguna manera se debe des normalizar los datos.

De alguna manera las bases de datos NoSQL no utiliza estructuras fijas para el almacenamiento de información, utiliza otro tipo de patrón para el almacenamiento como es la de grafos, objetos, tipo – valor.

Las bases de datos NoSQL se centran en tratar de compartir la información en varios equipos a través de uso de las tablas hash distribuidas, en cambio las bases de datos SQL se centran en un solo equipo como es el caso de la estructura maestro - esclavo.

2.5 MONGODB



Fuente: Reinosa, Enrique José, Maldonado, Calixto Alejandro, “Diseño de Bases de Datos” Alfa Omega 2012.

2.5.1 DEFINICIÓN

MongoDB es una base de base de datos multiplataforma libre que es orientada a documentos quiere decir que se agregan los datos o información que deseemos en documentos en vez de que se almacene en registros como es el caso de las bases de datos SQL, con un conjunto de documentos se forma una colección de documentos que en este caso serían las tablas lo cual nos brinda un valioso rendimiento, disponibilidad y escalabilidad.

Además, podemos arbitrariamente generar nuestro propio esquema y ser completamente diferente con otros esquemas, no es necesario seguir con un esquema, si se desea almacenar información semi estructurada pues se debería utilizar MongoDB.

Necesariamente si es que se desea escalabilidad, replicación y sharding³³, en MongoDB, son muy fáciles de manejar sin problemas. Se la puede descargar y usar de manera gratuita bajo la licencia AGPL³⁴ de Apache, si se la quiere comercializar, 10gen³⁵ brinda una licencia comercial.

³³Sharding es un proceso por el cual se pueden almacenar información en múltiples máquinas.

³⁴AGPL Es una licencia Free Software Foundation distribución de software libre en la cual indica que si se modifica una parte del código fuente se la debe compartir con el mundo.

³⁵10gen es una empresa que desarrolla aplicaciones, con licencias comerciales.

2.5.2 HISTORIA

MongoDB es una palabra en inglés que se denomina “homongous” que significa enorme o gigante, se desarrolló bajo la empresa de software 10gen en el año de 2007 que son fundadores directos de Doubleclick³⁶, Shop Wiki, GILT Groupe, primero como una aplicación como servicio PAAS (plataforma como servicio) como es el caso de Google App Engine. En la actualidad, MongoDB es una base de datos para su uso y producción que puede brindar varias características de una base de datos de nivel mundial, con una licencia de uso AGPL escrito en C++.

Esta base de datos es utilizada por MTV Network, Craigslist, Fousquare y empresas que desarrollan juegos, fue desarrollado para ser un motor de búsqueda en aplicaciones en la nube por eso fue el primer desarrollo en SAAS (software como servicio).

La GUI (graphical user interface) por parte de los desarrolladores, no se la ha decidió implementar para MongoDB ya que creadores en este caso 10gen a través de experiencias de otros usuarios que la usan fueron variados, dado que continuamente se han creado varias aplicaciones de MongoDB, los usuarios pueden fácilmente hacer uso de esto aunque puede que sean utilizando comandos.

TABLA 18: Versiones de MongoDB

Versión	Año de lanzamiento
Versión 0.8	Primera versión oficial febrero 2009
Versión 1.0	Agosto 2009
Versión 2.0	Septiembre 2011
Versión 2.2	Agosto 2012
Versión 2.4.3	Abril 2013

Fuente: Propia

³⁶Doubleclick plataforma tecnológica de anuncios para administrar publicidad digital.

2.5.3 CARACTERÍSTICAS DE MONGODB

La base de datos MongoDB está completamente escrita en C++, la principal ventaja de esta base de datos noSQL es la velocidad de consulta, el motivo de esto es porque la información se la almacena en archivos de formato BSON que son versiones modificadas de JSON, con este formato se puede lograr flexibilidad para datos, mejor búsqueda y localización de información, conversiones de objetos en una aplicación a documento JSON.

Esta base de datos contiene para su uso en desarrollo de aplicaciones, índices secundarios, consultas dinámicas, orden en los resultados que se obtiene, agregaciones, almacenamiento de información ilimitada.

Las características técnicas que posee la base de datos NoSQL MongoDB hacen que trabaje con grandes cantidades de información y asimismo de usuarios conectados indefinidamente en la aplicación, las características principales son:

✓ **Escalabilidad horizontal**

MongoDB está diseñado para escalar de manera ilimitada a lo que otras bases de datos no podrían, a través de su replicación y su sharding, se puede almacenar información de varios equipos conectados entre sí.

✓ **Consultas Ad hoc**

Permite la consulta de información a través de búsquedas de campos, expresiones regulares con comandos de manera rápida y eficaz.

✓ **Indexación**

MongoDB utiliza indexación para aumentar la eficiencia en la búsqueda de información.

✓ **Replicación**

MongoDB puede realizar replicación Master – Slave.

Master (maestro): ejecuta comandos para la lectura y escritura de los datos.

Slave (esclavo): realiza solo lectura de datos pero no los puede modificar, si es que por algún caso no pueda acceder al maestro que se le designo, este esclavo puede elegir otro maestro para seguir trabajando.

✓ **Blanco de carga**

MongoDB permite que pueda ejecutarse en distintos servidores a la vez, haciendo más fácil su distribución en la carga de usuarios que deseen conectarse así como acceder a la información.

✓ **Almacenamiento de archivos**

MongoDB puede ser usado como un gestor de archivos e información haciendo que esta base de datos tenga una gran capacidad de almacenar información a través de sus diferentes servidores activos.

Usos en MongoDB

MongoDB se lo puede implementar en:

- ✓ Sistemas de manejo de documentos
- ✓ Juegos
- ✓ Aplicaciones móviles
- ✓ Registro de eventos
- ✓ Almacenamiento de información en páginas web como locales
- ✓ Administración de estadísticas

Especificaciones de MongoDB

- ✓ Escrito en C++ como primer lenguaje para su creación, se crearon drivers para su manejo más óptimo.
- ✓ Mantiene alguna de las propiedades del lenguaje SQL.

- ✓ Licencia AGPL apache.
- ✓ Sharding o replicación maestro – esclavo.
- ✓ Utiliza sentencias en JavaScript.
- ✓ Realiza un mapeo de los archivos para mejor almacenamiento de la información.
- ✓ En sistemas de 32 bits se limita su capacidad a 2.5 Giga Bytes.
- ✓ Se agrega el campo de indexación geo espacial.
- ✓ Protocolo que trabaja sobre TCP/IP.
- ✓ Orientada a documentos almacenados en archivos BSON.
- ✓ La base de datos contiene específicamente colecciones que a su vez contiene documentos.

TABLA 19: Limitaciones técnicas de la base de datos MongoDB

Descripción	Valor
Máximo base de datos	Ilimitado
Máximo tamaño de Nodos	3 TB
Máximo cantidad de Nodos	Ilimitado
Máximo tamaño de documentos por nodo	12 millones de documentos
Sistemas operativos	Unix, Windows, Mac OS, Linux
Numero de lenguajes para funciones	C, C++, Java, JavaScript, PHP, Phyton, Ruby, .Net.

Fuente: El autor

2.5.4 ARQUITECTURA DE MONGODB

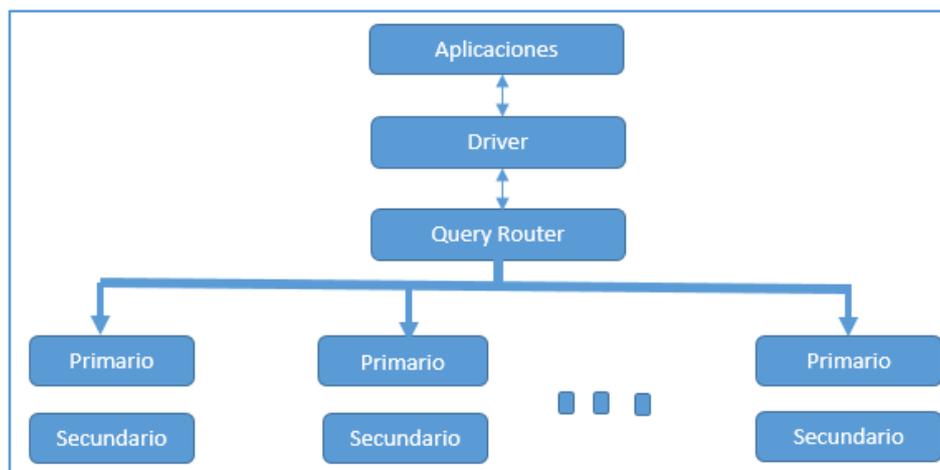


ILUSTRACIÓN 15: Arquitectura de base de datos MongoDB

Fuente: El autor

Se tiene una aplicación que se ha desarrollado en un gestor de aplicaciones ya sea libre o privativo con la base de datos MongoDB, esta base de datos provee diferentes tipos drivers de lenguaje para poder programar de manera rápida y eficaz incluyendo Net, Java, Ruby, PHP, JavaScript, Python entre otros.

El Shell interactivo que tiene MongoDB es rico en áreas que permite la interacción de la información de la base de datos con las consultas ad hoc que mencionamos anteriormente para su búsqueda eficaz.

Query Router o enrutador de consultas envía las consultas que se realizaron al momento de gestionar la información que desea un usuario y se las envía a los diferentes almacenadores de datos realizando así el sharding de la información, esta información se la envía a los servidores de almacenamiento de MongoDB primario y luego se los replica en las demás servidores.

2.5.5 COMO FUNCIONA MONGODB

El gestor de base de datos MongoDB contiene una consola que está desarrollada en JavaScript, por ese motivo se utiliza ese lenguaje para realizar nuestra gestión de base de datos no relacional para luego ponerla en marcha en algún gestor de aplicaciones, mencionada anteriormente que con el uso de drivers se puede realizar una conexión segura con los gestores de programas.

Ejemplo de un documento de MongoDB

```
{
  "_id": ObjectId ("4efa8d2b7d284dad101e4bc7"),
  "Nombre": "German",
  "apellido": "Correa",
  "Edad": 25,
  "Dirección": {
    "Calle": "Avn. Cristoval de Troya",
    "Ciudad": "Ibarra"
  }
}
```

Comandos que se utilizan frecuentemente en MongoDB

Ejemplos:

Se desea crear una colección de datos

```
db.createcollection("empleados")
```

Si se desea ingresar un nuevo doctor a la tabla empleados

```
db.empleados.insert({userid:"1234",nombre: "luis",apellido: "correa", edad: 25,})
```

Si se desea buscar todos los doctores de la tabla empleados en MongoDB se lo haría de la siguiente manera

```
SELECT * FROM empleados WHERE name="doctor"
```

```
db.empleados.find({name:"doctor"}).forEach(printjson);
```

Si se desea recuperar el primer elemento que cumple alguna restricción:

```
printjson (db.empleados.findOne({name:"doctor"}));
```

2.5.6 RECURSOS PRINCIPALES DE HARDWARE Y SOFTWARE.

Para hacer uso de este gestor de base de datos debemos utilizar los siguientes recursos físicos como lógicos.

Recursos Hardware

Con las siguientes características:

- ✓ Procesador: Intel(R) Core(TM) i5-337U CPU de 1.80GHz.
- ✓ Memoria RAM: 6 Giga Bytes.
- ✓ Disco duro de 1 Tera Byte.

Recursos Software

Con las siguientes características:

- ✓ Gestor de base de datos MongoDB.
- ✓ Sistema operativo Windows 8.1 de 64 bytes.

2.5.7 ESQUEMA DE BÚSQUEDA DE LAS BASES DE DATOS RELACIONALES

En el esquema de búsqueda de información, trabaja usando nodos, usando un proceso llamado Map/Reduce.

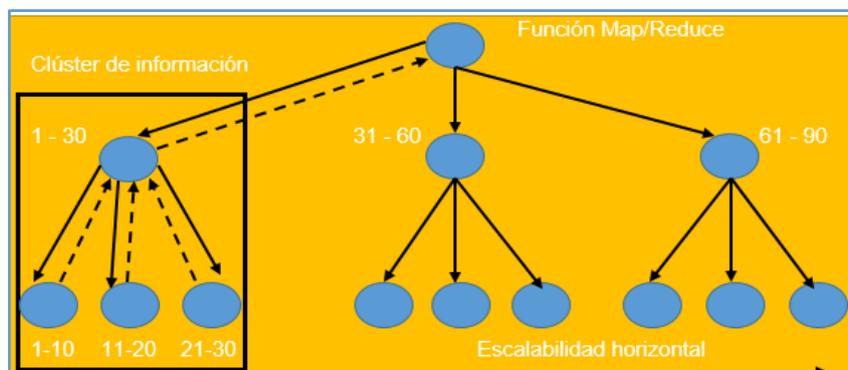


ILUSTRACIÓN 16: Esquema de búsqueda de PostgreSQL

Fuente: El autor

Como se observa en la ilustración del tipo de mapeo que se está realizando en las bases de datos NoSQL, indica que busca la información a través de una secuencia de nodos y usando el proceso llamado reducción.

Primero: el usuario indica que necesita información a la base de datos NoSQL.

Segundo: La base de datos NoSQL analiza el rango de claves hacia los nodos hijos, para encontrar la información que el usuario buscaba.

Tercero: Se realiza la búsqueda a cada nodo de la base de datos y sus hijos.

Cuarto: En el caso de que encuentra la información que se buscaba, se realiza el Reduce que devuelve la información de nodo en nodo hasta llegar al principal y entregarla al sumario.

2.5.8 HERRAMIENTAS DE DISEÑO DE MONGODB

Mongo: Shell interactivo que permite a los desarrolladores:

- ✓ Ver, insertar, eliminar y actualizar datos en su base de datos.
- ✓ Replicar información, configurar los Shards (fragmentación), apagar los servidores y ejecutar JavaScript. Link <https://www.mongodb.org/>

RockMongo: Permite un modo grafico web para la gestión de las bases de datos así como la vista de las colecciones y sus datos. Link <http://sourceforge.net/projects/rockmongoadmin/>

Mongolab: herramienta para la administración de bases de datos de MongoDB en la nube, brinda seguridad, almacenamiento, rendimiento, apoyo entre otros. Link <https://mongolab.com/>

2.5.9 REQUISITOS DE INSTALACIÓN PARA MONGODB

Software

- ✓ Para el uso del gestor de bases de datos MongoDB el cual vamos a hacer uso, es multiplataforma por lo cual se la puede instalar en Windows, Linux, FreeBSD, Solaris, así como en un sistema virtualizado.

- ✓ En el sistema Linux es más práctico ya que se descarga automáticamente los archivos de manera rápida.

Hardware

- ✓ Equipo de cómputo con procesador Pentium 4 en adelante.
- ✓ Memoria Cache de 2 Mega Bytes.
- ✓ Memoria RAM desde los 2 Giga Bytes en adelante.
- ✓ Almacenamiento en disco Duro desde los 120 Giga Bytes en adelante.

2.6 ANÁLISIS DE CARACTERÍSTICAS

Se debe conocer los requerimientos principales para que se logre escoger de manera adecuada la base de datos en la cual vamos a hacer uso para el aplicativo.

- ✓ El número máximo de datos y de tipo de datos ya sean textuales, binarios, imágenes u otro tipo.
- ✓ El número total de usuarios que va a hacer uso del sistema.
- ✓ La disponibilidad que tiene la base de datos.
- ✓ La escalabilidad, crecimiento de información.
- ✓ La seguridad para el manejo de usuarios, roles.
- ✓ Administración de la base de datos dependiendo de cuan amigable sea el sistema informático.

2.6.1 TABLA COMPARATIVA

A continuación se presentara una matriz en la cual se mostrara las bases de datos relacionales principales donde se encontrara más información detallada.

TABLA 20: Comparativa MongoDB

	Cassandra	CouchDB	MongoDB	Redis
Empresa	Apache software foundation.	Apache software foundation.	10gen	Slavatore Sanfilippo
Licencia	Licencia de Apache 2.0	Licencia de Apache 2.0	GNU AGPL V3.0 (licencia de apache)	BSD
Ventaja	Desarrollado como un servidor distribuido. Escalable horizontalmente.	Almacena documentos en formato JSON. Puede ser usada por varios usuarios master.	Almacena la información en documentos BSON. MongoDB está libre de esquemas, no se diseña tablas por adelantado.	Responde varias peticiones de consulta. Fácil de ajustar a servidores web.
Desventaja	No hace uso de joins.	No soporta joins. Almacena documentos en formato JSON.	La información puede perderse si no se la trata adecuadamente.	Almacena datos clave-valor.

Fuente: El autor

2.6.2 CRITERIOS DE EVALUACIÓN CONSIDERADOS

La sección en los criterios de evaluación de la base de datos MongoDB, depende de los resultados finales para la selección de la base de datos, que será utilizado en el desarrollo de la aplicación con la plataforma de Netbeans.

Por lo que se continuó con la búsqueda e investigación sobre las características de esta base de datos, para finalmente llegar a una conclusión de esta evaluación; los criterios de evaluación de la base de datos se resumen en la tabla siguiente:

TABLA 21: Criterios de Evaluación de MongoDB

Categorías	Criterios
Información general.	Toda la información de la base de datos MongoDB.
Soporte para sistemas operativos.	Cuales sistemas operativos se la puede instalar y manejar.
Características fundamentales.	Se tomara en cuenta las diferentes características principales.
Límites de la base de datos.	Capacidad de la base datos
Capacidades principales.	Información acerca del manejo de la información.
Tipos de datos.	Los diferentes tipos de datos que usa este gestor.

Fuente: El autor

TABLA 22: Información general MongoDB

	Desarrollador	Fecha de lanzamiento	Última versión estable	Fecha última versión	Licencia
MongoDB	MongoDB inc.	Febrero del 2009	2.6.4	Agosto 11 del 2014	GNU AGPL v3.0 (Licencia Apache)

Fuente: El autor

TABLA 23: Soporte para Sistemas Operativos MongoDB

	Windows	OS X	Linux	Unix	i OS	Android	OpenVMS
MongoDB	soporta	soporta	soporta	soporta	soporta	No soporta	No soporta

Fuente: El autor

TABLA 24: Características Fundamentales MongoDB

Características de MongoDB	
Rápido y fácil de instalar	Si
Idioma de internacionalización	Si (i18n español , francés, inglés, japonés, chino, alemán)
Protección a través de contraseña	Si
Servidor para información	PHP, servidor web, replicación, estado
Comandos para Consultas	CRUD en código de JavaScript
Estadísticas, perfiles, gestión de usuarios	Si
Transferencia de la información	Exportación e importación
Replicación en los datos	En 3 nodos (1 servidor para backup y 2 nodos en datacenter)
Particionamiento de datos	Si
Capacidad para recuperación de datos	Si
Soporta almacenamiento remoto	No

Fuente: El autor

TABLA 25: Límites de la Base de Datos MongoDB

	MongoDB
Tamaño máximo de documento	16 Mega Bytes
Nivel máximo de anidación de documento.	100 documentos dentro de documentos.
El namespace (espacio de nombres).	123 caracteres (name_bd + collection_name).
Nombre de la base de datos (name_bd).	Limita a 64 caracteres.
Archivo por defecto ns.	Almacena 24.000 namespaces (name_bd como collection)
Indexación de campos.	1024 bytes.
Nivel máximo de indexación	64 índices por collection.

Búsqueda de texto completo y geo referenciarían	No se las puede realizar en una misma consulta.
En Linux se puede almacenar	Un máximo de 64 Tera Bytes.
En Windows se puede almacenar	Un máximo de 4 Tera Bytes.
Número máximo de clave	512 bytes
Nombres en bases de datos son sensibles	En mayúsculas y minúsculas
Caracteres prohibidos en nombres de bases de datos	Linux: “/.” Windows: “/ . * <> : ”
Versiones de software de 32 bits	Almacenan hasta 2 Giga Bytes
Versiones de software de 64 bits	Sin limitación de almacenamiento
Tipo de procesadores	X86 x86_64
Mejoramiento con la RAM	Si permite
Numero de lenguajes que soporta	C, C++, Java, JavaScript, PHP, Python, Ruby, .Net.

Fuente: El autor

TABLA 26: Capacidades Principales MongoDB

	MongoDB
Índices Geoespaciales	Si permite
Análisis de Queries	Si permite
In-Place (almacenamiento en 2 fases)	Si permite
GridFS (almacenamiento meta data)	Si permite
Replicación (maestro esclavo)	Si permite
Auto sharding (seguridad en envío)	Si permite
Map/Reduce (consulta de data)	Si permite

Fuente: El autor

TABLA 27: Tipos de Datos MongoDB

	MongoDB
Tipo de sistema	Estático
Cadena de caracteres	Sin especificar
Entero (Integer)	Smallint 16 bits (entero pequeño), integer 32 bits (entero), bigint 64 bits (entero grande).
Double (doble)	Real 32 bits, double (doble) 64 bits.
Booleano (Boolean)	Verdadero o falso.
Tipo fecha (Date/Time)	Date (fecha), time (tiempo sin zona horaria), timestamp, intervalos.
Null (nulos)	Sin especificar.
Array (arreglos)	Char (texto pequeño), varchar (texto), text (texto)
Object (tipo objeto)	Objetos documentos
Binario (Binary)	Bytea (tipo bite)
Otros tipos	JavaScript códigos y funciones

Fuente: El autor

2.6.3 VENTAJAS DE MONGODB

Es un sistema gestor de base de datos muy estable para el almacenamiento de información, el control de acceso y la administración entre roles para usuarios que se caracteriza principalmente en la manipulación de la información a través de código en JavaScript, además de su gran velocidad de consulta como lo probamos anteriormente ya que los datos se guardan en formato JSON, y que contiene una escalabilidad horizontal para aumentar moderadamente la capacidad de trabajo y tamaño.

Para una aplicación web de gran escala se debe tomar esta base de datos documental NoSQL en cuenta ya que contiene un alto rendimiento para almacenamiento masivo desde 1 Tera Byte hasta los 70 Tera Bytes.

Permite rápidamente el acceso a información ya que estos datos se almacenan en memoria cache, además como desarrollamos el estudio pudimos observar que evita puntos de fallo por lo que el flujo de datos se distribuye a través de un balance de carga automático.

Estas bases de datos documentales NoSQL como es el caso de MongoDB son más conocidas mundialmente, ya que permite almacenar datos en relación a clientes, catálogos, almacenes de datos, tarjetas entre otros sistemas de consulta.

CAPÍTULO III

3 DESARROLLO DEL ESTUDIO COMPARATIVO.

3.1 PARÁMETROS DE COMPARACIÓN

3.1.1 INTRODUCCIÓN

En esta comparativa se establecerán lineamientos o parámetros para determinar cuál será la mejor plataforma en el desarrollo de bases de datos tango PostgreSQL como de MongoDB, para implementar por a la Clínica Odontológica Ortho Dent; se planea realizar pruebas de desempeño y calidad que ofrezcan estas bases de datos, cuyos datos obtenidos servirán para identificar cual es la mejor opción que se adapte a las necesidades de la clínica e infraestructura que ahí se posee. Una vez obtenido los resultados de la comparativa se procederán a desarrollar el sistema COOD.

Se presentan métodos o fases para realizar un buen proceso de estudio, este proceso cumple 3 fases fundamentales de comparación las cuales son:

- ✓ Planeación
- ✓ Análisis Comparativo
- ✓ Interpretación de resultados

3.1.2 FASE DE PLANEACIÓN

Esta es la primera fase fundamental es la de analizar cuáles son los parámetros a investigar de estos gestores bases de datos, comparar analizar el resultado.

3.1.3 PARÁMETROS ENTRE BASES DE DATOS

El estudio comparativo de los gestores que se van a analizar, definirá los tiempos principales a funciones de tal manera que la que tenga las mejores condiciones, se utilizará para realizar el aplicativo para la Clínica Odontológica Ortho Dent, los parámetros son:

- ✓ Instalación.
- ✓ Seguridad.
- ✓ Administración.
- ✓ Tiempos de Inserción y Recuperación.
- ✓ Uso de Recursos.
- ✓ Respuesta a Fallos.
- ✓ Costos de las Bases de Datos.

3.1.4 ESCENARIO DEL ANÁLISIS COMPARATIVO

Para evaluar a cada una de las plataformas de bases de datos es necesario determinar el software en el cual van a ser probadas o instaladas dependiendo del caso.

TABLA 28: Escenario para Análisis

Base de Datos	Versión	SSOO	Computadora
PostgreSQL	9.3.4	Windows 8.1	Sony Vaio – 6Gb Ram, 1000 Gb Disco Sistema 64 bits - Intel Core i5, 1.80GHz.
MongoDB	2.2.7	Windows 8.1	Sony Vaio – 6Gb Ram, 1000 Gb Disco Sistema 64 bits - Intel Core i5, 1.80GHz.

Fuente: Propia

3.1.5 FASE DE ANÁLISIS COMPARATIVO

Análisis de los parámetros de Comparación

Cada una de las especificaciones a ser comparadas es medible y confiable para aplicar a cada una de las plataformas, que en estado de actividad presentaran un cuadro amplio de posibilidades y desventajas. Se establecerán también los valores en base a la fiabilidad y tiempo de respuestas que presenten las plataformas, las cuales serán medidas por la escala Likert en base a una tabla de ítems.

Alternativas y Puntos Likert:

TABLA 29: Tabla de Valoración

Alternativas	Siglas	Valor
Si	S	1
No	N	0

Fuente: Escala Tipo Likert (Néstor Malave, 2007)

Estas alternativas serán como punto principal para el buen análisis de la información q se va a obtener con este estudio; se utilizó la escala tipo Likert.

3.1.6 INSTALACIÓN DE LAS BASES DE DATOS

En instalación obtendremos los diferentes tiempos y fases para configuración e instalación del gestor de base de datos de PostgreSQL a través de PgAdminIII y MongoDB, tales como el tiempo en cumplir la instalación, los pasos respectivos para configurar el gestor, el control de acceso, la asistencia en procesos para instalar.

TABLA 30: Análisis instalación

	PostgreSQL	Valor	MongoDB	Valor
Tiempo para cumplir la instalación	02:52:41 minutos	N	01:20:12 minutos	S
Pasos para configurar el gestor	01:12:11 minutos	N	00:36:18 minutos	S
Control de acceso al gestor	01:03:42 minutos	N	00:20:14 minutos	S
Asistencia en instalación	00:40:04 segundos	S	No contiene	N
Valoración		1		3

Fuente: El autor

Descripción de resultados

Los resultados obtenidos en este estudio del gestor de la base de datos no relacional MongoDB en lo que trata la instalación en sus tiempos, es menor comparada con la base de datos relacional PostgreSQL, en MongoDB carece de un modo gráfico de instalación por lo que los tiempos disminuyen y se hace de manera más rápida y sencilla.

En los pasos para configuración de MongoDB disminuye considerablemente, sin necesidad de ser un administrador o alguien que tenga los suficientes conocimientos acerca de este gestor no relacional.

En control de acceso, MongoDB no necesita usuario para el ingreso a la base de datos como es el caso de PostgreSQL que si lo necesita, sino que solo se debe iniciar el servicio por lo que se puede ingresar fácilmente.

En lo que es asistencia en la parte de la instalación de MongoDB, no necesita de una asistencia por lo que no tiene un modo gráfico, caso contrario a PostgreSQL que si lo tiene, indica explícitamente lo que se está aceptando.

3.1.7 SEGURIDAD DE LAS BASES DE DATOS

En seguridad obtendremos información acerca de qué tipo de autenticación será usada, el tipo de algoritmo, nivel de acceso en lo que es inicio de sesión del usuario en el gestor de base de datos no relacional MongoDB, como en la base de datos relacional PostgreSQL.

TABLA 31: Análisis: Seguridad

	PostgreSQL	Valor	MongoDB	Valor
Autenticación	Local y Remota	S	Local y Remota.	S
Algoritmo	SSL ³⁷ , MD5 ³⁸ , EAP ³⁹	S	Cifrado en reposo, encriptación.	N
Sesión de usuario	SSL, usando el inicio con su respectivo password	S	Roles, usando el inicio con su respectivo password.	N
Valoración		3		1

Fuente: El autor

³⁷SSL (Capa Segura de Comunicaciones): proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía.

³⁸MD5 es uno de los algoritmos de reducción criptográficos de 128 bits.

³⁹EAP Extensible Authentication Protocol es un framework de autenticación usado en redes.

Descripción de resultados

Los resultados obtenidos en este estudio del gestor de base de datos noSQL MongoDB como la base de datos PostgreSQL, la autenticación son de forma tanto local como remota, en la base de datos se debe configurar e incluir el puerto 27017 para manejar esta base de datos, caso contrario con que con el gestor de PostgreSQL.

El algoritmo para la seguridad en MongoDB de nuestras bases de datos es encriptándolo y codificándolo, en la base de datos de PostgreSQL, son de mayor valor ya que utiliza diferentes algoritmos que lo hace más seguro. Para el inicio de sesión de usuario es a través de roles que el administrador colocara.

3.1.8 ADMINISTRACIÓN DE LAS BASES DE DATOS

En administración obtendremos la información acerca de cómo maneja el gestor de base de datos no relacional MongoDB así como el gestor de base de datos PostgreSQL, como es el caso de su conexión local o remota, además de esto se revisara la facilidad como se diseña una base de datos, la creación de tablas, funciones, que tipo de consultas usa.

TABLA 32: Análisis: Administración

	PostgreSQL	Valor	MongoDB	Valor
Conexión local	Si permite conexión local.	S	Si permite conexión local.	S
Conexión remota	Si permite conexión remota.	S	Si permite conexión remota	S
Diseño de una base de datos	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Creación de colecciones (tablas)	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S

Creación de columnas	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Consultas Query	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Valoración		6		6

Fuente: El autor

Descripción de resultados

Los resultados obtenidos en este estudio del gestor de base de datos no relacional MongoDB en tanto a la administración, podemos obtener como resultados que esta base de datos no relacional, maneja la información y datos a través de comandos y sentencias Query en una consola de DOS, además esta base de datos no relacional contiene una aplicación llamada Rock Mongo, esta nos permite de manera gráfica la administración y manejo de sus bases de datos en una aplicación web. En PostgreSQL permite el diseño de bases de datos a través de una aplicación gráfica para distintos sistemas operativos con sus respectivas sentencias Query.

3.1.9 TIEMPOS DE INSERCIÓN Y RECUPERACIÓN

Se tomara en cuenta varias maneras de conocer el tiempo que se demora en insertar registros y asimismo cuanto se demoraran los gestores en recuperar dichos registros; también se tomara en cuenta la memoria RAM que se va a utilizar.

TABLA 33: Análisis: Tiempos de Peticiones y Recuperación

	PostgreSQL	Valor	MongoDB	Valor
Almacenar registros 10.000	946 milisegundos	N	103 milisegundos	S
Recuperación registros 10.000	141 milisegundos	N	102 milisegundos	S
Memoria RAM utilizada	22,00 %	N	12,00 %	S
Valoración		0		3

Fuente: El autor

Descripción de resultados

El tiempo en el cual se demora en almacenar 10000 registros en una colección de base de datos de MongoDB es de 103 milisegundos y 946 milisegundos por parte de PostgreSQL, lo que se pudo observar es que la velocidad de este gestor de base de datos es mucho mayor a la de la base de datos relacional PostgreSQL; el consumo de recursos en la memoria RAM al momento de generar estas pruebas de esta base de datos también baja drásticamente a un 12%, por lo que es más rápida y eficaz comparada con la base de datos PostgreSQL que fue de 22%.

El tiempo que se demoró en recuperar 10000 registros por parte de la base de datos NoSQL MongoDB es asimismo menor que la base de datos relacional PostgreSQL que es de es de 141 milisegundos, donde poseía 6 campos nombre, apellido, ci, dirección, teléfono y estado, por lo que significa que MongoDB sobrepasa la eficacia y rapidez que PostgreSQL.

3.1.10 USO DE RECURSOS

Mientras se usaba el gestor de bases de datos NoSQL MongoDB y el gestor de base de datos PostgreSQL, se monitorizaron los recursos tanto disco duro, memoria RAM y procesador del antes mencionado equipo en el que se realizaron las pruebas para así analizarlos pertinentemente.

TABLA 34: Análisis: Recursos

Recursos	PostgreSQL	Valor	MongoDB	Valor
Disco duro (físico) del 100%	19,52 %	N	13,03 %	S
Memoria RAM del 100%	51,02 %	N	21,46 %	S
Procesador del 100%	70,34 %	N	16,93 %	S
Valoración		0		3

Fuente: El autor

Descripción de resultados

Como podemos observar en el disco duro, ocupa mucho menor espacio para cargar la hacer uso de este gestor NoSQL comparado con el gestor de bases de datos PostgreSQL, la memoria RAM necesita de una mínima cantidad de espacio para el uso de la aplicación y mantenerla entre las diferentes aplicaciones del sistema operativo; el uso del procesador en el gestor de base de datos NoSQL es muchísimo menor o cual nos indica que ayuda en el rendimiento del computador.

3.1.11 RESPUESTA FALLOS

Debemos medir los porcentajes de inserciones fallidas para conocer su rendimiento en lo que es ingresos de los datos y acceso que se tuvo a la base de datos.

TABLA 35: Análisis: Respuesta a fallos

Parámetros	PostgreSQL	Valor	MongoDB	Valor
Total de accesos	10.000 registros	S/N	10.000 registros	S/V
Duración	5 minutos	S/N	5 minutos	S/V
Ingresos perdidos	216 registros	N	3 registros	S
Ingresos acertados	8.784 registros	N	9.997 registros	S
Porcentaje de error	2.16 %	N	0.003 %	S
Valoración		0		3

Fuente: El autor

Descripción de resultados

Como podemos observar en esta base de datos NoSQL MongoDB, contiene un mejor acceso a la información, según la prueba que realizamos, tiene una tasa del 0.003 % de errores, lo cual nos indica que tienen una mayor concurrencia de acceso e ingreso a la información contraria a la base de datos de PostgreSQL que indica que algunos registros se han perdido en la prueba.

3.1.12 COSTO DE LAS BASES DE DATOS

Se conocerá los diferentes costos de las bases de datos tanto PostgreSQL como MongoDB, las ventajas de estas bases de datos en parte es el costo monetario que conlleva el uso, ya que tiene un licenciamiento no propietario, es decir que no tiene algún costo en las descargas y uso de estas.

TABLA 36: Análisis: Costos de las Bases de Datos

	PostgreSQL	Valor	MongoDB	Valor
Costo licenciamiento	No propietario	S	No propietario	S
Costo Capacitación	1 día, 250,00 dólares	N	1 día, 150,00 dólares	S
Costo infraestructura	0 dólares	S	0 dólares	S
Agregados	0 dólares	S	0 dólares	S
Total	7.500,00 mensual	N	4.500,00 mensual	S
Valoración		1		5

Fuente: El autor

Descripción de resultados

Se tomó en cuenta los diferentes costos tanto mensuales como anuales los cuales nos indican que definitivamente MongoDB es el gestor de bases de datos con un costo menor a la que nos indica PostgreSQL por lo que de alguna manera se obtendría una gran disminución en lo que es gastos para una empresa.

3.2 ANÁLISIS COMPARATIVO

3.3 ANÁLISIS FINAL

El análisis anteriormente estudiado se puede observar los diferentes resultados con los cuales podremos conocer con cual base de datos ya sea relacional como lo es PostgreSQL o con NoSQL MongoDB se realizara el respectivo sistema para la Clínica Odontológica Ortho Dent.

Se concluye como la mejor opción para la Clínica Odontológica es la base de datos NoSQL MongoDB; En la siguiente tabla se podrá observar el resumen de cada uno de las comparaciones y el ganador de este estudio comparativo, a través del análisis de la escala tipo Likert (Néstor Malave, 2007) ; con esto se contabilizara para obtener el mejor valor.

TABLA 37: Análisis: Análisis Final

	PostgreSQL	Valor	MongoDB	Valor
Instalación				
Tiempo para cumplir la instalación	02:52:41 minutos	N	01:20:12 minutos	S
Pasos para configurar el gestor	01:12:11 minutos	N	00:36:18 minutos	S
Control de acceso al gestor	01:03:42 minutos	N	00:20:14 minutos	S
Asistencia en instalación	00:40:04 segundos	S	No contiene	N
Valoración		1		3
Seguridad				
Autenticación	Local y Remota	S	Local y Remota.	S
Algoritmo	SSL ⁴⁰ , MD5 ⁴¹ ,EAP ⁴²	S	Cifrado en reposo, encriptación.	N

⁴⁰SSL (Capa Segura de Comunicaciones): proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía.

⁴¹MD5 es uno de los algoritmos de reducción criptográficos de 128 bits.

⁴²EAP Extensible Authentication Protocol es un framework de autenticación usado en redes.

Sesión de usuario	SSL, usando el inicio con su respectivo password	S	Roles, usando el inicio con su respectivo password.	N
Valoración		3		1
Administración				
	PostgreSQL	Valor	MongoDB	Valor
Conexión local	Si permite conexión local.	S	Si permite conexión local.	S
Conexión remota	Si permite conexión remota.	S	Si permite conexión remota	S
Diseño de una base de datos	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Creación de colecciones (tablas)	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Creación de columnas	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Consultas Query	A través de una interfaz, contiene editor de SQL, también usando AdminpgIII.	S	Si permite, también usando herramienta en modo grafico Rock Mongo.	S
Valoración		6		6

Tiempos de Inserción y Recuperación				
	PostgreSQL	Valor	MongoDB	Valor
Almacenar registros 10.000	686 milisegundos	N	107 milisegundos	S
Recuperación registros 10.000	4.000 milisegundos	N	1.624 milisegundos	S
Memoria RAM utilizada	22,00 %	N	12,00 %	S
Valoración		0		3
Uso de Recursos				
	PostgreSQL	Valor	MongoDB	Valor
Disco duro (físico) del 100%	19,52 %	N	13,03 %	S
Memoria RAM del 100%	51,02 %	N	21,46 %	S
Procesador del 100%	70,34 %	N	16,93 %	S
Valoración		0		3
Respuesta a Fallos				
	PostgreSQL	Valor	MongoDB	Valor
Total de accesos	10.000 registros	S/N	10.000 registros	S/V
Duración	5 minutos	S/N	5 minutos	S/V
Ingresos perdidos	216 registros	N	3 registros	S
Ingresos acertados	8.784 registros	N	9.997 registros	S
Porcentaje de error	2.16 %	N	0.003 %	S
Valoración		0		3
Costo de las Bases de Datos				
	PostgreSQL	Valor	MongoDB	Valor
Costo licenciamiento	No propietario	S	No propietario	S
Costo Capacitación	1 día, 250,00 dólares	N	1 día, 150,00 dólares	S
Costo infraestructura	0 dólares	S	0 dólares	S

Agregados	0 dólares	S	0 dólares	S
Total	7.500,00 mensual	N	4.500,00 mensual	S
Valoración		1		5
	Total	11/27		24/27
	Porcentaje de 100%	11,12		88,88

Fuente: El autor

Como podemos observar en la tabla que nos indica el valor para los diferentes parámetros en los que se analizó, muchos de los parámetros demuestran que la base de datos NoSQL cumplen con todos los aspectos de ser una excelente base de datos para realizar aplicaciones como es la que se realizara en la Clínica Odontológica Ortho Dent.

Se ha analizado el porcentaje de éxito para realizar el uso de este gestor de bases de datos MongoDB en la aplicación para la Clínica Odontológica Ortho Dent que se observa a continuación.

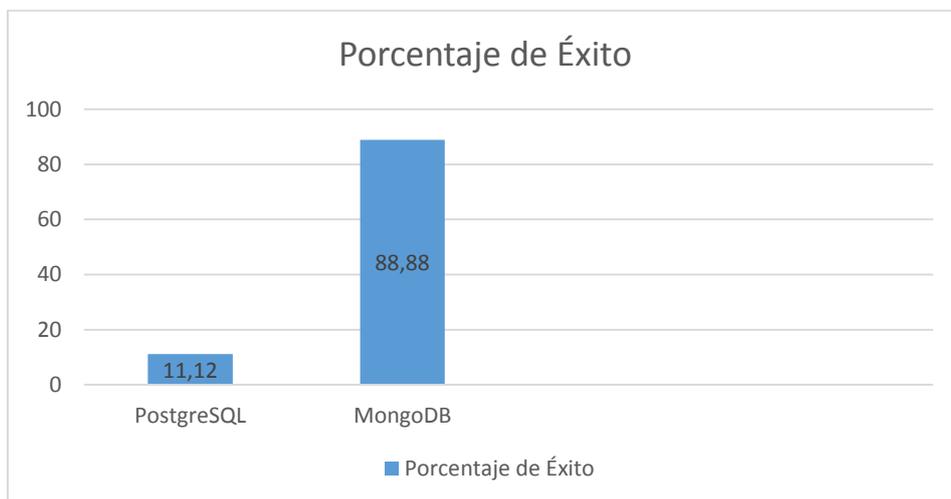


ILUSTRACIÓN 17: Porcentaje de éxito

Fuente: El autor

La opción más viable para realizar el sistema para la Clínica es a través de la base de datos NoSQL MongoDB para tener un porcentaje de éxito del 88,88% por ciento por lo que permitirá realizar la aplicación de manera rápida y eficaz.

Es muy importante indicar que tanto como el manejo de ACID y de transacciones en las ventajas que posee el gestor de bases de datos MongoDB son de grande inconveniente.

La velocidad en respuesta a obtención de información de la base de datos NoSQL de MongoDB ha denotado que grandes compañías como es el caso de uso de FourSquare, Electronic Arts, Disney, Gilt, Viacom, IGN, github, viber entre otras están haciendo uso de esta excelente base de datos documental. Debido a su gran escalabilidad horizontal, hace que cada año sea la más usada.

3.3.1 CONCLUSIÓN MONGODB

Este estudio de dos bases de datos SQL como NoSQL es principalmente útil para la gestión y almacenamiento de información; por lo que se las están usando paulatinamente en la sociedad.

Por lo que en este estudio se ha logrado obtener información detallada acerca de este gestor de bases de datos NoSQL, lo que formo varias ideas que son ventajosas para el tesista; por lo que se desarrollara con este gestor, la aplicación para la clínica Odontológica Ortho Dent de la ciudad de Ibarra.

MongoDB llega a ser el doble o triple de rápido con respecto a PostgreSQL, si se desea utilizar velocidad pero poca con pérdida de información en el tiempo, MongoDB es la mejor elección; caso contrario necesitas de PostgreSQL.

NoSQL como lo es MongoDB es enteramente más rápido que bases de datos SQL, por lo que soluciona distintos problemas, estas bases de datos NoSQL se centran más en almacenar y consultar de manera rápida y eficaz con la única desventaja de que sus datos puedan perderse con el tiempo.

Es posible el desarrollo de aplicaciones web en los que se necesita de una base de datos no relacional como es el caso de MongoDB.

En diferentes sitios web que han manejado este tipo de bases de datos NoSQL, han logrado manejar de manera exitosa la información de sus sitios en tiempo real.

En lo que trata la base de datos MongoDB en lo que es seguridad en su información, esta no posee y por esta razón se debe considerar en diferentes aplicaciones importantes su uso de esta base de datos, como por ejemplo cuentas bancarias y sus transacciones.

Se sugiere como una buena opción de base de datos, además esta base de datos NoSQL lograría el ahorro para la Clínica Odontología Ortho Dent de la ciudad de Ibarra.

En conclusión por poseer un conocimiento acerca de esta base de datos Documental NoSQL MongoDB, se concluye que esta herramienta que se la estudió profundamente es apta para realizar el desarrollo de sistema web para la Clínica Odontológica Ortho Dent de la ciudad de Ibarra.

3.4 METODOLOGÍA DE DESARROLLO

3.4.1 INTRODUCCIÓN A RUP

Varias empresas que se dedican a desarrollar aplicaciones han usado esta metodología para su creación y conocimiento de dichas aplicaciones, que permita el soporte y conocimiento de las aplicaciones y de la compañía.

RUP es una metodología que indica el porque, el cómo, el quien, el que tiempo y acciones que se van a desarrollar en el sistema, esta metodología nos ayudara a conocer cuáles son los diferentes roles que un usuario poseerá en la aplicación y que parte del sistema es la que va a ser uso. Esta metodología indica las diferentes etapas ordenadamente que indicaran su ciclo de vida del sistema, como es la fase de planificación, inicio, elaboración, construcción e implementación; para así obtener el éxito en la aplicación.

Tanto las aplicaciones de mayor tamaño como las de menor tamaño, RUP es la indicada ya que nos ayudara en el manejo de un proceso complejo en las diferentes etapas de la aplicación, la cual se conseguirá menores costos en la dedicación de la aplicación.

3.4.2 HISTORIA DE RUP

El antecedente más importante se ubica en 1967 con la Metodología Ericsson (Ericsson Approach) elaborada por Ivar Jacobson, una aproximación de desarrollo basada en componentes, que introdujo el concepto de Caso de Uso. Entre los años de 1987 a 1995 Jacobson fundó la compañía Objectory AB y lanza el proceso de desarrollo Objectory (abreviación de Object Factory).

Posteriormente en 1995 Rational Software Corporation adquiere Objectory AB y entre 1995 y 1997 se desarrolla Rational Objectory Process (ROP) a partir de Objectory 3.8 y del Enfoque Rational (Rational Approach) adoptando UML como lenguaje de modelado.

Desde ese entonces y a la cabeza de Grady Booch, Ivar Jacobson y James Rumbaugh, RationalSoftware desarrolló e incorporó diversos elementos para expandir ROP, destacándose especialmente el flujo de trabajo conocido como modelado del negocio. En junio del 1998 se lanza RationalUnified Process⁴³.

3.4.3 RUP (PROCESO UNIFICADO RACIONAL)

El Proceso Unificado de Rational (Rational Unified Process en inglés, habitualmente resumido como RUP) es un proceso de desarrollo de software y junto con el Lenguaje Unificado de Modelado UML, constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos.⁴⁴

⁴³Rational Unified Process (RUP), tomado de <http://ima.udg.edu/~sellares/EINFES2/Present1011/MetodoPesadesRUP.pdf>

⁴⁴Franco, N,& Amparo, A (2013) Aplicación Web para la administración online de citas médicas en el Centro Médico de Orientación y Planificación familiar CEMOPLAF-Otavaló; utilizando el patrón de arquitectura MVC en PHP. aplicativo: CEMOCITE Sistema de citas médicas online CEMOPLAF-Otavaló. Universidad Técnica del Norte, Ecuador

3.4.4 CARACTERÍSTICAS RUP

RUP contiene tres características principales que son:

✓ **Casos de Uso**

Estos casos de uso son guías que indican las actividades, requisitos del sistema, entre otras, que se realizan en todo el proceso de desarrollo del sistema como es en el diseño, implementación y prueba.

✓ **Arquitectura centrada**

Esta metodología permite obtener una visión común para todos los desarrolladores como usuarios que van a hacer uso del sistema, además de esto, permite la toma de decisiones para mejorar elementos principales como la calidad, evolución del sistema y rendimiento de la aplicación.

✓ **Iterativo incremental**

Se planifica el sistema en ciclos los cuales se verán reflejados en el tiempo, incrementando la información de cada paso, haciendo que su resultado sea completo indicando sus actividades en cualquier parte del desarrollo.

3.4.5 ESTRUCTURA DE LA METODOLOGÍA RUP

La metodología RUP se la divide a lo largo de varios ciclos de la vida del sistema, consta de cuatro ciclos para el desarrollo de las aplicaciones como son: Inicio, Elaboración, Construcción y Transición, estas iteraciones tienen un tiempo definido en las que nos ayudan a definir decisiones importantes en el desarrollo y alcanzar metas.

✓ **Fase de Inicio**

Nos indicara su visión inicial del desarrollo de la aplicación, alcance, los casos de uso y todos los requisitos principales que necesitamos para su desarrollo eficaz, además se analizará su factibilidad, el costo de implementación, los objetivos principales entre otras. El tiempo en demorarse esta fase no es más que una semana; se realizara los siguientes puntos:

1. Se indicara cuáles son sus límites.
2. Indicara los casos de uso principales de la aplicación y funcionalidad.
3. Indicara la arquitectura de la aplicación.
4. Indicara los costes de tiempo y recursos de la aplicación.
5. Indicara riesgos de la aplicación.

✓ **Fase de Elaboración**

En esta fase se analizara y planificara actividades para el desarrollo de la aplicación hacia el equipo de trabajo, se iniciara con los cimientos de la aplicación así como el problema principal que conlleva en el desarrollo. Se identifican las necesidades y diseño de la arquitectura.

Se diseña un prototipo inicial de la arquitectura y avanza drásticamente hasta convertirse en un sistema final; se realizara los siguientes puntos:

1. Validar y definir la arquitectura para la aplicación.
2. Visión de manera completa.
3. Indicara cuál será el plan para el desarrollo de la aplicación.
4. Indicara que de acuerdo a lo investigado, se desarrollara la aplicación.

✓ **Fase de Construcción**

Esta fase es el desarrollo total del sistema desde su inicio hasta su finalización, la entrega a la empresa; abarcaría de forma incremental desde las iteraciones antes mencionadas pasando por sus componentes, requisitos, calidad de la aplicación; se realizara los siguientes puntos:

1. Obtendrá la mejor calidad en el desarrollo de la aplicación
2. Obtendrá una primera versión de la aplicación, un prototipo beta.
3. Obtendrá una arquitectura total.

✓ Fase de Transición y Despliegue

En esta fase de transición y despliegue, indica la instalación total en la empresa del producto solicitado para que puedan hacer uso del mismo; si es que el sistema necesitara de un nuevo requisito por parte del usuario, se lo deberá analizar y crear una nueva versión del sistema para satisfacer las necesidades de la empresa.

En esta fase se deberá entregar la documentación principal para conocimiento del usuario en lo que es manejo del sistema, además se deberá preparar al usuario para su uso de la aplicación, si por algún caso se necesitara ajustes, configuración del sistema, se lo deberá indicar en esta fase; se realizara los siguientes puntos:

1. Se realizaran pruebas de la aplicación beta.
2. Se entrenara a los usuarios para el uso de esta aplicación.
3. Se transmitirá el producto a la empresa para el buen uso de esta.

3.4.6 ESTRUCTURA DE LOS PROCESOS, ROLES, ACTIVIDADES, ARTEFACTOS Y FLUJOS DE TRABAJO

En este proceso de la metodología de desarrollo RUP define estos elementos como son los procesos, roles, actividades, flujos de trabajo y artefactos como partes principales para el entendimiento de los usuarios y equipo de trabajo que va a ser parte del proceso de desarrollo.

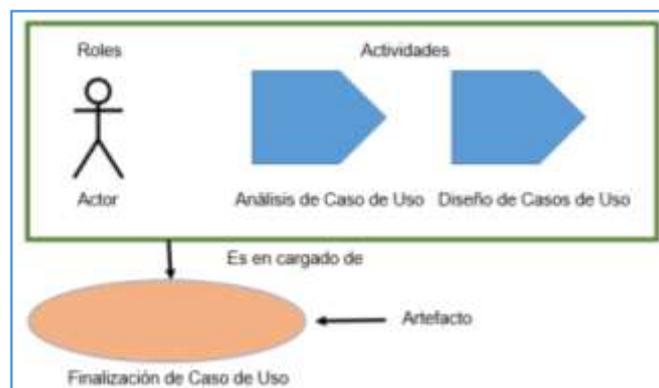


ILUSTRACIÓN 18: Estructura de Procesos, Roles, Actividades, Artefactos

Fuente: El autor

En esta estructura de RUP se definen cuatro partes esenciales para el desarrollo de la metodología como son:

- ✓ **Roles:** Es el principal usuario o grupo de usuarios que realizan a cabalidad actividades que se indican y así finalizar los artefactos de manera eficaz.
- ✓ **Actividades:** Son tareas que las realiza un rol o usuario, con el conjunto de estas actividades realizadas se crean los artefactos.
- ✓ **Artefactos:** Es toda la documentación o archivos esenciales que se obtiene a través del desarrollo de la aplicación.
- ✓ **Flujos de trabajo:** son las diferentes secuencias que cumplen los roles o usuarios en la realización de las actividades.

CAPÍTULO IV

4 DESARROLLO DEL SISTEMA DE GESTIÓN DE CLIENTES Y REGISTRO DE PAGOS PARA LA EMPRESA.

4.1 FASE DE INICIO

En esta fase de la metodología de desarrollo RUP, se definen el alcance del proyecto a través de un documento llamado Visión, este documento se obtiene a través de entrevistas, actas de trabajo y además se desplegó un plan de desarrollo de software.

4.1.1 DOCUMENTO DE LA VISIÓN

Introducción

Para conocer y entender el funcionamiento de la Clínica Odontológica Ortho Dent en adelante (COOD), en el cual se implantara este sistema, fue necesario realizar varias visitas y dialogar con las personas que trabajan en ella, para con esto definir los diferentes procesos que necesitan ser automatizados.

Propósito

El propósito de este documento es de definir los diferentes requerimientos para el sistema para la gestión de clientes y registro de pagos para la "COOD".

Este sistema antes mencionado se encargará de los procesos principales que son la Gestión de Clientes y Registro de Pagos que se detallará de manera automática y comprensible; este documento especificara las necesidades de los usuarios a través de los Casos de Uso.

Alcance

Este documento de visión se lo utiliza para especificar los diferentes problemas que tiene la Clínica Odontológica Ortho Dent y tener las oportunidades de resolverlas a través del sistema de gestión de clientes y registro de pagos, que será desarrollado por mi persona, también describe la funcionalidad y características de la solución general a través de Modelos de Caso de Uso.

Oportunidades de Negocio

A partir de la información obtenida anteriormente y a través de los procedimientos establecidos en la Clínica Odontológica Ortho Dent por la administración, se determina a desarrollar el sistema para la gestión de clientes y registro de pagos que ayudara a la clínica a agilizar toda la gestión de actividades relacionadas con el manejo de clientes y pagos de manera digital, además de cumplir con los requerimientos de los implicados para mejorar estos servicios.

Permitirá a la Clínica Odontología Ortho Dent contar con un registro de la información de manera digital en los diferentes tipos de tratamiento, ofrecerá un completo control de pagos y facturación así como recibos por cliente a través de las diferentes interfaces que se les facilitará en el manejo de información. Con el tiempo esto conllevara a menos pérdidas de información importante y un completo registro informático que favorecerá a esta empresa.

Luego de haber realizado un análisis de lo recolectado, se estableció que la Clínica Ortho Dent cuenta con algunas falencias como son:

- ✓ **No cuenta con un sistema computarizado:** La clínica no contiene un sistema que almacene información de clientes o pagos por lo que la información puede perderse o ser modificada.
- ✓ **No existe un gestor de bases de datos:** la Clínica no cuenta con un espacio de almacenamiento de información importante digital de clientes provocando modificación y pérdida de datos.
- ✓ **Tecnología desactualizada:** La clínica no cuenta con un sistema digital sino que almacena información personal de clientes en ficheros o hojas manuales, además tiene un control de pagos en agendas por lo que podrían ser susceptibles a alteración.
- ✓ **Poco control y auditoría:** En la clínica carece de ningún control de información de clientes o pagos que permita validar información.
- ✓ **Seguridad ineficiente:** La clínica no maneja seguridad de la información por lo que se compromete la confidencialidad de la información.

Con las referencias antes expuestas, la administración de la Clínica Odontológica Ortho Dent, ha planteado el desarrollo de un sistema informático para trabajar de manera integral y eficientemente. Este nuevo sistema que se implantará, permitirá facilidades para el manejo de la información de clientes como de pagos generados por la Clínica. Los módulos siguientes serán:

- ✓ Módulo de Gestión de Usuarios.
- ✓ Módulo de Facturación.
- ✓ Módulo de Gestión de Aplicación.
- ✓ Módulo de Autoría.
- ✓ Módulo de Reportes

Especificaciones de Requerimientos

Una de las piezas fundamentales para el desarrollo de este proyecto son los requerimientos, describe las funciones y el desempeño que debe tener el sistema, cuenta con los siguientes puntos que se deberán realizar:

- ✓ Analizar los recursos que se necesitaran en este sistema.
- ✓ Especificar el tipo de verificación que se realizaran al sistema.
- ✓ Planear pruebas funcionales a la que se someterá el sistema.

El sistema que se propone realizar será capaz de permitir a la Clínica Odontológica Ortho Dent; con el almacenamiento y la gestión de la información de clientes nuevos así como de antiguos, llevar un registro de pagos para cada uno de ellos que se han registrado en esta Clínica, mediante el uso de una página web, la aplicación tendrá como nombre SYS-OD.

Definición de Problema

TABLA 38: Definición del Problema

Características	Observaciones
El problema	No contar con un sistema que administre los módulos de gestión de clientes y registro facturación. No existe seguridad en el manejo de la información. Manejo inadecuado del capital obtenido por los clientes.
En que Afecta	Todo el personal que maneja los registros tanto como clientes como el caso de monetario.
Impacto	Falta de un sistema para la gestión de pagos. Existen muchas actividades manuales de la clínica que no permite una gestión eficiente. Falta de inconsistencia.
Solución	Solucionar los requerimientos de los involucrados en el proceso de gestión de clientes y registro de pagos. Cubrir las necesidades de la Clínica Odontológica. Automatizar el manejo de la información importante para la Clínica. Brindar documentación confiable sobre el manejo del sistema. Implementar la solución informática de calidad a través de una metodología eficaz de desarrollo de software.

Fuente: El autor

Descripción de interesados usuarios

En las siguientes tablas indican la responsabilidad que tiene el interesado y usuarios.

Interesados, son todas personas que están involucradas en la definición y alcance del proyecto, a continuación se definen los siguientes interesados:

TABLA 39: Involucrados Interesados

Nombre	Descripción	Responsabilidad
Coordinador del proyecto	Responsable a nivel de directivo de la clínica odontológica Ortho Dent.	Establecer lineamientos, requerimientos y recursos esenciales para el buen desenvolvimiento de la aplicación.
Responsable del proyecto	Desarrollador de la tesis, responsable del proyecto.	Establece el buen desarrollo, diseño y análisis de la aplicación.
Responsable Funcional	Responsable por cada módulo que se desarrollara. Proporciona información de los clientes y pagos de la clínica odontológica.	Responsable de establecer todos los requerimientos de la aplicación para un buen desarrollo oportuno. Define la información que se va a analizar y utilizar en el sistema como clientes y pagos.

Fuente: El autor

Los usuarios son aquellos que harán uso del sistema SYS-OD para la Clínica Odontológica Ortho Dent.

TABLA 40: Involucrados Usuarios

Nombre	Descripción	Responsabilidad
Administrador funcional	Persona de la clínica odontológica Ortho Dent que administra el sistema SYS-OD.	Administra Funcionalmente el sistema dándole la potestad de investigar acerca de la información del cliente y sus pagos.
Usuario del sistema	Personal que labora en la clínica odontológica Ortho Dent.	Ingresa información principal de los clientes y sus pagos.

Fuente: El autor

Entorno de los usuarios

- ✓ El personal de la clínica odontológica, serán los usuarios de este sistema, que permitirá registrar en el sistema y llevara un estricto control de clientes y pagos que se realizan diariamente en la empresa.
- ✓ Existirán dos módulos los cuales facilitaran el mejor manejo de información.
- ✓ Actualmente existen sistemas para clínicas odontológicas, pero se debe tomar mucho en cuenta que cada empresa tiene diferentes lineamientos por lo que no existe un sistema que cumpla todas las necesidades que requiere la clínica odontológica Ortho Dent de la ciudad de Ibarra.
- ✓ Estos módulos tanto de clientes como de pagos, formaran un solo sistema integrado de información de la "COOD", los mismos que deben interactuar entre sí.

TABLA 41: Coordinador del Sistema

Representante	Dra. Beatriz Rosero
Descripción	Responsable a nivel directivo del proyecto por parte de la Clínica Odontológica Ortho Dent de la ciudad de Ibarra.
Tipo	Gerente de la Clínica Odontológica Ortho Dent.
Responsabilidades	Establece el lineamiento principal para el desarrollo del sistema. Coordina los requerimientos que surjan en el desarrollo del sistema.
Criterios de éxito	Mantener activo y usar este sistema en la clínica odontológica.
Implicación	Revisor de la administración.
Entregables	N/A
Comentarios	Mantiene la información con el desarrollo del sistema.

Fuente: El autor

TABLA 42: Responsable del Sistema

Representante	Luis Correa
Descripción	Responsable del desarrollo del sistema para Clínica Odontológica Ortho Dent.
Tipo	Analista de sistemas.
Responsabilidades	Responsable del diseño, análisis, construcción e implementación del sistema. Gestionar el correcto desarrollo del sistema, así como la excelente comunicación con los administradores y personal que labora en la clínica.
Criterios de éxito	Cumplir con todas las necesidades de los involucrados de la clínica con respecto al sistema. Cumplir a cabalidad el cronograma antes analizado. Desarrollar un sistema de calidad que cumpla los requisitos de la Clínica Odontológica Ortho Dent.
Implicación	Jefe del proyecto (Project Manager).
Entregables	Documento de visión. Resumen del modelo de casos de uso. Especificaciones del modelo de casos de uso.

Fuente: El autor

TABLA 43: Responsable Funcional

Representante	María José Chasiquiza
Descripción	Responsable a nivel de personal por parte de la Clínica Odontológica Ortho Dent de la ciudad de Ibarra.
Tipo	Secretaria de la Clínica Odontológica Ortho Dent.
Responsabilidades	Responsable de coordinar con el personal el correcto manejo del sistema. Coordinar las pruebas de validación del sistema. Coordinar la excelente capacitación del sistema a los usuarios.

Criterios de éxito	Obtener un sistema de calidad que cumpla con los requerimientos funcionales establecidos.
Implicación	Aprueba todos los requerimientos y pruebas realizadas.
Entregables	Documento de revisión de todas las especificaciones funcionales. Documento de las pruebas funcionales.
Comentarios	N/A

Fuente: El autor

TABLA 44: Administrador del Sistema

Representante	Dra. Beatriz Rosero
Descripción	Administradora de la Clínica Odontológica Ortho Dent.
Tipo	Gerente de la Clínica Odontológica Ortho Dent.
Responsabilidades	Administrar funcionalmente el sistema.
Criterios de éxito	N/A.
Implicación	Revisor de la administración.
Entregables	Documentación de control de nuevos requerimientos. Documentación de incidencias del sistema. Manual técnico Manual de Instalación Manual de Usuario
Comentarios	N/A

Fuente: El autor

TABLA 45: Auditor del Sistema

Representante	Dra. Beatriz Rosero
Descripción	Administradora de la Clínica Odontológica Ortho Dent.
Tipo	Gerente de la Clínica Odontológica Ortho Dent.
Responsabilidades	Administrar funcionalmente el sistema.
Criterios de éxito	N/A.
Implicación	N/A
Entregables	Manual de Usuario
Comentarios	N/A

Fuente: El autor

TABLA 46: Usuario del Sistema

Representante	María José Chasiquiza
Descripción	Responsable a nivel de personal por parte de la Clínica Odontológica Ortho Dent de la ciudad de Ibarra.
Tipo	Secretaria de la Clínica Odontológica Ortho Dent.
Responsabilidades	Responsable de coordinar con el personal el correcto manejo del sistema. Coordinar las pruebas de validación del sistema. Coordinar la excelente capacitación del sistema a los usuarios.
Criterios de éxito	Obtener un sistema de calidad que cumpla con los requerimientos funcionales establecidos.
Implicación	Aprueba todos los requerimientos y pruebas realizadas.
Entregables	Documento de revisión de todas las especificaciones funcionales. Documento de las pruebas funcionales. Manual técnico Manual de Instalación Manual de Usuario
Comentarios	N/A

Fuente: El autor

TABLA 47: Necesidades de Usuario del Sistema

Necesidad	Prioridad	Inquietudes	Situación Actual	Solución propuesta
Desarrollar un sistema que automatice los procesos en el manejo de la información.	Alta	Se debe unificar toda la información para facilitar la gestión y administración de la información.	La clínica no cuenta con un sistema de gestión de clientes y registro de pagos.	Implementar un sistema para la gestión de clientes y registro de pagos.
Desarrollar un sistema utilizando un sistema de software libre para así poder agilizar y facilitar el desarrollo.	Alta	Se debe utilizar un software libre para el desarrollo del sistema.	N/A	Desarrollar un sistema con plataforma JEE, framework JSF, Hibernate, base de datos MongoDB.
La interfaz de la aplicación debe ser fácil de usar, efectuando los diferentes requerimientos que se establecieron.	Alta	Se debe efectuar todos los requerimientos descritos por los usuarios que van a hacer uso del sistema.	N/A	Desarrollo de la aplicación con la ayuda de los usuarios que trabajan en la clínica odontológica Ortho Dent.

Fuente: El autor

Vista general del Producto

En este conjunto de información a alto nivel, se mostrara las funciones del sistema que se desarrollaran.

Perspectiva del Producto

La perspectiva del producto señalara cuales son los módulos que se implantaran en el sistema SYS-OD para la Clínica Odontológica Ortho Dent.

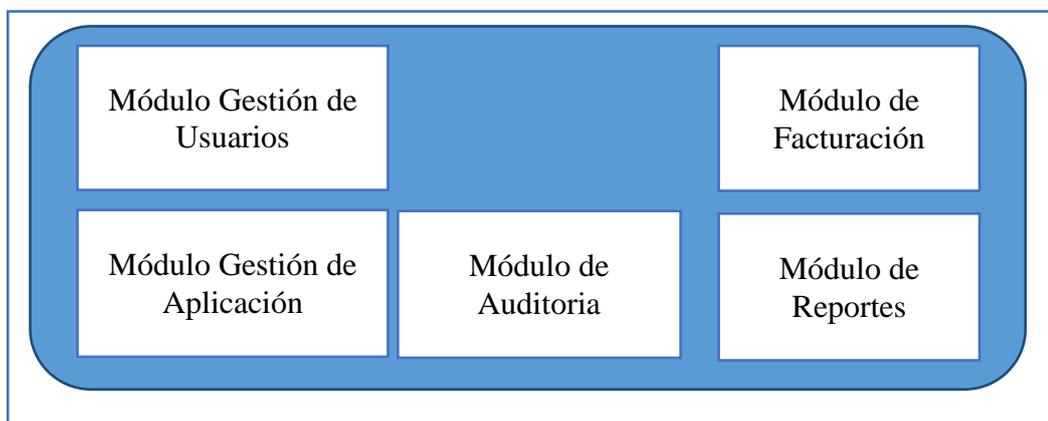


ILUSTRACIÓN 19: Prospectiva del producto

Fuente: El autor

Resumen de capacidades

El sistema de gestión de clientes y registro de pagos mejorara el proceso ya que se lo automatizara, además permitirá:

- ✓ Controlar la seguridad de la información.
- ✓ Almacenar la información en formato digital.
- ✓ Disminuir tiempos de respuesta a búsquedas de información del cliente.
- ✓ Controlar los pagos de cada cliente detallándolo en reportes.
- ✓ Administrar los diferentes tipos de usuarios que van a usar el sistema.

TABLA 48: Módulo de Gestión de Cliente

Beneficios para el Usuario	Características que lo soportan
La información del cliente se almacenará de manera digital.	De acuerdo a su facilidad de manejo y entendimiento del módulo, se podrá ingresar la información de cada cliente de manera rápida y segura.
El acceso a la información del cliente será de manera fácil y sencilla.	Al contar con un sistema computacional, se podrá disminuir tiempos de respuesta. Brindará los reportes respectivos.
Se tendrá alta disponibilidad.	La aplicación se lo realiza a través de un sistema web, por lo que permitirá un acceso inmediato desde cualquier punto de la intranet de la Clínica Odontológica Ortho Dent.
El módulo de gestión de clientes interactúa con el módulo de facturación.	Estos dos módulos participaran en el buen desempeño y mantener un control estricto de cada cliente.

Fuente: El autor

TABLA 49: Módulo de Facturación

Beneficios para el Usuario	Características que lo soportan
El registro de facturación se lo realiza con mayor seguridad.	El módulo de facturación permitirá visualizar toda la información con respecto a los pagos de cada cliente.
El acceso a la información será de manera fácil y sencilla.	Permite mostrar la información del cliente con respecto a los pagos realizados y pendientes que tengan con la Clínica Odontológica.
Se tendrá alta disponibilidad de la información.	La aplicación se lo realiza a través de un sistema web, por lo que permitirá un acceso inmediato desde cualquier punto de la intranet de la "COOD".
El módulo de facturación se unifica la información.	Al ser el módulo importante para la "COOD" ya que es el módulo donde se hace el desenvolvimiento de ingresos de cada pago.

Fuente: El autor

TABLA 50: Módulo de Auditoria

Beneficios para el Usuario	Características que lo soportan
El control completo de la información con mayor seguridad.	El módulo permitirá el control completo en el momento de que se modifique o exista algún tipo de cambio en la información de algún campo; se observara la información más significativa.
El módulo de auditoria se unifica la información.	Este módulo de auditoria refleja toda la información que se modifique en el sistema de la "COOD" ya que es el módulo donde se hace el control en la modificación de la información.

Fuente: El autor

TABLA 51: Módulo de Gestión de Aplicación

Beneficios para el Usuario	Características que lo soportan
Gestiona toda la aplicación	El módulo permitirá controlar toda la parte interna del sistema en general.
Controla los parámetros del sistema.	Este módulo controlara los diferentes parámetros que conlleve la modificación en campos.
Se tendrá integridad de datos en la gestión de la aplicación.	La información que se ingrese al sistema web, será almacenada de tal manera que será completamente segura.
El módulo de gestión de la aplicación unifica la información.	Al ser el módulo importante en lo que es seguridad para la "COOD".

Fuente: El autor

TABLA 52: Módulo de Reportes

Beneficios para el Usuario	Características que lo soportan
El módulo de reportes mostrará información primordial	Este módulo que forma parte del sistema para la "COOD" permitirá mostrar los reportes pertinentes en los pagos, recibos e información principal.

Fuente: El autor

Costos y Precios

Para el adecuado uso e implantación del sistema SYS-OD para la Clínica Odontológica Ortho Dent, la clínica deberá tomar en cuenta estas especificaciones.

TABLA 53: Costos y Precios

		HARDWARE		
Nro.	Descripción	Costo Clínica	Costo Tesista	Costo Real
1	Equipo portátil	\$ -	\$ 950,00	\$ -
	Subtotal	\$ -	\$ 950,00	\$ -
		CAPACITACIONES		
1	Curso de capacitación continua FICA	\$ -	\$ 80,00	\$ 80,00
2	Cursos externos	\$ -	\$ 150,00	\$ 150,00
	Subtotal	\$ -	\$ 230,00	\$ 230,00
		SOFTWARE		
1	IDE Netbeans	\$ -	\$ -	\$ -
2	Base de datos MongoDB	\$ -	\$ -	\$ -
3	JasperReport	\$ -	\$ -	\$ -
4	JDK 8,25 Y JRE 8,25	\$ -	\$ -	\$ -
	Subtotal	\$ -	\$ -	\$ -
		MATERIALES DE OFICINA		
1	Resmas de papel bond	\$ -	\$ 30,00	\$ 30,00
2	Copia de documentos, libros	\$ -	\$ 40,00	\$ 40,00
3	Internet	\$ -	\$ 30,00	\$ 30,00
4	Memoria Flash	\$ -	\$ 30,00	\$ 30,00
5	Anillados y empastados	\$ -	\$ 140,00	\$ 140,00
	Subtotal	\$ -	\$ 270,00	\$ 270,00

		VARIOS		
1	Movilización	\$ -	\$ 200,00	\$ 200,00
2	Imprevistos	\$ -	\$ 300,00	\$ 300,00
	Subtotal	\$ -	\$ 500,00	\$ 500,00
	TOTAL	\$ -	\$1950,00	\$1000,00

Fuente: El autor

Licenciamiento e instalación

Dado que la aplicación SYS-OD para la Clínica Odontológica Ortho Dent se desarrollara con el uso de software libre, no se requiere de ningún tipo de licencia; la instalación y puesta en marcha se verá a cabo por el tesista que ha desarrollado el sistema.

Características del producto

Facilidad de acceso y uso

La aplicación será desarrollado con tecnología Web, en la parte visual con el uso de PrimeFaces ya que mejora considerablemente en el rendimiento y facilidad de uso, utilizando páginas web de manera clara y sencilla para los usuarios de la "COOD".

Mejorar el control y verificación de datos

El control de la información de los clientes así como la información contable de la "COOD" es parte importante de los requisitos del sistema para que no exista ningún dato modificado ni errores al momento de almacenar la información con respecto a los pagos que hacen los clientes.

Restricciones

Por cuanto la "COOD" y sus sucursales, no cuentan con los implementos necesarios para implementar un sistema web, se indica que el sistema no será publicado al internet, para que el sistema esté en línea se necesita las correspondientes seguridades del caso para trabajar en el entorno especificado.

Rangos de calidad

El desarrollo del sistema SYS-OD para la “COOD” se ha elaborado siguiendo la metodología de desarrollo de software RUP, empleando los requisitos que esta metodología implica para su desarrollo íntegro y así asegurar la producción de un sistema software de alta calidad, y que se ajuste a las necesidades de los usuarios de la Clínica Odontológica Ortho Dent.

Precedencia y Prioridad

Las prioridades del sistema son:

- ✓ Mantener la información segura en un sistema digital.
- ✓ Asegurar que la información no sea alterada de ninguna forma.
- ✓ Realizar de manera correcta los procesos financieros de la “COOD”.
- ✓ Integrar los dos módulos del sistema.
- ✓ Facilitar con herramientas que sean amigables con el usuario.
- ✓ Asegurar que la información solo pueda ser manipulada con el usuario pertinente a través de controles de acceso y privilegios dependiendo de los roles.
- ✓ Implementar métodos de búsqueda fáciles de utilizar para obtener información detallada de cada cliente.

Otros requisitos del sistema a desarrollar

Para implementar el sistema SYS-OD se necesita cumplir con los siguientes requisitos:

- ✓ Un computador que sea servidor con las siguientes aplicaciones y características.

- Sistema operativo Windows 8,1.
- Base de datos MongoDB versión 2.6.5.
- Servidor Glassfish versión 4.0.
- Puerto de red.
- ✓ Red de área local.
- ✓ Cualquier tipo de Navegador de preferencia (Google Chrome).
- ✓ No se requiere de conexión a internet.

4.1.2 PLAN DE DESARROLLO DE SOFTWARE

Introducción

De acuerdo a este plan de desarrollo de software será establecida una versión para la propuesta del sistema para la Clínica Odontológica Ortho Dent; Este documento indicara una visión del sistema propuesto.

El desarrollo del sistema SYS-OD para la “COOD”, maneja la metodología RUP mismas que concederán el uso de normas para especificar el sistema y documentación de la misma.

Esta documentación será de mucha importancia ya que se establece procesos RUP dependiendo de las características del sistema como son: roles, actividades, artefactos (Entregables) como lo es este documento.

Propósito

El propósito de este plan de Desarrollo de Software es el de abastecer la información necesaria para planificar, definir y controlar el desarrollo del sistema SYS-OD para la “COOD”.

Participantes:

- ✓ Analista del proyecto, este define recursos, tiempos, esquemas y herramientas que serán utilizadas en la continuación del sistema.
- ✓ Desarrollador del sistema el cual ejecutara y desarrollara el contenido de sistema.

Alcance

En el Plan de Desarrollo de Software cuenta con un plan global que indica todas las iteraciones y documentos de manera individual; en el artefacto de Visión se indican las principales características del sistema que se desarrollara, por lo que es mucha importancia ya que define cada etapa. Para la versión 1.0 del Plan de Desarrollo de Software, se ha basado en obtener todos los requisitos por parte de la secretaria y administradora de la “COOD”, una vez comenzado este documento se generara la primera versión del artefacto llamado “Visión”, necesario para clarificar a este documento. Durante el proceso el cual se va a desarrollar el sistema, se ajustara este documento generando nuevas versiones actualizadas.

Vista General del Proyecto

Propósito, alcance y objetivos

La información obtenida por la “COOD” fue obtenida mediante las reuniones que se realizaron por parte de la secretaria y la administradora de la empresa, Srta. María José Chasiquiza y Dra. Beatriz Rosero.

La “COOD” como clínica odontológica, busca ofrecer a la ciudadanía una clínica con la más alta calidad en la que todo el servicio de salud oral, donde personal capacitado en el tratamiento de salud oral. La clínica con estas características es previsible a adaptarse a nuevos sistemas digitales y evolución tecnológica. Por consiguiente la “COOD” está dispuesta a usar un sistema integrado de información, por lo que la clínica no cuenta con un sistema digital y solo se maneja la información en fichas, de allí la idea de implementar un sistema con arquitectura moderna y además usando software libre como lo es con MongoDB, y contribuir con el crecimiento y desarrollo de la clínica.

El proyecto definirá la propuesta de desarrollo para los dos módulos y cómo interactúan entre sí para garantizar un buen funcionamiento y cumplir con los objetivos antes mencionados en el documento de “Visión”.

✓ Módulo de Gestión de Usuarios

○ Administrar de Usuarios.

○ Administrar sesiones.

○ Control de Acceso.

✓ Módulo de Facturación

○ Registrar Clientes.

○ Registrar Recibos.

○ Registros de Facturas.

○ Registros de Cotizaciones.

○ Realizar Consultas.

✓ Módulo de Auditoria

○ Analizar Pistas y Reportes en el Tiempo.

✓ Módulo de Reportes

○ Mostrar Reportes.

✓ Módulo de Gestión de Aplicación

○ Gestionar Parámetros.

○ Gestionar Roles.

○ Gestionar Productos.

○ Gestionar Servicios.

Suposiciones y Restricciones

Las suposiciones y restricciones con respecto al sistema SYS-OD en desarrollo y provienen a través de las entrevistas establecidas tanto por la secretaria como la administradora del "COOD":

Se debe examinar las siguientes importancias a estos puntos críticos:

- ✓ Sistema debe ser seguro para proteger la información.
- ✓ Seguridad en la elaboración de canjes como de facturas de clientes.
- ✓ Adaptabilidad y facilidad de uso del sistema.

La lista de restricciones como de supuestos de las dará a conocer en el transcurso del desarrollo del sistema, una vez establecido en el artefacto de "Visión".

Entregables del proyecto

En lo que trata la metodología RUP es generar documentación como entregables en donde se explica las actividades que se han realizado a lo largo del desarrollo del sistema. A continuación se describen los artefactos que se desarrollaran en el proyecto.

En la metodología RUP, cada documento que se crea, se maneja a través de versiones, ya que se puede modificar a lo largo del desarrollo del proyecto.

- ✓ **Visión:** Este documento, presenta el enfoque del cliente indicando las necesidades y características del producto.
- ✓ **Plan de Desarrollo de Software:** El presente documento.
- ✓ **Glosario:** Indica un resumen de términos no conocidos con su respectivo significado.

- ✓ **Modelo de Casos de Uso:** Indica una descripción de pasos que conlleva un proceso en el sistema.
- ✓ **Especificaciones de Casos de Uso:** Es una explicación detallada de los casos de uso, con el uso de un documento plantilla donde se indicara paso a paso que es lo que hace el proceso.
- ✓ **Prototipo de Interfaz de Usuario:** Son modelos gráficos en los que el cliente puede hacerse una idea clara de cómo es lo que se va a desarrollar.
- ✓ **Casos de Prueba:** Indica las condiciones por las que se debe realizar las pruebas a través de instrucciones paso a paso.
- ✓ **Lista de Riesgos:** Contiene una lista de riesgos de acuerdo a su importancia y acciones específicas para disminuir el riesgo.
- ✓ **Manual de Instalación:** Este documento indicara una guía de pasos para la instalación del sistema.
- ✓ **Producto:** El sistema se otorgara en un DVD con la documentación y mecanismos para su instalación.
- ✓ **Actas de trabajo:** Contienen actividades, responsabilidades, observaciones y revisiones que se desenvuelven en el desarrollo del sistema

Participantes del proyecto

La "COOD" brindara los diferentes requisitos del sistema SYS-OD para evaluarlos y documentarlos en los artefactos.

Roles y responsabilidades

En la siguiente tabla se indican las responsabilidades de cada uno de los personajes en el equipo de desarrollo en las etapas de Inicio y Elaboración, de acuerdo a la metodología RUP.

TABLA 54: Roles y Responsabilidades

Puesto	Responsabilidad
Analista	Obtiene las especificaciones y evalúa las validaciones de requisitos, interactuando con los usuarios a través de las entrevistas. Ayuda también en las pruebas y modelos de datos.
Programador	Es responsable del desarrollo del sistema, creando los módulos, versiones hasta finalizar con una versión estable.

Fuente: El autor

Gestión del Proceso

Indican las actividades a realizar en cada fase de la metodología de desarrollo RUP.

Plan del Proyecto

Se presenta las diferentes fases e iteraciones de la metodología antes mencionada.

Plan de Fases

En el desarrollo del sistema SYS-OD, se maneja fases establecidas con una o varias iteraciones; en la siguiente tabla se muestra una aproximación del tiempo e iteraciones de cada fase.

TABLA 55: Plan de Fases

Fase	Nro. de iteraciones	Duración
Fase de Inicio	1	3 semanas
Fase de Elaboración	1	3 semanas
Fase de Construcción	2	10 semanas
Fase de Transición	1	4 semanas

Fuente: El autor

A continuación en la siguiente tabla se indicara las diferentes iteraciones de cada fase que serán desarrolladas.

TABLA 56: Objetivos de Iteraciones

Fase	Iteración	Descripción	Hitos Asociados	Objetivos
Incepción	Iteración inicial	Definición del modelo de negocio y plan del proyecto.	Actas de trabajo documento de Visión.	Clasificar requerimientos del usuario, alcance, factibilidad.
Elaboración	Desarrollar prototipo.	Desarrollar prototipo inicial, casos de uso.	Documento de Arquitectura	Definir arquitectura y herramientas.
Construcción	Desarrollar versión beta.	Implementar y probar casos de uso.	Versión beta	Implementación de las características claves.
	Desarrollar reléase inicial.	Corregir errores y observaciones de la versión beta.	Producto (Software)	Revisión de los usuarios finales, sistema de alta calidad.
	Desarrollo del sistema completo	Mejorar errores y defectos.	Producto (Software)	Funcionalidad completa del sistema para liberación.
Transición	Liberar el software	Empaquetar e instalar el software	Software final	Sistema en producción.

Fuente: El autor

Calendario del proyecto

A continuación se indica un calendario de las tareas que se dividen en esta metodología de desarrollo RUP; los artefactos obtenidos por esta metodología a lo largo de la documentación del sistema van desarrollándose en mayor o menor grado dependiendo de la fase en la que se encuentre. En la siguiente ilustración se muestra el rango de cada fase dependiendo de su complejidad.

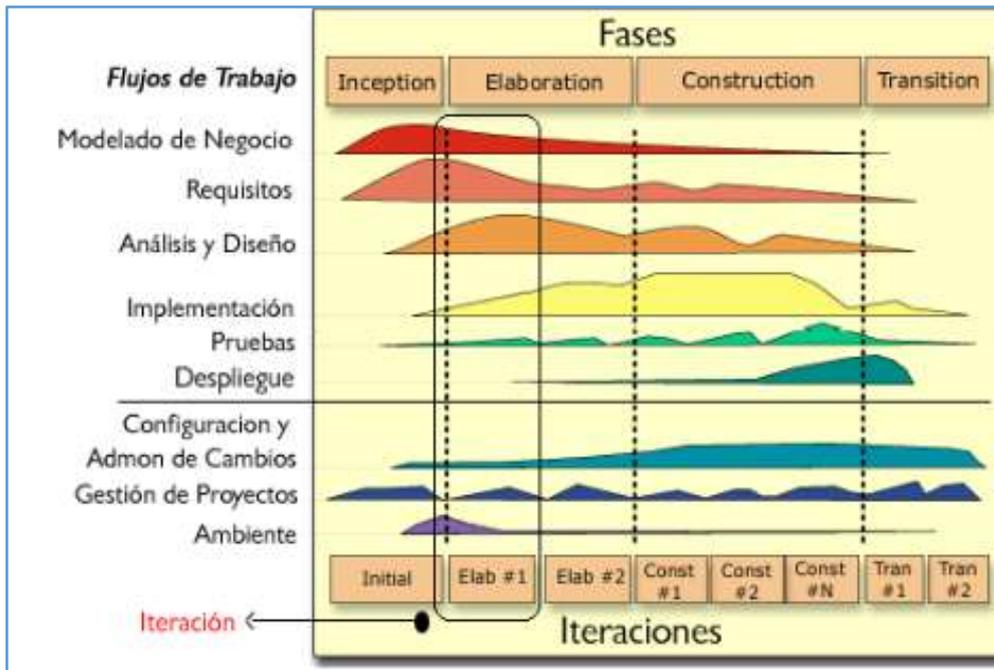


ILUSTRACIÓN 20: Calendario de las tareas de la Metodología de Desarrollo RUP

Fuente: Metodología RUP

Seguimiento y control del proyecto

Gestión de Requisitos

Los requisitos que el sistema necesita, están indicados en el artefacto de “Visión” donde cada atributo tendrá mucha importancia y permitirá realizar un efectivo seguimiento.

Control de Plazos

Se realizara una evaluación y seguimiento a cada punto principal del cronograma a través del jefe de proyecto, como la persona encargada.

Control de Calidad

Los defectos si es que el sistema se le encuentra alguno, también tendrán un seguimiento para asegurar la efectividad del sistema por lo que se desarrollaran listas y guías de revisión que mantiene la metodología RUP.

Gestión de Riesgos

Desde la fase Inicial, se creara una lista de riesgos y acciones para disminuirlos en el tiempo y tomar las acciones debidas.

Gestión de Configuración

Se realizara la gestión y configuración para el almacenamiento de registros de los artefactos con sus versiones; se agregará también las solicitudes de cambio con sus versiones.

4.1.3 ACTAS DE TRABAJO

En las actas de trabajo se registra todos los requerimientos en cada una de las reuniones establecidas en la Clínica Odontológica Ortho Dent para el levantamiento de la información y procesos.

Acta de Trabajo Nro. 1

Análisis, Requerimientos y Recopilación de información para sistema a desarrollar



UNIVERSIDAD TÉCNICA DEL NORTE
Principal: Av. 17 de Julio – Ciudadela Universitaria
Ibarra - Ecuador
Teléfono: (593 6) 2997800 ext. 7188
OE - UTN Oficina del Estudiante
Web: www.utn.edu.ec

ACTA DE TRABAJO No. 1		
Proyecto: Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent.		
Tema a tratar: Análisis, Requerimientos y Recopilación de información para sistema a desarrollar.		
Fecha: 20/Octubre/2014		
Participantes:		
Nombre	Cargo	Firma
Luis Correa Real	Tesista	
María José Chasiquiza	Secretaria de clínica	
Temas tratados:		
<p>A las 10:00 horas del día Lunes 20 de Octubre del 2014, en la Clínica Odontológica Ortho Dent, se procede a realizar una reunión con los involucrados, para analizar requerimientos y recopilar información, el mismo que se sugiere para el desarrollo del sistema.</p> <ol style="list-style-type: none">Análisis del sistema: Se procedió a la toma de modelos necesarios para realizar el bosquejo de la aplicación que se lo presentara en una página web.Recopilación de la información: Se indicó las diferentes características que cada módulo debe tener para un sistema que ayude a la Clínica Odontológica Ortho Dent.		
Observaciones adicionales y compromisos adquiridos:		
<ul style="list-style-type: none">• Tomar en cuenta que aspectos necesitamos para poder cumplir con un requerimiento.• Adjunto Anexo (Bosquejo inicial del sistema a desarrollar).• En el sistema no se agregara el módulo de impresión de facturas.		

Acta de Trabajo Nro. 2

Requerimientos y análisis de los módulos del sistema



UNIVERSIDAD TÉCNICA DEL NORTE
Principal: Av. 17 de Julio - Ciudadela Universitaria
Ibarra - Ecuador
Teléfono: (593 6) 2997800 ext. 7188
OE - UTN Oficina del Estudiante
Web: www.utn.edu.ec

ACTA DE TRABAJO No. 2		
Proyecto: Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent.		
Tema a tratar: Requerimientos y análisis de la Gestión de Clientes y Registro de Pagos.		
Fecha: 25/Octubre/2014		
Participantes:		
Nombre	Cargo	Firma
Luis Correa Real	Tesista	
María José Chasiquiza	Secretaria de clínica	
Temas tratados:		
<p>A las 10:00 horas del día Sábado 25 de Octubre del 2014, en la Clínica Odontológica Ortho Dent, se procede a realizar una reunión con los involucrados, para analizar requerimientos y recopilar información de los diferentes módulos, el mismo que se sugiere para el desarrollo del sistema.</p> <ol style="list-style-type: none">Revisión del prototipo del sistema: Se procedió a la revisión del prototipo que se realizó con anterioridad por lo que se modificó en varias ocasiones hasta llegar a un modelo aceptable para los usuarios.Obtención de requerimientos funcionales del sistema: Se indicó los diferentes requerimientos que debe tener cada página con su respectivo modulo.		
Observaciones adicionales y compromisos adquiridos:		
<ul style="list-style-type: none">Tomar en cuenta que aspectos necesitamos para poder cumplir con un requerimiento.Distintos requerimientos y limitantes para el desarrollo del sistema.		

Acta de Trabajo Nro. 3

Requerimientos de los campos para los módulos del sistema



UNIVERSIDAD TÉCNICA DEL NORTE
Principal: Av. 17 de Julio - Ciudadela Universitaria
Ibarra - Ecuador
Teléfono: (593 6) 2997800 ext. 7188
OE - UTN Oficina del Estudiante
Web: www.utn.edu.ec

ACTA DE TRABAJO No. 3		
Proyecto: Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent.		
Tema a tratar: Requerimientos de los campos para la Gestión de Clientes, Registro de Pagos y Auditoría.		
Fecha: 06/Noviembre/2014		
Participantes:		
Nombre	Cargo	Firma
Luis Correa Real	Tesista	
María José Chasiquiza	Secretaria de clínica	
Temas tratados:		
<p>A las 12:00 horas del día Jueves 06 de Noviembre del 2014, en la Clínica Odontológica Ortho Dent, se procede a realizar una reunión con los involucrados, para obtener requerimientos de los campos para los diferentes módulos, el mismo que se sugiere para el desarrollo del sistema.</p> <ol style="list-style-type: none">1. Obtención de campos de las diferentes entidades: Se procedió a la tomar la información que se requiera para realizar el modelo de base de datos para el desarrollo de la aplicación como es la información que llevara clientes, facturas e historiales.2. Análisis de los diferentes módulos: Se analizó si se podrá agregar la opción de impresión de facturación como las diferentes opciones que tendrá el sistema, así como la lista de productos para la clínica.		
Observaciones adicionales y compromisos adquiridos:		
<ul style="list-style-type: none">• Tomar en cuenta que aspectos necesitamos para poder cumplir con un requerimiento.		

Acta de Trabajo Nro. 4

Análisis de modelo de la aplicación desarrollada.



UNIVERSIDAD TÉCNICA DEL NORTE
Principal: Av. 17 de Julio - Ciudadela Universitaria
Ibarra - Ecuador
Teléfono: (593 6) 2997800 ext. 7188
OE - UTN Oficina del Estudiante
Web: www.utn.edu.ec

ACTA DE TRABAJO No. 4		
Proyecto: Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent.		
Tema a tratar: Análisis de modelo de la aplicación a desarrollar.		
Fecha: 20/Noviembre/2014		
Participantes:		
Nombre	Cargo	Firma
Luis Correa Real	Tesista	
María José Chasiquiza	Secretaría de clínica	
Temas tratados:		
<p>A las 10:00 horas del día Jueves 20 de Noviembre del 2014, en la Clínica Odontológica Ortho Dent, se procede a realizar una reunión con los involucrados, para obtener requerimientos de los campos para los diferentes módulos, el mismo que se sugiere para el desarrollo del sistema.</p> <p>1. Análisis de modelo de la página web: Se procedió a la tomar la información que se requerirá para escoger el modelo de formato que tendrá la página web que se desarrollara para esta empresa, se procedió a escoger un formato nombrado MetroUI a través de la página de PrimeFaces</p> <p>Página que se utilizó para escoger el modelo:</p> <p>http://www.primefaces.org/showcase/ui/misc/themeSwitcher.xhtml</p>		
Observaciones adicionales y compromisos adquiridos:		
<ul style="list-style-type: none">• Una vez escogido el modelo de página web no se podrá realizar un cambio de dicho modelo.• Se comenzara a realizar la aplicación para entrega del sistema.		

Acta de Trabajo Nro. 5

Cambios mínimos de la aplicación desarrollada.



UNIVERSIDAD TÉCNICA DEL NORTE
Principal: Av. 17 de Julio – Ciudadela Universitaria
Ibarra - Ecuador
Teléfono: (593 6) 2997800 ext. 7188
OE - UTN Oficina del Estudiante
Web: www.utn.edu.ec

ACTA DE TRABAJO No. 5		
Proyecto: Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent.		
Tema a tratar: Primera vista del modelo de aplicación del sistema.		
Fecha: 12/Enero/2015		
Participantes:		
Nombre	Cargo	Firma
Luis Correa Real	Tesista	
María José Chasiquiza	Secretaria de clínica	
Temas tratados:		
<p>A las 10:00 horas del día Lunes 12 de Enero del 2015, en la Clínica Odontológica Ortho Dent, se procede a realizar una reunión con los involucrados, se presenta como se encuentra el sistema para los diferentes módulos, el mismo que se sugiere para la presentación del sistema.</p> <ol style="list-style-type: none">Análisis de los módulos de la página web: Se procedió a la revisar los diferentes módulos que se mencionaron anteriormente, que es lo que contienen y si se encuentra bien diseñado como se indicó en los requisitos anteriores.Vista de las páginas del sistema: Se indicó las diferentes páginas que en sistema cuenta con los cinco módulos que se desarrollaron para el desarrollo de la aplicación.Obtención de requisitos en caso de encontrarse con alguno: Se realizó el estudio de cada elemento de la aplicación por si se encontraba con algún problema.		
Observaciones adicionales y compromisos adquiridos:		
<ul style="list-style-type: none">Se obtuvo con esta acta que se encuentra el sistema en buen estado para la aplicación en la Clínica Odontológica Ortho Dent.Se desarrollaron cambios mínimos en el sistema, por lo que se continuará realizando la aplicación tal como se estuvo realizando.		

4.2 FASE DE ELABORACIÓN

En esta fase se realizara el diseño básico de la arquitectura con sus casos de uso respectivos para implementarla al sistema.

4.2.1 INTRODUCCIÓN

A continuación se presenta el documento de arquitectura que es generado por la Metodología de Desarrollo RUP.

Arquitectura del Sistema

La arquitectura del sistema SYS-OD para la “COOD” abarcara elementos importantes de la estructuración de dicha empresa tales como el rendimiento, adaptación, reutilización, compresión del sistema y restricciones.

Propósito

El presente documento indica un panorama de la arquitectura del sistema “SYS-OD” para la “COOD”, con sus respectivos glosarios como es los objetivos, restricciones, casos de uso, patrones de arquitectura que se aplicaron.

Alcance

Se definió los componentes arquitectónicos del sistema para así obtener una visión clara de cómo se desarrollara el sistema.

Representación de la Arquitectura

En este documento se desarrolló una sucesión de vistas, casos de uso, procesos, despliegue del sistema e implementación. Estos modelos se desarrollaron a través de Cacco, diagramador Draw de Outlook, UML, Excel, JDeveloper, ArgoUML.

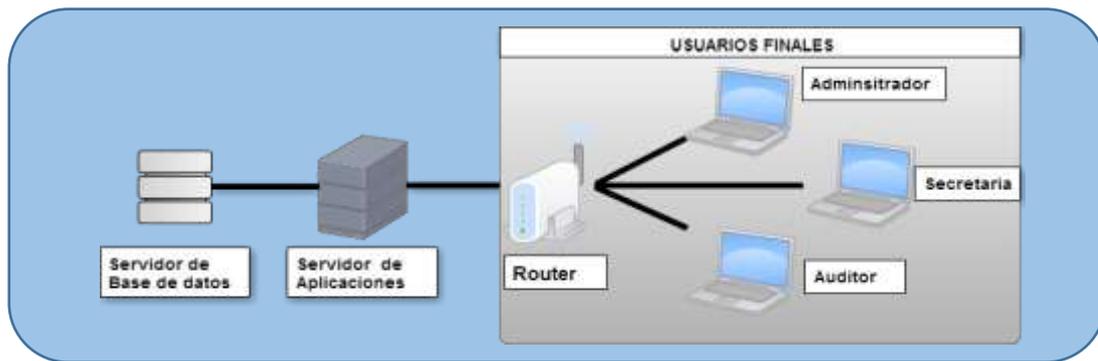


ILUSTRACIÓN 21: Arquitectura del Sistema

Fuente: El autor

En la ilustración se detalla cómo se organiza el sistema SYS-OD para la “COOD”, y como cada uno de los elementos se relacionan entre sí; por el lado de la vista se encuentran los diferentes usuarios que van a hacer uso del sistema a través de un navegador web, al digital el respectivo URL del sistema, permitirá conectarse con el servidor de aplicaciones y así contiguamente con el servidor de base de datos como es MongoDB dependiendo del rol que realizó la petición.

Objetivos y Restricciones de la arquitectura

Existen diferentes requerimientos como restricciones importantes para definir la arquitectura como son:

- ✓ Se diseñó a través de capas para aclarar ciertas partes importantes.
- ✓ Capas reutilizables.
- ✓ Todos estos requerimientos se los debe tomar en consideración para un buen desarrollo de la arquitectura.

4.2.2 VISTA DE CASOS DE USO

Los casos de uso y los procesos de cada módulo nos ayudaran a observar claramente cómo es el funcionamiento de la empresa y así poder desarrollar un sistema que cumpla sus expectativas.

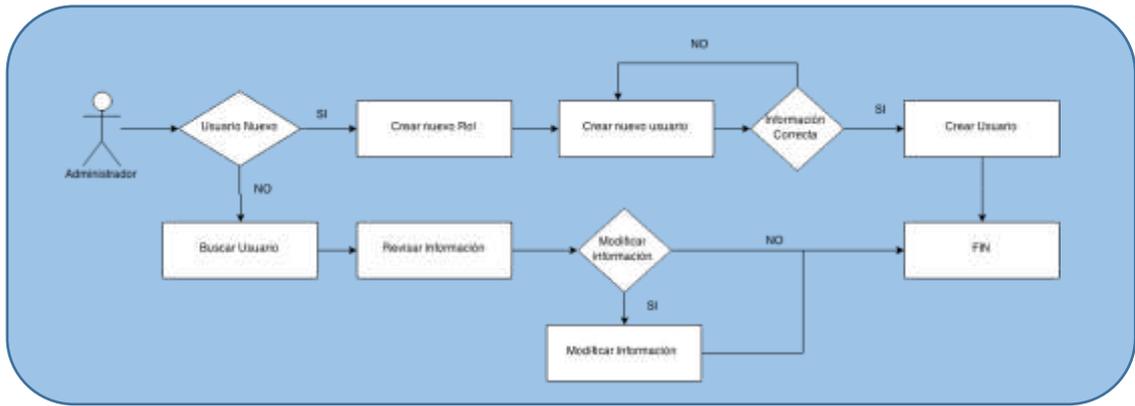
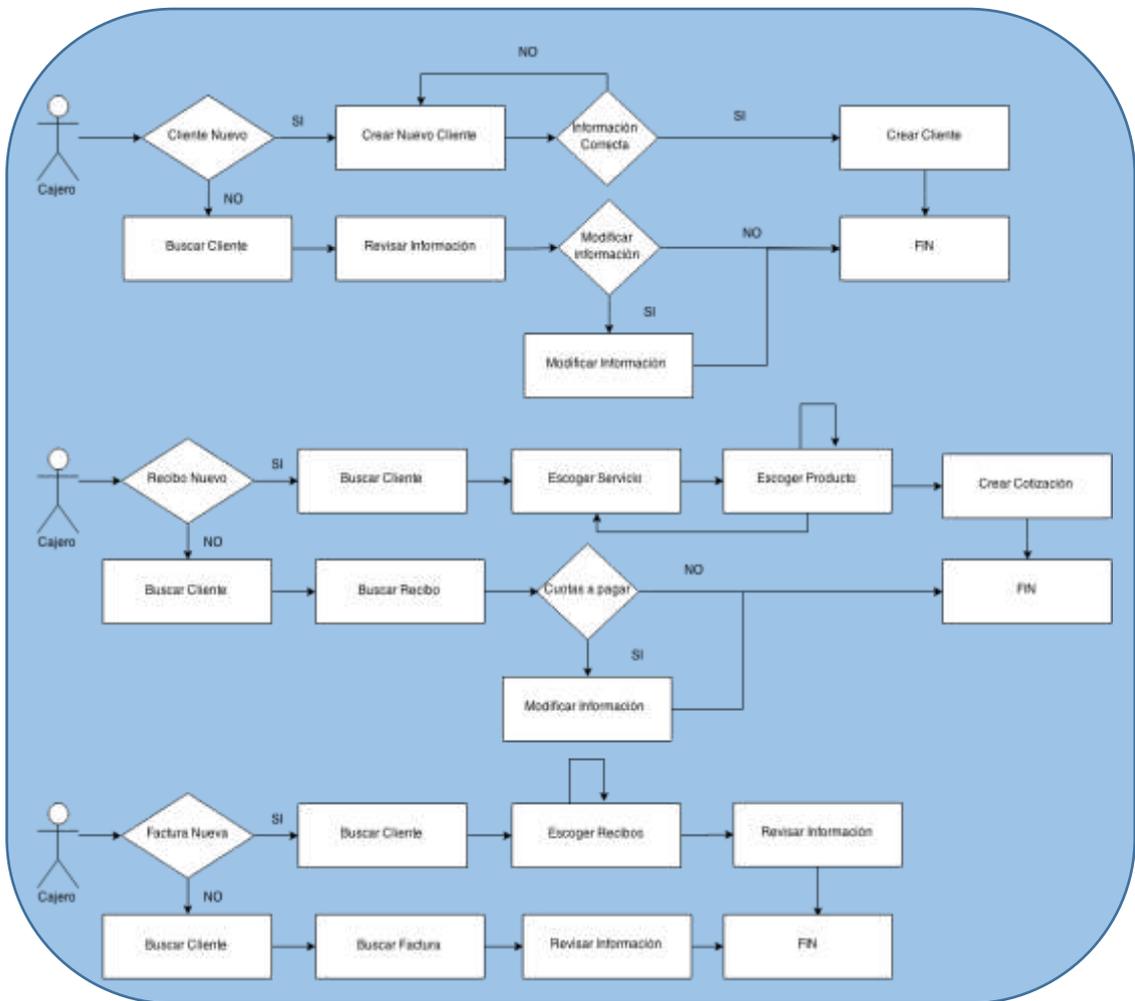


ILUSTRACIÓN 22: Procesos Módulo de Gestión de Usuarios

Fuente: El autor



ILUSTRACION 23: Procesos Modulo de facturación

Fuente: El autor

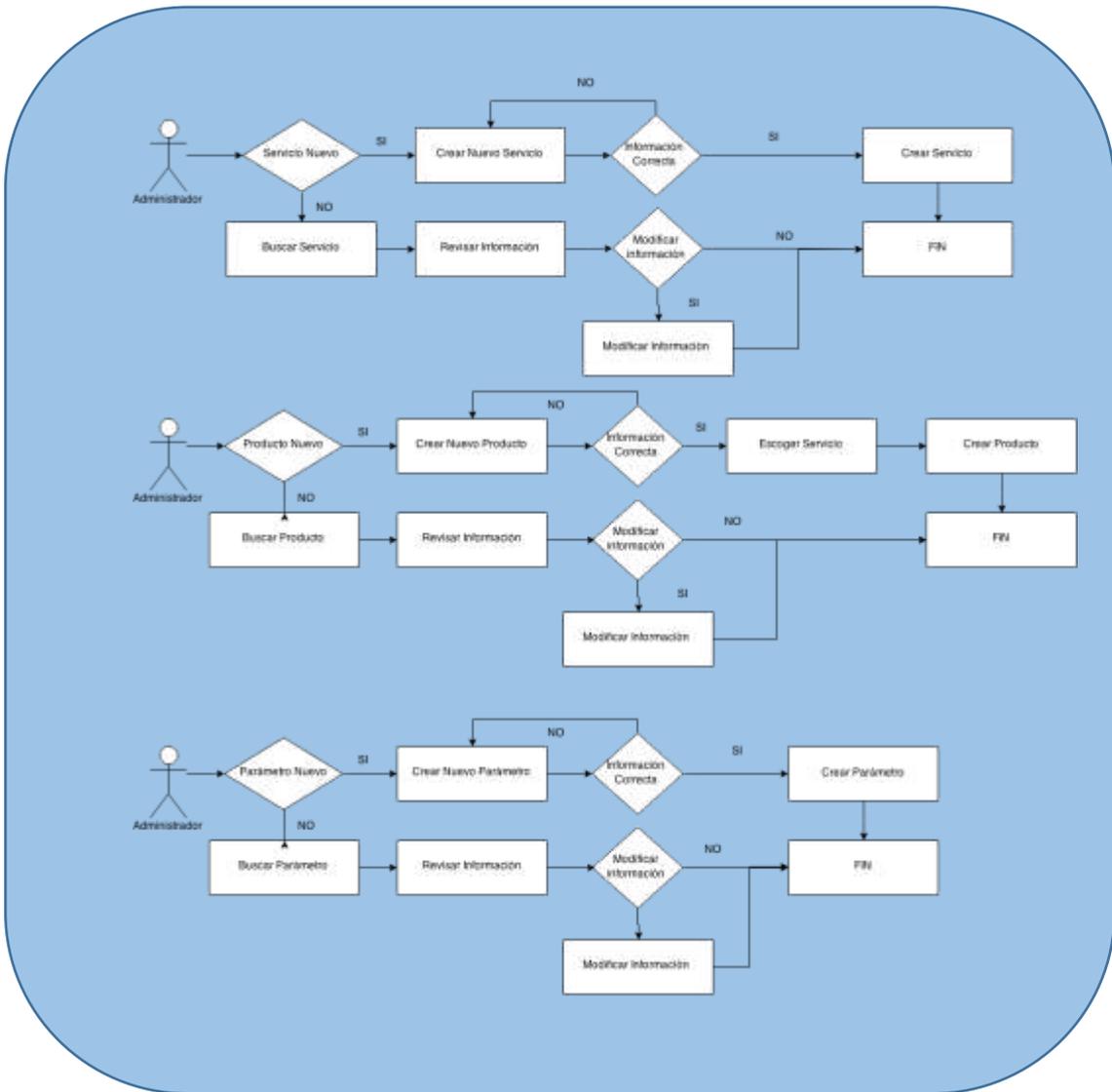


ILUSTRACIÓN 24: Procesos Módulo de Gestión de Aplicación

Fuente: El autor

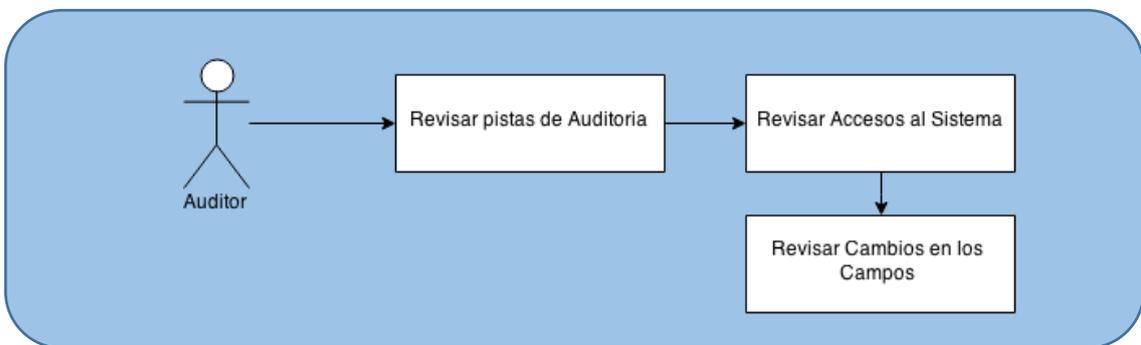


ILUSTRACIÓN 25: Procesos Módulo de Auditoría

Fuente: El autor

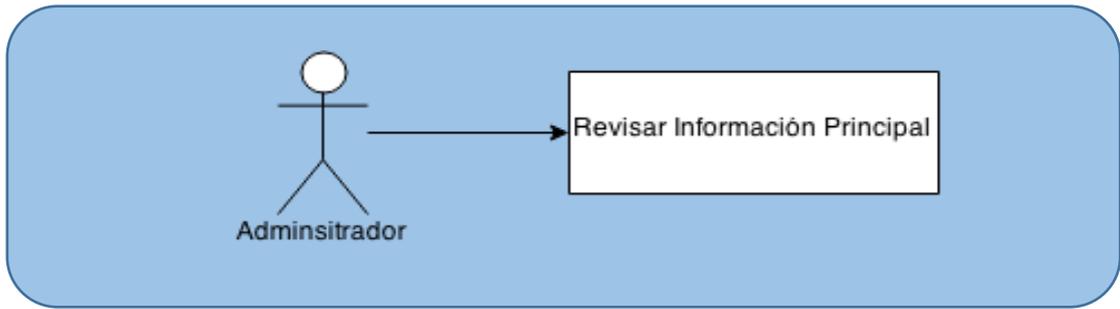


ILUSTRACIÓN 26: Procesos Módulo de Reportes

Fuente: El autor

Actores

Son los diferentes usuarios que van a hacer uso del sistema de la “COOD”

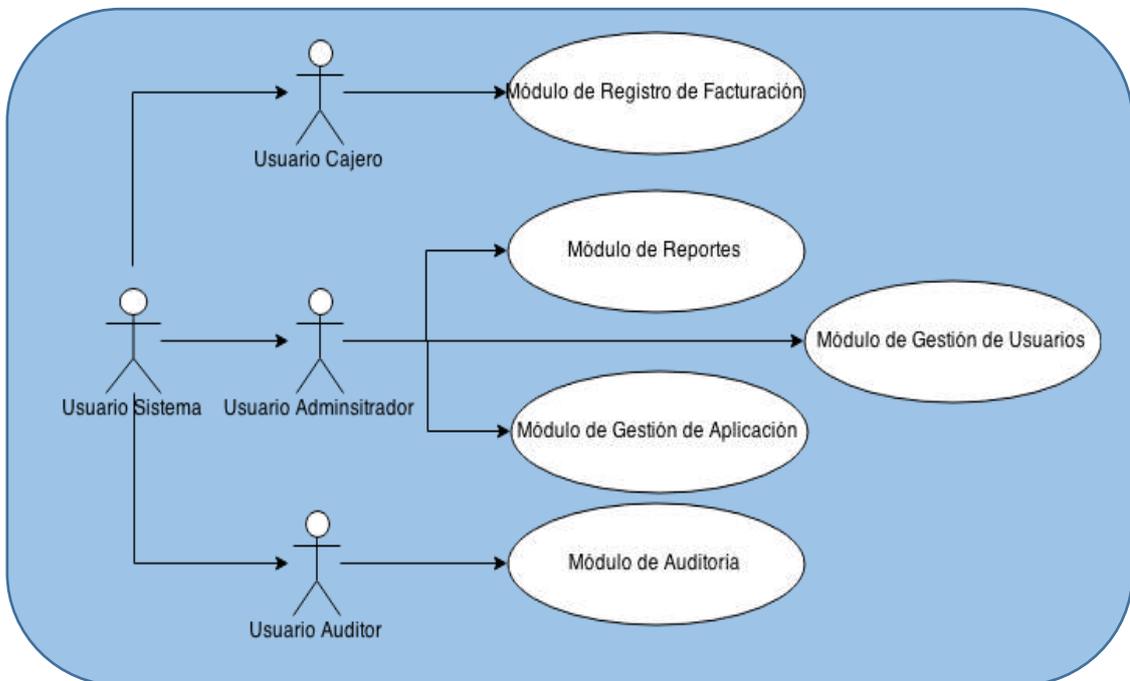


ILUSTRACIÓN 27: Actores del Sistema

Fuente: El autor

TABLA 57: Descripción de Usuarios

Nro.	Stakeholder	Descripción
1	Usuario	El usuario general del sistema podrá ingresar requerimientos y consultar en la base de datos.
2	Usuario Administrador	Administrador del sistema podrá acceder a toda la información del sistema, tanto información de cliente como de los pagos de cada cliente.
3	Usuario Cajero	Se encargara de realizar toda la Gestión de Clientes y Registro de Facturación.
4	Usuario Auditor	Se encarga de la revisión y control de las diferentes modificaciones en el sistema.

Fuente: El autor

El modelo de casos de negocios indica la visión general del sistema desde el punto de vista del negocio

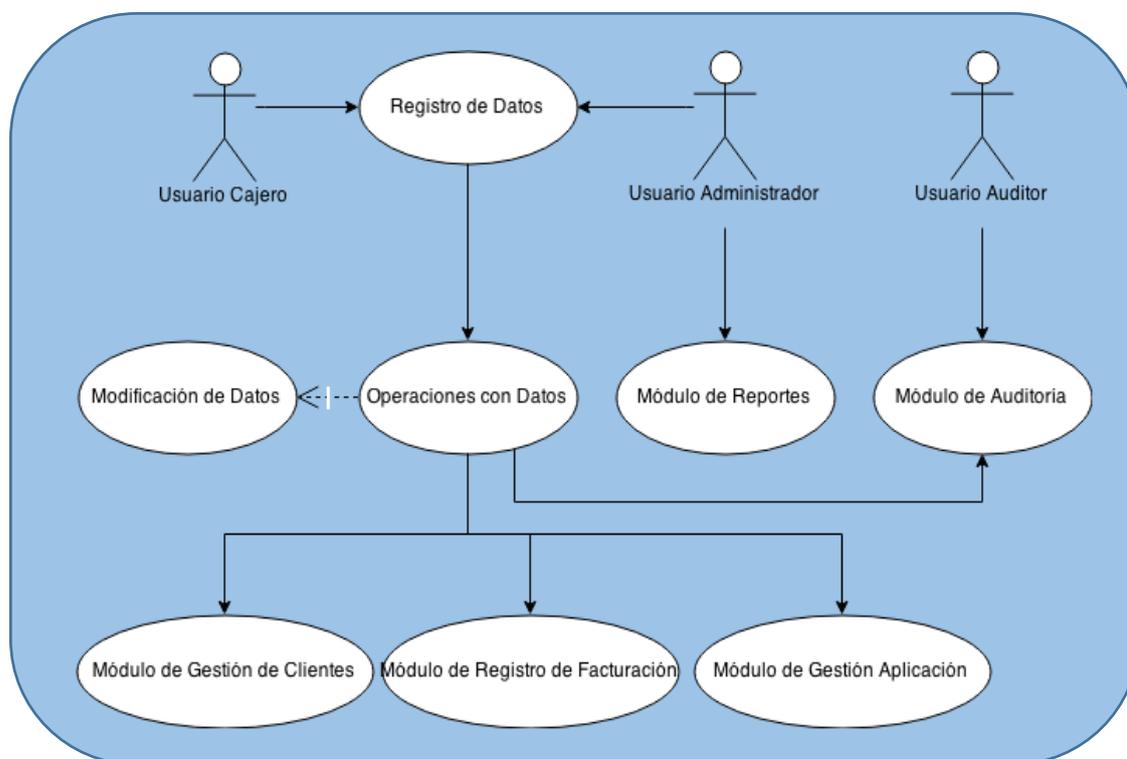


ILUSTRACIÓN 28: Modelos de Casos de Negocios

Fuente: El autor

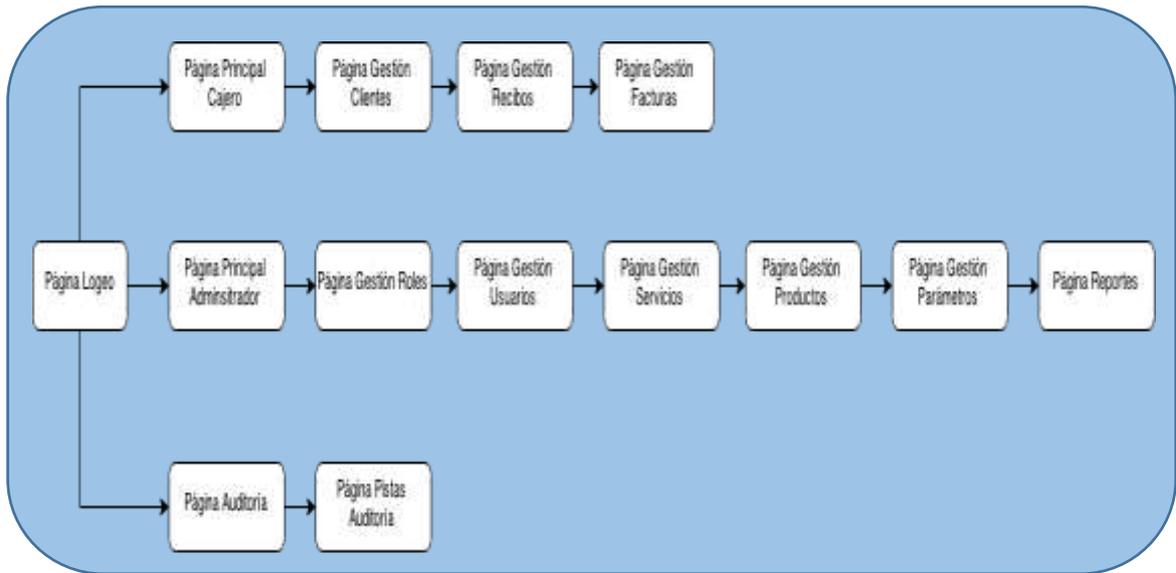


ILUSTRACIÓN 29: Diagrama de Flujo de Páginas

Fuente: El autor

Modelos de Casos de Uso

A continuación se presentaran los casos de uso para cada módulo del sistema SYS-OD para “COOD”

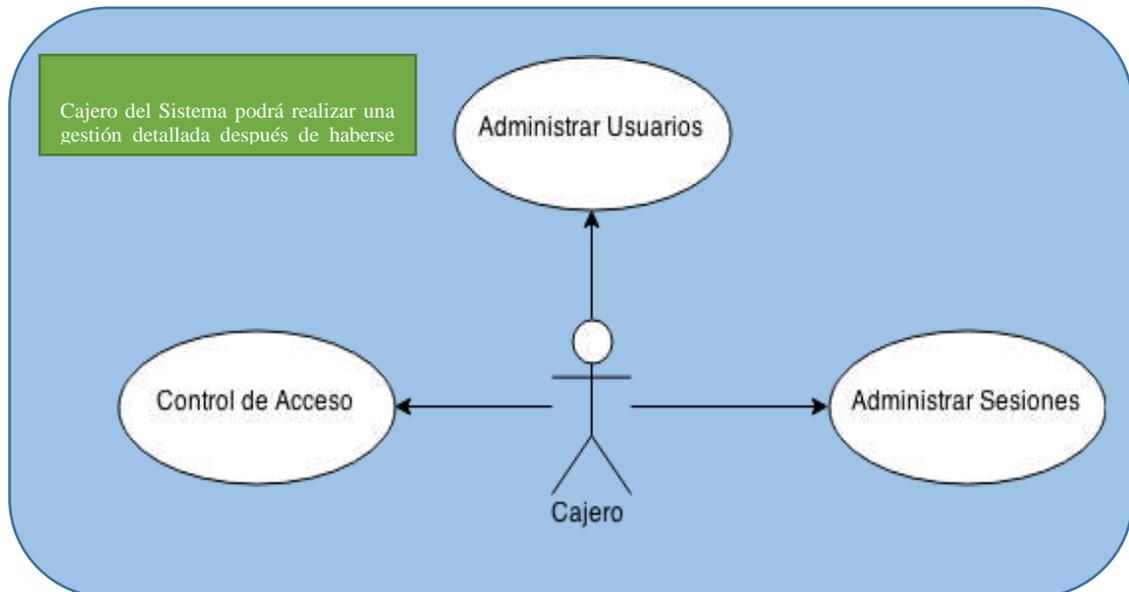


ILUSTRACIÓN 30: Modelo de Caso de Uso Módulo de Gestión de Usuarios

Fuente: El autor

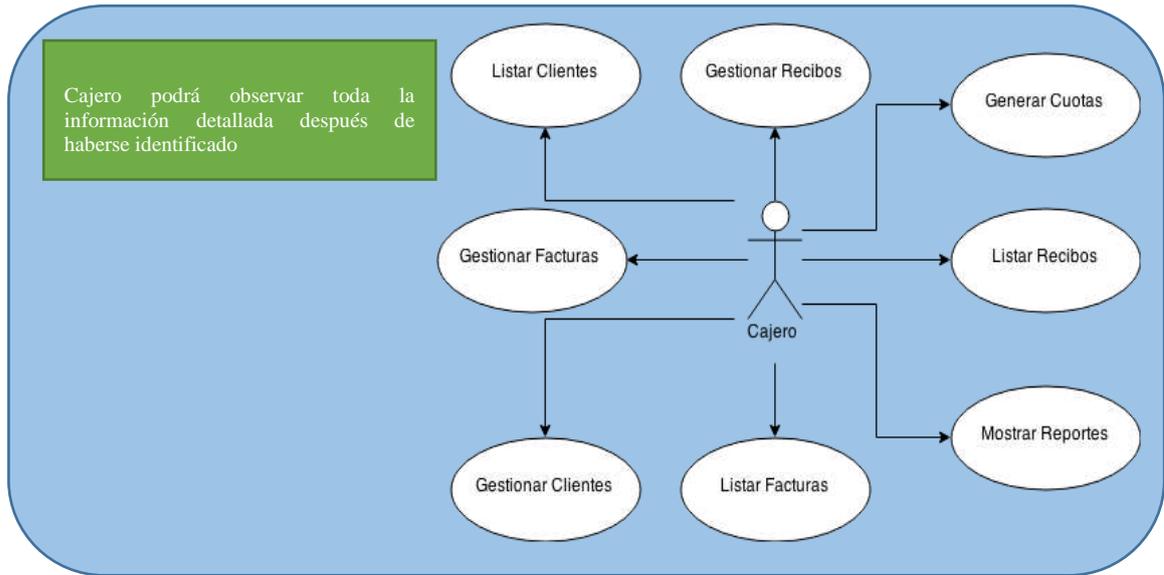


ILUSTRACIÓN 31: Modelo de Caso de Uso Módulo de Facturación

Fuente: El autor

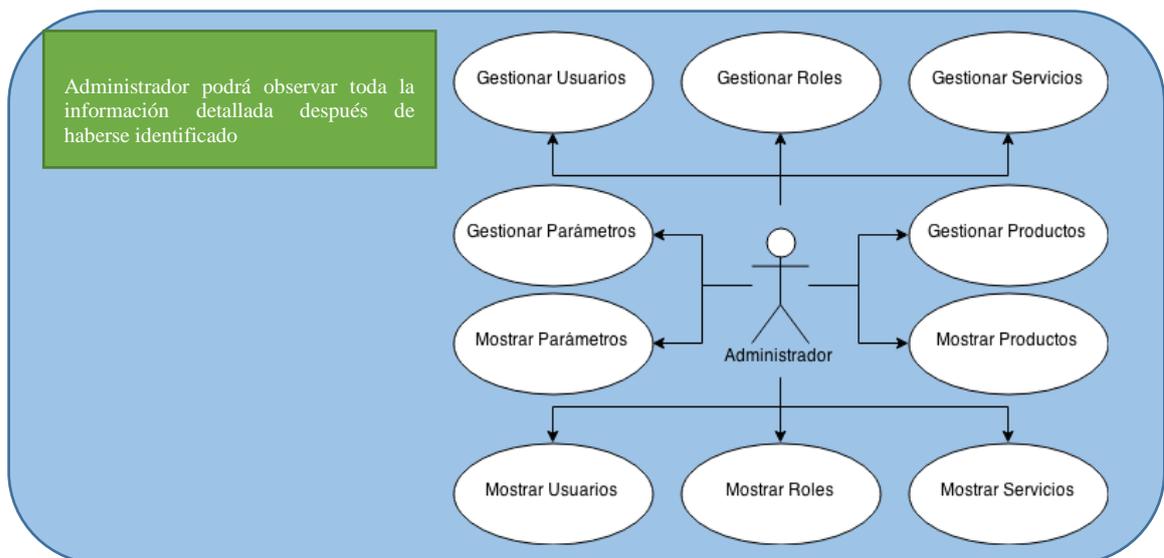


ILUSTRACIÓN 32: Modelo de Caso de Uso Módulo de Gestión de Aplicación

Fuente: El autor



ILUSTRACIÓN 33: Modelo de Caso de Uso Módulo de Auditoría

Fuente: El autor

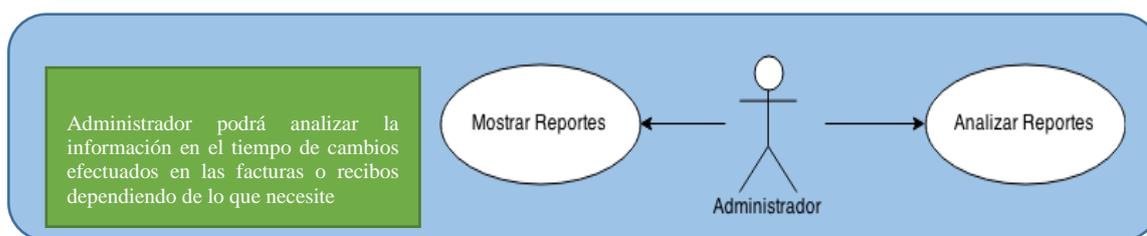


ILUSTRACIÓN 34: Modelo de Caso de Uso Módulo de Reportes

Fuente: El autor

A continuación se observara las prioridades de cada módulo con sus respectivos procesos.

TABLA 58: Prioridad de los Casos de Uso

Caso de Uso	Prioridad de Negocio	Prioridad Técnica
1. Módulo de Gestión de Usuarios		
Administrar Usuarios	Alta	Alta
Administrar sesiones	Media	Media
Control de Acceso	Media	Media
2. Módulo de Facturación		
Registrar Clientes	Alta	Alta
Registrar Recibos	Alta	Alta
Registrar Facturas	Alta	Alta
Registrar Cotizaciones	Alta	Alta
Realizar Consultas	Media	Baja
3. Módulo de Auditoría		
Analizar Pistas y Reportes en el Tiempo	Alta	Alta
4. Módulo de Reportes		
Mostrar Reportes	Alta	Alta
5. Módulo de Gestión de Aplicación		
Gestionar Parámetros	Alta	Alta
Gestionar Roles	Media	Baja
Gestionar Productos	Media	Baja
Gestionar Servicios	Media	Baja

Fuente: El autor

4.2.3 DESCRIPCIÓN DE LOS CASOS DE USO

A continuación se indicara detalladamente como se describe cada caso de uso de cada módulo del sistema SYS-OD.

Descripción del Módulo de Gestión de Usuarios

TABLA 59: Descripción Administrar Usuarios

Caso de Uso	Administrar Usuarios
Resumen	Permite al administrador la búsqueda y gestión de usuarios, usuarios nuevos que desee hacer uso del sistema "COOD" a través de su respectivo Login y password, el usuario se le considera privilegios del sistema.
Actor	Administrador
Precondición	"SYS-OD" se encuentra en espera a nueva transacción. Una página de logeo para el administrador.
Descripción	<p>Inicia cuando el administrador intenta ingresar al sistema con su respectivo usuario y contraseña.</p> <ol style="list-style-type: none">1.- El sistema solicita el usuario y contraseña del administrador.2.- El administrador digita su usuario y contraseña.3.- El sistema lee el usuario y contraseña del administrador.4.- El sistema verifica la validez del administrador.5.- El sistema solicita al administrador que se dirija a menú gestión de la aplicación para realizar alguna acción.6.- Selecciona en crear o gestionar usuario o roles en el menú antes mencionado.7.- Realiza una transacción y se almacena la información. <p>Fin administración usuario</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none">1.- Usuario o contraseña no validas error.2.- El caso de uso finaliza.
Postcondición	La transacción es almacenada por el sistema.

Fuente: El autor

Usuario	Rol	Rol	Acciones
administrador	ADMIN	Activo	Options
auditor	AUDITOR	Activo	Options
cajero	CAJERO	Activo	Options

ILUSTRACIÓN 35: Administrar Usuarios

Fuente: El autor

TABLA 60: Descripción Administrar Sesiones

Caso de Uso	Administrar Sesiones
Resumen	Permite al administrador gestionar la información de quienes son los usuarios que ingresan al sistema “SYS-OD” de la “COOD”.
Actor	Administrador, Auditor
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página de logeo para usuarios.
Descripción	<p>Inicia cuando el usuario intenta ingresar al sistema con su respectivo usuario y contraseña.</p> <ol style="list-style-type: none"> 1.- El sistema solicita el usuario y contraseña del usuario. 2.- El administrador digita su usuario y contraseña. 3.- El sistema lee el usuario y contraseña del usuario. 4.- El sistema verifica la validez del usuario. <p>Fin administración sesión</p> <p>Trayectoria alternativa.</p> <ol style="list-style-type: none"> 1.- Usuario o contraseña no validas error. 2.- El caso de uso finaliza.
Postcondición	La transacción indica que el usuario se dirigirá a la página principal pertinente a su rol.

Fuente: El autor

Rol	Rol	Usuarios	Acciones
ADMIN	administra toda la información	Ver	Options
CAJERO	Administra la caja	Ver	Options
AUDITOR	audita la información	Ver	Options

ILUSTRACIÓN 36: Administrar Sesiones

Fuente: El autor

TABLA 61: Descripción Control de Acceso

Caso de Uso	Control de Acceso
Resumen	Permite al administrador el control total de los usuarios, modificaciones y parámetros del sistema “SYS-OD” de la “COOD” en el caso de que exista algún cambio.
Actor	Administrador, Auditor
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página de logeo para el administrador o auditor.
Descripción	<p>Inicia cuando el usuario intenta ingresar al sistema con su respectivo usuario y contraseña.</p> <ol style="list-style-type: none"> 1.- El sistema solicita el usuario y contraseña del usuario. 2.- El administrador digita su usuario y contraseña. 3.- El sistema lee el usuario y contraseña del usuario. 4.- El sistema verifica la validez del usuario. 5.- El módulo de auditoria almacena los datos del usuario. <p>Fin control de acceso</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- Usuario o contraseña no validas error 2.- El caso de uso finaliza
Postcondición	La transacción es almacenada por el módulo de auditoria como intento de ingreso.

Fuente: El autor

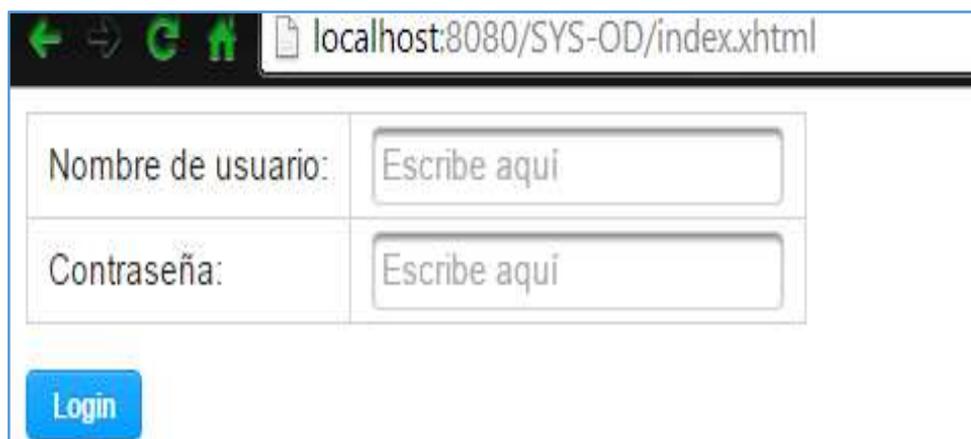


ILUSTRACIÓN 37: Control de Acceso

Fuente: El autor

Descripción del Módulo Registro de Facturación

TABLA 62: Descripción Gestionar Clientes

Caso de Uso	Gestionar Clientes
Resumen	Permite al cajero la gestión de clientes que ingresa a la "COOD", observar sus datos principales.
Actor	Cajero
Precondición	"SYS-OD" se encuentra en espera a nueva transacción. Una página principal de cajero.
Descripción	<p>Inicia cuando el cajero intenta gestionar un cliente.</p> <ol style="list-style-type: none"> 1.- Una vez logeado en el sistema, se mostrara la información de los clientes. 2.- Podrá ingresar un nuevo cliente con los datos respectivos. 3.- Se crea un cliente. <p>Fin Gestión cliente</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- Datos inválidos. 2.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

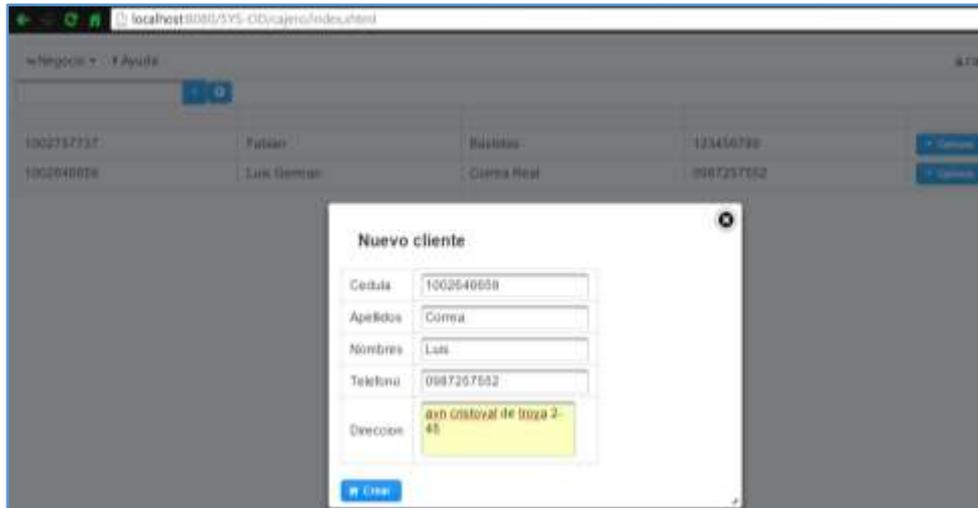


ILUSTRACIÓN 38: Registrar Clientes

Fuente: El autor

TABLA 63: Descripción Registrar Recibos

Caso de Uso	Registrar Recibos
Resumen	Permite al cajero la gestión de recibos de cada cliente que ingresa a la “COOD”, observar sus cuotas y pagos respectivos en las diferentes fechas.
Actor	Cajero
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal de cajero.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- Se dirigirá a crear o ver los recibos de un cliente. 2.- Buscara el cliente pertinente, agregará el tipo de tratamiento, ingresara los diferentes productos y se creará un recibo con cuotas dependiendo del monto. 3.- Se crea un recibo. <p>Fin Registrar Recibos</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- registro del recibo mal digitado. 2.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

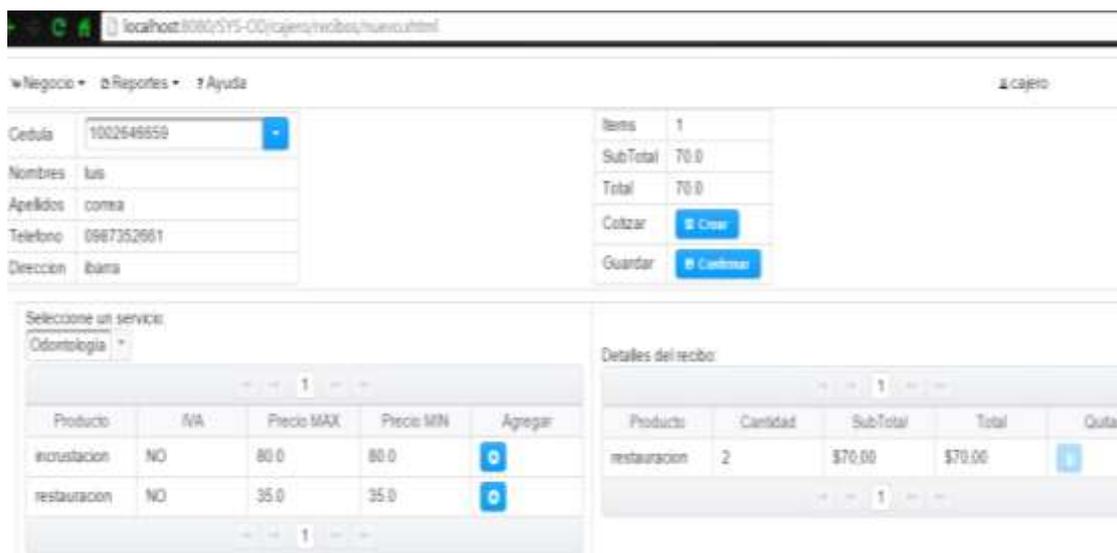


ILUSTRACIÓN 39: Registrar Recibos

Fuente: El autor

TABLA 64: Descripción Registrar Facturas

Caso de Uso	Registrar Facturas
Resumen	Permite al cajero la gestión de facturas de cada cliente, a través de recibos se generara una factura de algún tratamiento que se ha desarrollado en la "COOD".
Actor	Cajero
Precondición	"SYS-OD" se encuentra en espera a nueva transacción. Una página principal del cajero.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- El cajero ingresa a la página principal donde se mostrara la información del cliente. 2.- Se dirigirá a crear o ver las facturas de dicho cliente. 3.- Buscara el cliente pertinente, agregará los recibos que se necesite. 4.- Se crea una factura. <p>Fin Registrar Facturas</p>
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor



ILUSTRACIÓN 40: Registrar Facturas

Fuente: El autor

TABLA 65: Descripción Registrar Cotizaciones

Caso de Uso	Registrar Cotizaciones
Resumen	Permite al cajero realizar un estricto control de cuotas en el tiempo, indicando su valor total, número de cuotas, entre otras.
Actor	Cajero
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del cajero.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- El cajero ingresa a la página principal donde se mostrara la información del cliente. 2.- Se dirigirá a crear o ver los recibos de dicho cliente. 3.- Buscara el cliente pertinente, agregará el tratamiento, los productos. 4.- Se analizara si desea el cliente realizar un solo pago o un pago por cuotas. 5.- Si es un pago por cuotas, se indicara los montos pertinentes 6.- Se creará una cuota. <p>Fin Registrar Cotizaciones</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- Registro de la cuota mal digitado. 2.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

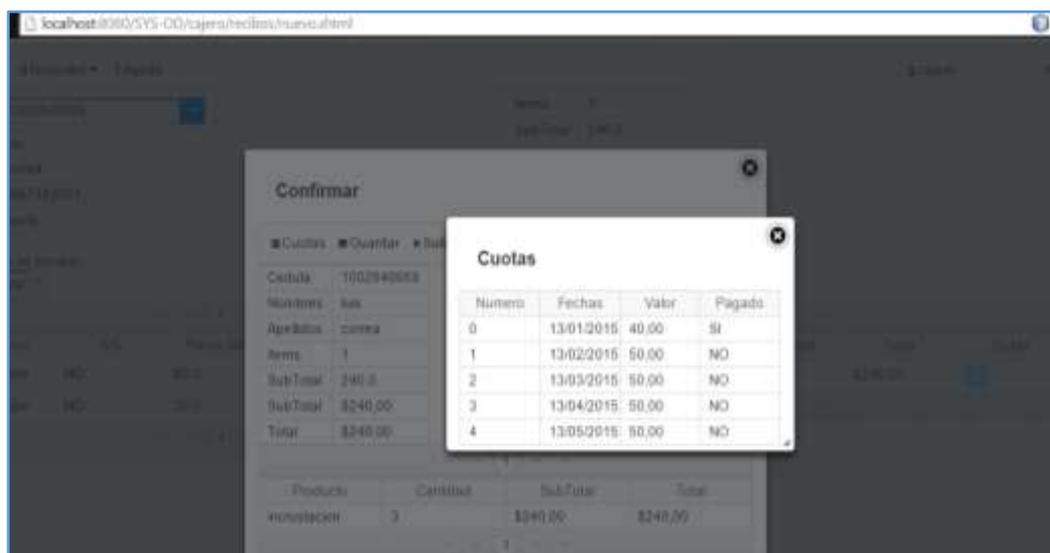


ILUSTRACIÓN 41: Registrar Cotizaciones

Fuente: El autor

TABLA 66: Descripción Registrar Consultas

Caso de Uso	Registrar Consultas
Resumen	Permite al cajero la búsqueda de la información tanto facturación como recibos para conocimiento financiero como información del cliente, si se encuentra debiendo a la "COOD".
Actor	Cajero
Precondición	"SYS-OD" se encuentra en espera a nueva transacción. Una página principal del cajero.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <p>1.- El cajero ingresa a la página principal donde se mostrara la información del cliente.</p> <p>2.- Se podrá observar los detalles de recibos y facturas del cliente</p> <p>Fin Registrar Consultas</p> <p>Trayectoria alternativa</p> <p>1.- El caso de uso finaliza.</p>
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

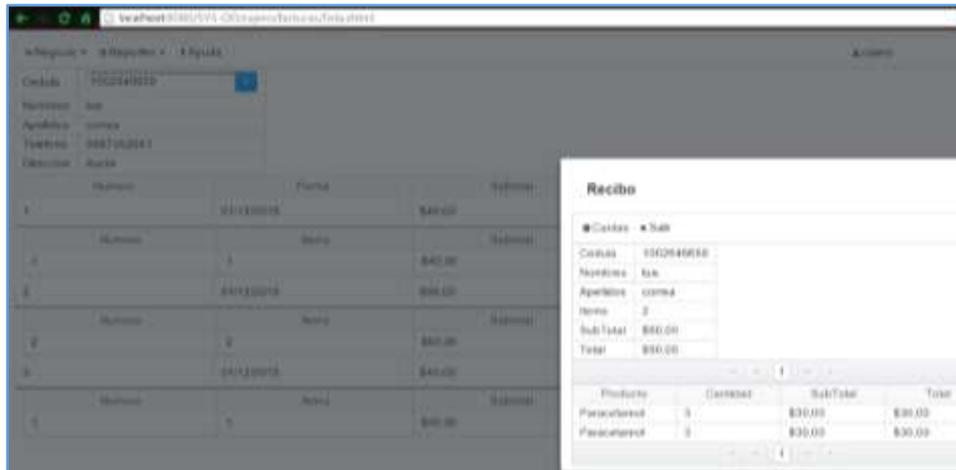


ILUSTRACIÓN 42: Realizar Consultas

Fuente: El autor

Descripción de Módulo de Auditoria

TABLA 67: Descripción Analizar Pistas y Reportes en el Tiempo

Caso de Uso	Analizar Pistas y Reportes en el Tiempo
Resumen	Permite analizar la información de los cambios efectuados en el tiempo.
Actor	Auditor
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del auditor.
Descripción	<p>Una vez logeado en el sistema, se mostrara las diferentes pistas de auditoria.</p> <p>1.- El auditor ingresa a la página principal donde se mostrara la información de todos los cambios que haya habido en el tiempo de uso del sistema.</p> <p>2.- Se podrá observar los detalles de recibos, facturas, ingresos y cambios.</p> <p>Fin Analizar Pistas y Reportes en el Tiempo</p> <p>Trayectoria alternativa</p> <p>1.- El caso de uso finaliza.</p>
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

Rol	Usuario	Fecha	Descripción	Cambios
ROOT		2015-01-12 10:09:07	Creación de un nuevo usuario	Ver
ROOT		2015-01-12 10:08:55	Creación de nuevo rol	Ver
ROOT		2015-01-12 10:08:42	Creación de nuevo rol	Ver
ROOT		2015-01-12 10:08:12	Cambio de datos de un rol	Ver
ROOT		2015-01-12 10:08:04	Cambio de datos de un rol	Ver
ROOT		2015-01-12 10:08:04	Cambio de datos de un rol	Ver
ROOT		2015-01-12 10:08:03	Cambio de datos de un rol	Ver
ROOT		2015-01-12 10:08:03	Cambio de datos de un rol	Ver
ROOT		2015-01-12 10:08:02	Cambio de datos de un rol	Ver
ROOT		2015-01-12 10:08:02	Cambio de datos de un rol	Ver

ILUSTRACIÓN 43: Analizar Reportes en el Tiempo

Fuente: El autor

Descripción de Módulo de Reportes

TABLA 68: Descripción Mostrar Reportes

Caso de Uso	Mostrar Reportes
Resumen	Permite un análisis profundo de la información financiera de un cliente en específico, si está al día con sus pagos a la “COOD”
Actor	Administrador
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del administrador.
Descripción	Una vez logeado en el sistema, se mostrara la información de los clientes. 1.- El administrador ingresa a la página principal donde se mostrara la información del cliente. 2.- Se podrá observar los detalles completos de recibos y facturas del cliente, además de las deudas que debe pagar. Fin Mostrar Reportes Trayectoria alternativa 1.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

Numero	Fecha	Subtotal	Total	Recibos
1	01/12/2015	\$40.00	\$40.00	Ver
Numero	Fecha	Subtotal	Total	Recibos
1	01/12/2015	\$40.00	\$40.00	Ver
2	01/12/2015	\$80.00	\$80.00	Ver
Numero	Fecha	Subtotal	Total	Recibos
2	01/12/2015	\$80.00	\$80.00	Ver
3	01/12/2015	\$40.00	\$40.00	Ver
Numero	Fecha	Subtotal	Total	Recibos
1	01/12/2015	\$40.00	\$40.00	Ver

ILUSTRACIÓN 44: Mostrar Reportes

Fuente: El autor

Descripción de Módulo de Gestión de la Aplicación

TABLA 69: Descripción Gestionar Parámetros

Caso de Uso	Gestionar Parámetros
Resumen	Permite al administrador analizar los diferentes parámetros financieros según las necesidades de la “COOD” en el caso de que exista algún cambio.
Actor	Administrador
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del administrador.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- El administrador ingresa a la página principal donde podrá gestionar los parámetros del sistema. 2.- Digitara la información pertinente. 3.- Crear parámetro. <p>Fin Gestionar Parámetro</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

Parametro	Valor	Acciones
Iva	27	Options
numeroRecibo	5	Options
numeroFactura	4	Options

LUSTRACIÓN 45: Gestionar Parámetros

Fuente: El autor

TABLA 70: Descripción Gestionar Roles

Caso de Uso	Gestionar Roles
Resumen	Permite al Administrador controlar y gestionar los roles que se desempeñan en la “COOD”.
Actor	Administrador
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del administrador.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- El administrador ingresa a la página principal donde podrá gestionar los roles del sistema. 2.- Digitara la información pertinente. 3.- Crear rol. <p>Fin Gestionar Roles</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor



ILUSTRACIÓN 46: Gestionar Roles

Fuente: El autor

TABLA 71: Descripción Gestionar Productos

Caso de Uso	Gestionar Productos
Resumen	Permite al Administrador la gestión de los diferentes productos que contiene un servicio.
Actor	Administrador
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del administrador.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- El administrador ingresa a la página principal donde podrá gestionar los productos del sistema. 2.- Digitara la información pertinente. 3.- Crear producto. <p>Fin Gestionar Productos</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

Servicio	Producto	Estado	IVA	Precio Maximo	Precio Minimo	Acciones
Odontología	Paracetamol	Activo	SI	\$10,00	\$5,50	Opciones
Ortodoncia	modelos	Activo	SI	\$40,00	\$4,00	Opciones
Odontología	incrustacion	Activo	NO	\$80,00	\$40,00	Opciones
Odontología	restauracion	Activo	NO	\$35,00	\$20,00	Opciones
Cirujía	cirujf	Activo	NO	\$50,00	\$4,00	Opciones

ILUSTRACIÓN 47: Gestionar Productos

Fuente: El autor

TABLA 72: Descripción Gestionar Servicios

Caso de Uso	Gestionar Servicios
Resumen	Permite al Administrador la gestión de los servicios de la “COOD”.
Actor	Administrador
Precondición	“SYS-OD” se encuentra en espera a nueva transacción. Una página principal del administrador.
Descripción	<p>Una vez logeado en el sistema, se mostrara la información de los clientes.</p> <ol style="list-style-type: none"> 1.- El administrador ingresa a la página principal donde podrá gestionar los productos del sistema. 2.- Digitara la información pertinente. 3.- Crear Servicio. <p>Fin Gestionar Productos</p> <p>Trayectoria alternativa</p> <ol style="list-style-type: none"> 1.- El caso de uso finaliza.
Postcondición	La transacción es almacenada en el sistema.

Fuente: El autor

Servicio	Cantidad Productos	Productos	Acciones
rayos x	0	Ver	Ver
Ortodoncia	1	Ver	Ver

Producto	Precio minimo	Precio maximo	Eliminar
modelos	\$4,00	\$40,00	Eliminar
Odontología	2	Ver	Ver

Producto	Precio minimo	Precio maximo	Eliminar
exclusividad	\$40,00	\$80,00	Eliminar
restauracion	\$20,00	\$35,00	Eliminar
Cruje	0	Ver	Ver

ILUSTRACIÓN 48: Gestionar Servicios

Fuente: El autor

4.2.4 VISTA DE RESTRICCIONES

En esta vista se presenta las restricciones tanto normativas como estándares de tecnologías a las cuales el sistema está sujeto tanto en el desarrollo como el producto final.

Licenciamiento

Dado que el sistema SYS-OD para la “COOD” se lo va a desarrollar en su totalidad varias herramientas, tecnologías y estándares libres, con Java con licencia GNU GPL, el sistema no necesita de ninguna inversión con respecto al licenciamiento de herramientas a utilizar.

Estándares

Los estándares que se van a utilizar son:

- ✓ **UML:** Es todo artefacto que se generó para la comunicación y documentación, entre el equipo de desarrollo, como usuarios que van a hacer uso del sistema.
- ✓ **Interfaz Web:** La interfaz de usuario debe estar designada para la web. Se debe asegurar que el contenido sea utilizado por cualquier navegador para su uso.

4.2.5 TECNOLOGÍA

El desarrollo total del sistema se realizara con los siguientes parámetros.

TABLA 73: Tecnologías

Elemento	Tecnologías	Observaciones
Base de Datos	MongoDB 2.2.7	Gestor de base de datos libre bajo la licencia BSD
Framework	JSF 2.2 JPA 2.2 PrimeFaces 4.0 Maven 3.3	Tecnología que simplifica el desarrollo en interfaces para el usuario. Tecnología de persistencia que maneja los datos para un desarrollo más rápido. Libería para JSF para facilitar el desarrollo de aplicaciones. Tecnología que permite gestionar las librerías que se va a usar. Arquitectura definida para desarrollo de aplicaciones.
IDE	NetBeans 8.0	Entorno de desarrollo de aplicaciones libres en lenguaje Java
Servidor Web	Glassfish 4.0	Servidor web libre, a base de multiplataforma muy eficaz.
Tipo MIME	CSS HTML JavaScript	Indica la presentación de la aplicación escrito en XHTML para separarlo entre estructuras. Lenguaje marcado de hipertexto para el desarrollo de aplicaciones web. Lenguaje de programación para mejorar la interfaz de usuario y web.
Control de Versiones	GIT (BitBucket)1.9.4	Controla las versiones de los diferentes cambios en la aplicación a través de la web.
Sistema Operativo	Desarrollo y producción: Windows v. 8.1	Última versión de Windows muy estable.
Metodología	RUP	Metodología de desarrollo muy utilizada para realizar aplicaciones de mayor como de menor tamaño.
Reportes	JasperReports	Gestión de reportes

Fuente: El autor

Sistemas existentes

No se encuentra en existencia un sistema de automatización en la “COOD” en el uso de gestión de clientes y registro de facturación.

4.3 FASE DE CONSTRUCCIÓN

En esta fase se realizara el diseño e implementación del prototipo inicial con los puntos antes vistos en la fase de elaboración; se desarrollara el producto operacional en su primera versión. Los documentos que se generan son el modelo de datos, plan de pruebas y lista de riesgos.

4.3.1 INTRODUCCIÓN

Después de realizar el análisis y establecer los diferentes requerimientos por los usuarios de la “COOD”, se llegó a constituir cómo será el diseño específico del sistema SYS-OD a realizar.

Modelo de Datos

En este punto indica la estructuración de las entidades de cada uno de los módulos especificados anteriormente.

Propósito

El propósito de este documento es como definir los diferentes modelos de datos en los diferentes módulos que se detalló en la fase de elaboración para la “COOD”.

Alcance

Se desean desarrollar los diferentes modelos para el sistema “SYS-OD” para la Clínica Odontológica Ortho Dent.

Referencias

- ✓ Resumen de los Interesados

- ✓ Resumen del Modelo de casos de uso

A continuación se presenta un diseño del Diagrama de Entidades, la misma que muestra relaciones de dependencia y asociación, las mismas que serán implementadas para la base de datos MongoDB.

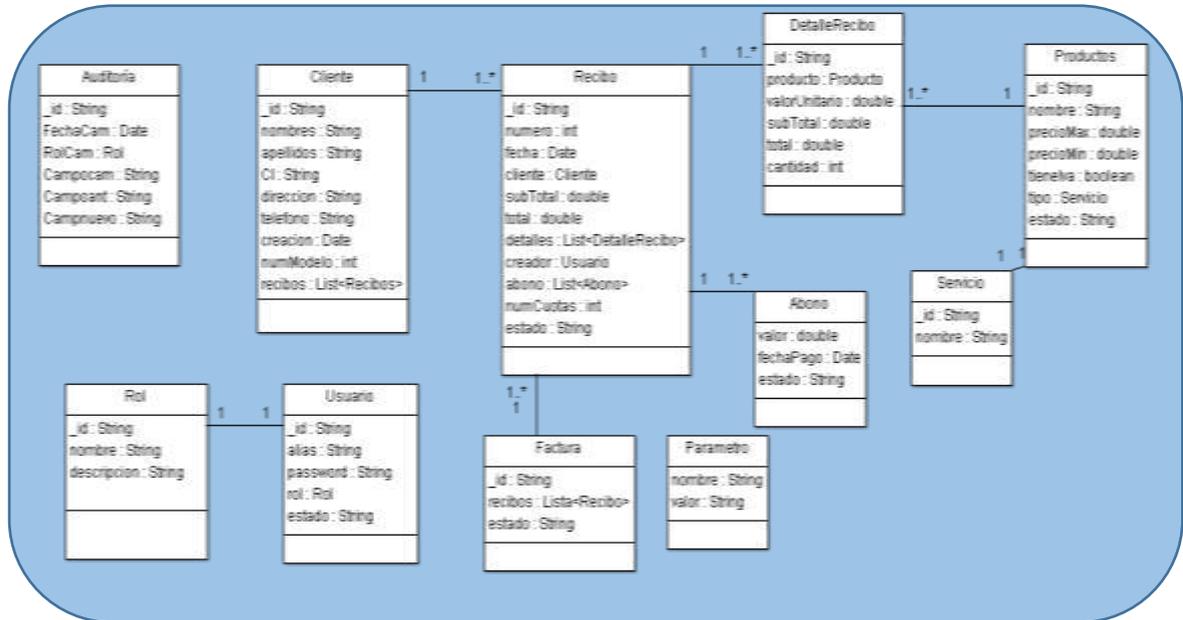


ILUSTRACIÓN 49: Diagrama de Entidades SYS-OD

Fuente: El Autor

Debido a que el Módulo de Auditoría se encarga del análisis de la información se ha creado una sola tabla, el usuario además tendrá un respectivo rol que lo identificara en el momento del ingreso a la información de la clínica; existirán diferentes tipos de servicio para la clínica como son: Ortodoncia, Odontología, Cirugía, Implantes y Rayos X.

4.3.2 PLAN DE PRUEBAS

El presente documento indica el Plan de pruebas como respuesta del el sistema SYS-OD a desarrollar para la Clínica Odontológica Ortho Dent.

Propósito

El propósito del plan de pruebas es de crear pruebas frecuentes para el sistema SYS-OD de la “COOD” con el motivo de conocer diferentes problemas en el desarrollo.

Alcance

Este plan de pruebas se aplicara a todos los diferentes componentes del sistema para identificar amenazas y arreglar diferentes errores durante el proceso de registro y gestión de la información.

4.3.3 ESTRATEGIA DE PRUEBAS

La prueba de software es una parte importante de calidad; la estrategia indica si el comportamiento del sistema SYS-OD cumple con las expectativas funcionales establecidas por la "COOD", por lo que se desarrollaran pruebas, tomando las diferentes consideraciones y obtener un criterio para saber cuándo esta completada la prueba.

Esta etapa implica:

- ✓ Verificación de la integración de componentes.
- ✓ Verificación de requisitos implementados se cumplen correctamente.
- ✓ Verificación de defectos encontrados se han corregido antes de la entrega al usuario de la "COOD".
- ✓ Herramientas.
- ✓ Recursos.
- ✓ Entregables.
- ✓ Lista de riesgos.

Este proceso trata de sacar a luz todos los errores haciéndolo en el menor tiempo, se ha diseñado casos de prueba para el análisis del sistema; por lo que se realizó distintos casos de prueba.

TABLA 74: Pruebas de Integridad de Datos

Objetivo	Asegurar la integridad de datos
Técnica	<ol style="list-style-type: none"> 1.- Registrar datos con tipos válidos. 2.- Registrar datos en entidad que tengan relación con otras. 3.- Asegurarse que los datos se guardan con éxito.
Fin de Criterio	Todos los métodos de acceso y procesos funcionan correctamente como se los diseño.
Consideraciones	<ol style="list-style-type: none"> 1.- Con el uso de conjunto de datos, incrementa la visibilidad a cualquier caso de problema. 2.- Las pruebas deben hacerse con datos reales.

Fuente: El Autor

TABLA 75: Pruebas del Sistema

Objetivo	Asegurar la navegación interna del sistema, ingreso de datos, procesamiento y recuperación.
Técnica	<p>Se Verificó que:</p> <ol style="list-style-type: none"> 1.- El ingreso de datos están controlados y se muestra el debido resultado de error. 2.- Los mensajes de error aparecen en el momento indicado. 3.- Cada regla de negocio antes indicada esta aplicada adecuadamente.
Fin de Criterio	<ol style="list-style-type: none"> 1.- Todas las pruebas han sido ejecutadas correctamente. 2.- Todos los defectos han sido identificados y modificados.
Consideraciones	Se consideró los diferentes impactos en la implementación.

Fuente: El Autor

TABLA 76: Pruebas del Ciclo del Negocio

Objetivo	Asegurar que el sistema funciona con el sistema de negocio antes implementado.
Técnica	<p>Ejecución de cada caso de uso antes mencionado, verificando los campos válidos.</p> <p>1.- Todas las funciones que involucren tiempo fueron probadas.</p> <p>2.- Simular ingresos de clientes, productos, recibos y facturas para simular tiempos.</p> <p>3.- Todas las funciones se han ejecutado correctamente como se mencionó en las charlas con el cliente.</p> <p>4.- Los resultados que se ha esperado cumplen con datos válidos.</p> <p>5.- cada regla de negocio esta aplicada adecuadamente.</p>
Fin de Criterio	<p>Todas las pruebas han sido ejecutadas con éxito.</p> <p>Todos los defectos han sido identificados y tomados en cuenta.</p>
Consideraciones	N/A

Fuente: El Autor

TABLA 77: Pruebas de Interfaz de Usuario

Objetivo	Asegurar la buena navegación de la aplicación y su funcionalidad, se realiza un seguimiento por todos los menús, botones, iconos, alertas.
Técnica	<p>Con la ayuda de los usuarios que van a hacer uso del sistema, verificar la adecuada interacción con el software.</p> <p>Se realizaron las diferentes actividades en la navegación de la aplicación, los usuarios que trabajan son reales y en su respectiva área.</p> <p>Son advertidos que de que el sistema puede fallar.</p>
Fin de Criterio	Se establece un periodo de pruebas, en donde los errores son detectados.
Consideraciones	Se debe establecer un tipo de comunicación entre los usuarios del sistema y los desarrolladores para que puedan detectar problemas y resolverlos rápidamente.

Fuente: El Autor

TABLA 78: Pruebas de Seguridad y Control de Acceso

Objetivo	Asegurar la seguridad y el funcionamiento de la aplicación, verificar que los usuarios solo accedan a las funciones que se les ha dado permiso.
Técnica	Identificar cuáles son los usuarios que van a hacer uso del sistema y controlar en donde pueden navegar. Verificar permisos de usuario, así mismo como registrar los cambios que se han desarrollado en una transacción.
Fin de Criterio	Para cada usuario se ha asignado diferentes funciones en la aplicación.
Consideraciones	El acceso al sistema debe ser revisado por el auditor que se designara para encontrar alguna anomalía.

Fuente: El Autor**TABLA 79:** Recursos

Rol	Recursos Requerido	Responsabilidad
Administrador de pruebas	Luis Correa	Suministrar directrices de pruebas, recursos necesarios
Diseñador de pruebas	Luis Correa	Identificar y planear pruebas.
Gestión de Proyecto	Luis Correa	Ejecuta pruebas y resultados.
Administrador de BDD	Luis Correa	Administra y asegura datos de pruebas.

Fuente: El Autor**TABLA 80:** Entregables

Entregable	Propietario	Revisión/ Distribución
Plan de pruebas	Luis Correa	Coordinador del proyecto

Fuente: El Autor

TABLA 81: Lista de Riesgos

Ranking	Descripción del Riesgo	Estrategia de Reducción de Riesgo
8	La "COOD" no se acople al nuevo sistema "SYS-OD" desarrollado.	A través de los manuales de usuario indicar a los usuarios el buen manejo del sistema.
8	Las características de los equipos que contiene la "COOD", no contemplan los requisitos de la aplicación.	Especificar los debidos requerimientos antes del utilizar el nuevo sistema "SYS-OD".
4	Abandono del proyecto por parte del desarrollador.	Incrementar el compromiso que se tiene en la "COOD" para terminar el sistema.
6	Cambio de algún parámetro de la aplicación para el sistema en la "COOD".	Realizar actas de trabajo para ser aprobadas el interesado, con su firma respectiva.
3	Incompatibilidad del navegador que se usa en la "COOD".	Instalar el navegador para que funcione correctamente.
1	Presencia de virus en los equipos de la "COOD".	Instalar un antivirus en los equipos de cómputo.

Fuente: El Autor

Se realizó el acta pertinente a estas pruebas que se desarrollaron con éxito donde se registró el análisis de los diferentes módulos en la reunión establecida en la Clínica Odontológica Ortho Dent.

Acta de Pruebas
Pruebas de funcionalidad del sistema a entregar a la empresa.



UNIVERSIDAD TÉCNICA DEL NORTE
 Principal: Av. 17 de Julio – Ciudadela Universitaria
 Ibarra - Ecuador
 Teléfono: (593 6) 2997800 ext. 7188
 OE - UTN Oficina del Estudiante
 Web: www.utn.edu.ec

ACTA DE PRUEBAS		
Proyecto: Sistema para la gestión de clientes y registro de pagos de la Clínica Odontológica Ortho Dent.		
Tema a tratar: Pruebas de funcionalidad del sistema a entregar a la empresa.		
Fecha: 26/Enero/2015		
Participantes:		
Nombre	Cargo	Firma
Luis Correa Real	Tesista	
María José Chasiquiza	Secretaría de clínica	
Temas tratados:		
<p>A las 10:00 horas del día Lunes 26 de Enero del 2015, en la Clínica Odontológica Ortho Dent, se procede a realizar una reunión con los involucrados, se presenta como se encuentra el sistema para los diferentes módulos, el mismo que se sugiere para la presentación del sistema.</p> <p>1. Pruebas de funcionalidad: Se desarrollaron distintas pruebas de funcionalidad de cada módulo que se implementó en el sistema dando como resultado un correcto funcionamiento.</p> <p>Se analizaron los diferentes módulos</p> <ul style="list-style-type: none"> ✓ Módulo de Auditoría: se realiza las pistas de auditoría con éxito. ✓ Módulo de Gestión de Aplicación: Se realiza la gestión de los parámetros respectivos con éxito. ✓ Módulo de Gestión de Usuarios: Se realiza el control y acceso seguro a usuarios con éxito. ✓ Módulo de Facturación: Se realiza la gestión la creación, modificación y lectura de la información de la clínica con éxito. ✓ Módulo de Reportes: Se observa los diferentes reportes que necesita la clínica para su completo análisis de datos con éxito. 		
Observaciones adicionales y compromisos adquiridos:		
<ul style="list-style-type: none"> • Se obtuvo con esta acta que se ha realizado todas las pruebas pertinentes al sistema para su correcto funcionamiento e implementación para la aplicación en la Clínica Odontológica Ortho Dent. • No se desarrollaron más cambios en el sistema. 		

4.4 FASE DE TRANSICIÓN

4.4.1 PLAN DE DESPLIEGUE

El objetivo de esta fase de la Metodología RUP es de que el Usuario que va a hacer uso de la aplicación este satisfecho con el trabajo realizado y además el de lograr un producto final que cumpla con todas las expectativas así como la eficacia en su desempeño en la Clínica Odontológica Ortho Dent. Las actividades incluyen

- ✓ Entregar el sistema SYS-OD que se construyó.
- ✓ Experimentar con el producto su entorno de uso, subirlo a la web para poder hacer uso.
- ✓ Proveer asistencia a los usuarios, a través de la capacitación a cada actor que hará uso del sistema.
- ✓ Migrar información existente de la Clínica Odontológica Ortho Dent, almacenando en la base de datos no relacional MongoDB los datos pertinentes para hacer uso de ella.

Una vez que se haya implementado y probado exitosamente el sistema, se debe llevar el producto a la clínica. El propósito final es producir los releases (entregables) del producto a los usuarios finales.

4.4.2 PASO A PRODUCCIÓN

Se establece y describe las diferentes actividades a ejecutar con el fin de aprobar el paso a producción del sistema que se ha desarrollado, con el fin de indicar que el sistema se encuentra en producción, es necesario que se cumpla lo siguiente:

Revisión de documentación y entrega de la fuente del sistema.

En este caso se revisara toda la documentación y la fuente del sistema.

Documentos:

- ✓ Acta de Aprobación de Usuarios.
- ✓ Manual de Usuario.
- ✓ Manual Técnico del Sistema.
- ✓ Manual de instalación.

En la siguiente tabla se indica las aprobaciones para el paso a producción, una vez que se entregó toda la información pertinente.

TABLA 82: Actividades de Paso a Producción

Aprobado por: Dra. Beatriz Rosero		Aprobado
Tema: SYS-OD		
Detalle: Sistema para la Clínica Odontológica Ortho Dent		
Revisión de documentación y entrega de programa fuente		Ok
	Acta de Aprobación de Usuarios.	Ok
	Manual de instalación.	Ok
	Manual de Usuario.	Ok
	Manual Técnico del Sistema.	Ok

Fuente: El Autor

4.4.3 TRANSFERENCIA TECNOLÓGICA Y CAPACITACIÓN

La transferencia de tecnología se lo realizara para casa uno de las personas que están involucradas en el uso de la aplicación web SYS-OD, además se proveerá de documentos para el buen uso del sistema como son:

- ✓ Manual de usuarios
- ✓ Manual técnico
- ✓ Manual de instalación

Estos manuales se incluyen en los anexos.

CAPITULO V

5 CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES.

Éste estudio de bases de datos estableció un análisis con resultados técnicos por lo que se ha logrado obtener información detallada acerca de este gestor de bases de datos NoSQL MongoDB, lo que generó varias opiniones que son ventajosas para el tesista y para la empresa donde se desarrolló el sistema.

Se creó e implementó un sistema integrado de calidad para la Clínica Odontológica Ortho Dent de la ciudad de Ibarra con los módulos que se describieron en este proyecto; todas las clínicas que se encuentran en la ciudad de Ibarra deberían realizar sus diferentes procesos en un sistema de calidad que permita transacciones automáticas.

A través del estudio entre la base de datos noSQL MongoDB y la base de datos relacional PostgreSQL, MongoDB llega a ser el doble o triple de rápido con respecto a la base de datos PostgreSQL; Si se desea utilizar velocidad pero poca con pérdida de información en el tiempo, MongoDB es la mejor elección.

MongoDB es apta para el desarrollo de aplicaciones web en los que se necesita de una base de datos no relacional tanto para la Clínica Odontológica Ortho Dent de la ciudad de Ibarra como para cualquier sistema que se desee desarrollar, como es el caso de la base de datos noSQL que se la estudió que se centra más en almacenar y consultar de manera rápida y eficaz con la única desventaja de que sus datos puedan perderse con el tiempo.

RUP es un proceso de mejores prácticas por el cual la ingeniería de software se disciplina completamente entre responsabilidades y trabajos en el transcurso de un desarrollo de la aplicación, además, reduce considerablemente el número de cambios que se desarrollan en el desarrollo de una aplicación y analiza cualquier información que se destaque en el software.

El software libre realiza muchas aportaciones en la construcción y desarrollo de aplicaciones con una excelente calidad, tanto para pequeñas y medianas empresas (PYMES) que satisfagan las necesidades del usuario, haciendo de ésta una táctica para competir en esta carrera entre tecnologías frente al software privativo.

Después de automatizar los distintos procesos que contiene la Clínica Odontológica Ortho Dent mediante el uso de software libre, que anteriormente se realizaban de manera manual, se obtuvo un ahorro eficaz en el tiempo y costos al momento de realizar transacciones.

Los resultados finales que se obtuvieron en las diferentes etapas de las pruebas del sistema fueron de manera satisfactoria, por tanto se concluye que a la aplicación desarrollada se la puede perfeccionar con nuevas implementaciones o módulos para así mejorar su objetivo inicial.

5.2 RECOMENDACIONES

Se recomienda prepararse perfectamente para un estudio y desarrollo de un tema, conociendo toda la información existente en el tema que estemos analizando, asimismo se necesita del apoyo de las herramientas principales para un buen levantamiento de la información y desarrollo de un buen estudio y aplicación.

Se recomienda que se continúe con las prácticas como lo es el pensum de estudio que brinda la Universidad Técnica del Norte se logra un mejor adiestramiento para el desarrollo de software de calidad, logrando disminuir tiempo, esfuerzo y asegurando un software y análisis de calidad.

En diferentes sitios web que han manejado la base de datos NoSQL MongoDB, han logrado manejar de manera exitosa la información de sus sitios en tiempo real, por lo que se sugiere a la base de datos MongoDB como una buena opción para el desarrollo de aplicaciones informáticas libres de calidad como lo fue para la Clínica Odontológica Ortho Dent.

En el desarrollo de aplicaciones se debería profundizar más en la investigación de herramientas libres como lo fue con la base de datos noSQL MongoDB y así analizar los diferentes beneficios que brinda cada una.

Es recomendable realizar una aplicación software con la metodología de desarrollo RUP para conservar la información de manera ordenada y documentada durante todo el proceso de desarrollo e implementación, ya que garantizan una aplicación de calidad.

Se recomienda incentivar el uso de los sistemas informáticos libres para reducir el impacto privativo como ambiental una empresa que no tenga sus recursos para invertir en una herramienta privativa y que aún se maneja con el almacenamiento de la información a papel.

Se necesita adiestrar a los usuarios que van a hacer el uso del sistema web desarrollado, conociendo todas las partes funcionales pertinentes para aprovecharlas al máximo, y obtengan los diferentes beneficios que se indicó en el alcance y delimitación.

Dentro de los posibles temas a tratar en este proyecto de tesis como nuevos módulos importantes son: la consolidación de los diferentes centros de la Clínica Odontológica Ortho Dent para el control médico de los clientes; evitando una duplicación de información y así obtener un historial médico único, impresión de facturas, administración de citas médicas para descongestionar el masivo número de clientes en la Clínica Odontológica Ortho Dent de Ibarra y brindar un mejor servicio a los clientes.

5.3 ANÁLISIS DE IMPACTOS

El análisis de impacto permite a determinar los beneficios importantes y la aceptación que tiene el sistema con la Clínica Odontológica Ortho Dent.

Para realizar este análisis sobre la implementación en la "COOD" se estableció con tiempos en transacciones de los procesos antes y ahora como: gestión de facturas, gestión de clientes, gestión de recibos, consumo de papel, generación de reportes y tiempos de atención al cliente.

TABLA 83: Análisis de impactos

Descripción	Cantidad	Costo	
		Antes	Ahora
Gestión de facturas	1 factura	20 min	10 min
Gestión de recibos	1 recibo	15 min	5 min
Generación de reportes	Cualquier reporte	40 min	15 min
Tiempos de atención al cliente	1 cliente	20 min	5 min
Consumo de papel		150 hojas	70 hojas

Fuente: El Autor

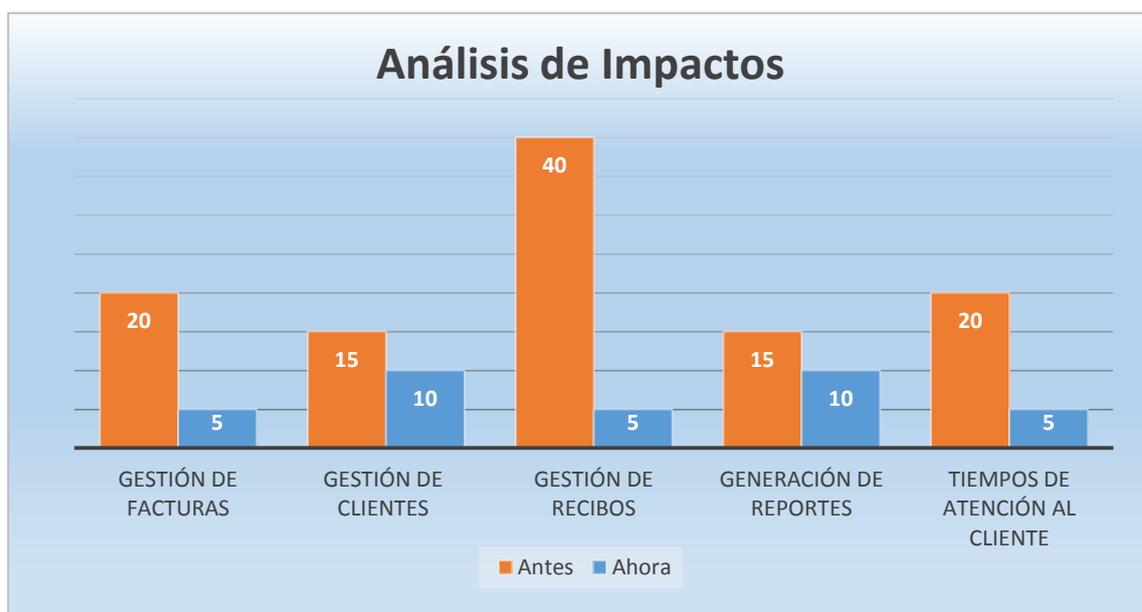


ILUSTRACIÓN 50: Análisis de impactos

Fuente: El Autor

Lista de beneficios

- ✓ Reducción del uso de papel.
- ✓ Información agilizada y actualizada.
- ✓ Generación de reportes a tiempo.
- ✓ Buena imagen institucional.
- ✓ Control adecuado de la información.
- ✓ Satisfacción de los clientes en el servicio.

En este caso de beneficios no se representa en cantidades monetarias sino en manera cualitativa con la implementación de este sistema para la Clínica Odontológica Ortho Dent, que si hizo con el fin de mantener la información de los clientes y de la clínica en sí y asegurar los riesgos que estaba expuesta.

Con lo expuesto anteriormente se da como resultado el análisis como beneficio positivo para la Clínica Odontológica Ortho Dent sobre el sistema "SYS-OD" que se implementó en ella.

La creación de clientes es mucho más rápida ya que se la almacena en un repositorio digital, el almacenamiento de los recibos así como de facturas es mucho más rápida con respecto a lo que anteriormente se realizaba en agendas manuales que se podrían perder en el tiempo; ahora con el sistema actual implementado se encuentra la información muy bien protegida al estar almacenada en una estructura definida de datos

Con el sistema actual también se puede realizar y analizar estadísticas de los clientes que adeudan a la Clínica Odontológica Ortho Dent, el sistema genera un reporte con datos del cliente, anteriormente este análisis se demoraba mucho al tener que examinar los diferentes pagos en el tiempo del cliente.

5.4 GLOSARIO

5.5 INTRODUCCIÓN.

El glosario de términos es también parte del sistema “SYS-OD” y una recopilación de términos empleados en el sistema. Este glosario representa una guía general de las definiciones y conceptos para la adecuada comprensión dentro del sistema.

5.6 PROPÓSITO

La función principal del presente glosario es proporcionar un estándar de términos y conceptos asociadas a los términos técnicos del sistema.

5.7 REFERENCIAS.

Éste documento hace referencia a términos empleados en toda la documentación del sistema.

5.8 GLOSARIO DE TÉRMINOS

Aplicación web: Aplicación informática que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet

Arquitectura: Representación abstracta de los componentes de un sistema y su comportamiento.

API.- Interfaz de Programación de Aplicaciones (Application Programming Interface) es el conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Artefacto.- Un artefacto es un documento tangible como resultado de desarrollo de software. Un documento generado a lo largo del desarrollo del producto.

Beans.- Son un modelo de componentes creado por Sun Microsystems para la construcción de aplicaciones en Java.

Bussines Intelligence: Conjunto de estrategias y aspectos relevantes enfocados a la administración y creación de conocimiento.

Casos de Uso.- Es una descripción de los pasos o las actividades que deberán realizarse para llevar a cabo algún proceso.

CSS.- El nombre hojas de estilo en cascada viene del inglés Cascading Style Sheets.

CRUD.- Es el acrónimo de crear, obtener, actualizar y eliminar, es decir las funciones básicas en bases de datos o en la capa de persistencia de un sistema de software.

Cardinalidad: Indica el número de entidades con las que puede estar relacionada una entidad dada, estas pueden ser (uno a uno) (uno a varios) (varios a no) (varios a varios).

DBMS: (Sistema Manejador de Bases de Datos). Es un software que controla la organización, almacenamiento, recuperación, seguridad e integridad de datos.

DataWherehousing: Es una colección de datos orientada a almacenar información, datos

Esquema: El esquema define las tablas, campos en cada tabla y las relaciones entre cada campo y cada tabla.

Escalabilidad: Que puede crecer de manera que pueda adaptarse al medio.

Entidades: Es un objeto donde se recoge información de interés de cara a la base de datos.

Exabyte: Es una unidad de medida de almacenamiento de Datos cuyo símbolo es el 'EB', equivale a 10¹⁸ bytes.

Framework.- Es un esquema (patrón) para el desarrollo y/o la implementación de una aplicación.

HTML.- (Siglas de Hyper Text Markup Language) Lenguaje de marcado de hipertexto.

Inconsistencia: Cuando existe redundancia puede existir inconsistencia y nos dice que si un dato es actualizado en un sitio, puede que otro dato duplicado este no actualizado

JSF.- Es una tecnología y framework para aplicaciones Java basadas en web que simplifica el desarrollo de interfaces de usuario en aplicaciones Java EE.

Java: Lenguaje de Programación orientado a objetos desarrollado por Sun Microsystems.

JDBC: Java DataBase Connectivity es el API de Java que define como una aplicación cliente accederá a una base de datos, independientemente del motor de base de datos al que accedamos.

JSON: JavaScript Object Notation, es un formato ligero para el intercambio de datos.

Mapping o Mapeo.- Es una técnica de programación para convertir datos entre el sistema de tipos utilizado en un lenguaje de programación orientado a objetos y el utilizado en una base de datos relacional.

MongoDB: Es una base de base de datos multiplataforma libre que es orientada a documentos quiere decir que se agregan los datos o información que deseemos en documentos.

PostgreSQL: es un sistema de gestión de base de datos relacional y libre, que agiliza la interacción de cliente, servidor y base de datos.

PrimeFaces.- Framework de código abierto que añade la capacidad de ajax en aplicaciones JSF sin recurrir a JavaScript.

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

Redundancia: Quiere decir que un dato almacenado se encuentren guardado varias veces en diferentes lugares.

SQL: Es un lenguaje de acceso a bases de datos que explota la flexibilidad y eficacia de sistemas relacionales.

SSL (Capa Segura de Comunicaciones): proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía.

Trigger: Es un proceso en el cual se ejecuta cuando se cumple una condición de una operación.

5.9 BIBLIOGRAFÍA

Kurth, J. (1998). Fundamentos de Bases de Datos. Madrid – España: McGraw-Hill.

Freire, T. (2011). Sistema de información odontológica bienestar universitario (Tesis de pregrado). Universidad Técnica del Norte, FICA, Ecuador. Recuperado de: <http://repositorio.utn.edu.ec/handle/123456789/571>

Bushik S. (2012), “A vendor-independent comparison of NOSQL databases: Cassandra, HBase, MongoDB, Riak Altoros Systems Inc.”, Special to Network World. Recuperado de: <http://www.networkworld.com/news/tech/2012/102212-nosql-263595.html>

Ivar J, Grady B y Rumbaugh J. (2010). “El Proceso Unificado de Desarrollo de Software”. Recuperado de: <http://www.slideshare.net/Sofylutqm/el-proceso-unificado-3943047>.

Tienle (2013), “NoSQL Data Modeling Techniques”, TIENLE'S BLOG, Web Development Sharing, Recuperado de: <http://www.tienle.com/2013/02-26/nosql-data-modeling-techniques.html>

Argenis, M., Rojas W. (2005), “Comparación entre sistemas de gestión de bases de datos (SGBD), Bajo Licenciamiento Libre y Comercial” (Tesis de pregrado), Universidad Católica de Colombia, faculta de Ingeniería de Sistemas, Bogotá. Recuperado de: <http://www.ilustrados.com/documentos/sghbd.pdf>

NoSQL la evolución de las bases de datos, SG #28 (Mayo - Julio 2010). Recuperado de http://sg.com.mx/revista/42/nosql-la-evolucion-las-bases-datos#.U4YSY_I5MUN

Díaz W. (2013, 27 de mayo). Bases de Datos NoSQL: llegaron para quedarse. Mensaje dirigido a <http://basesdedatosnosql.blogspot.com/>

Rational Unified Process (RUP), tomado de <http://ima.udg.edu/~sellares/EINF-ES2/Present1011/MetodoPesadesRUP.pdf>

Chodorow, K. (2010). MongoDB la guía definitiva. Recuperado de: <http://books.google.es/books?id=uGUKiNkKRJ0C&pg=PA342&dq=mongodb&hl=es&sa=X&ei=LzRFVIDNNdSNsQSbu4HACQ&ved=0CD4Q6AEwAg#v=onepage&q=mongodb&f=false>

Douglas, K. Douglas, S. (2003). PostgreSQL. Recuperado de: http://books.google.es/books?id=gkQVL9pyFVYC&printsec=frontcover&dq=postgresql&hl=es&sa=X&ei=rzZfVNqGK--LsQTh_4GQAw&sqi=2&ved=0CCAQ6AEwAA#v=onepage&q=postgresql&f=false

Obe, R. y Hsu, L. (2012). PostgreSQL up and running. Recuperado de: http://books.google.es/books?id=Q8jklZkMTPcC&printsec=frontcover&dq=postgresql&hl=es&sa=X&ei=rzZfVNqGK--LsQTh_4GQAw&sqi=2&ved=0CEIQ6AEwBA#v=onepage&q=postgresql&f=false

Plugge, E. Membrey, P. y Hawkins, T. (2012). The definitive guide to MongoDB. Recuperado de: <http://books.google.es/books?id=6wAJLJRxFt8C&pg=PA22&dq=mongodb&hl=es&sa=X&ei=LzRFVIDNNdSNsQSbu4HACQ&ved=0CE4Q6AEwBA#v=onepage&q=mongodb&f=false>

Carvajal, Rubinos, A. M. y Alina, H. (2011). Security Database (English). Recuperado de: <http://web.b.ebscohost.com/ehost/detail/detail?vid=20&sid=6519a0ab-e6e4-4fdc-84e0-425b5320eb09%40sessionmgr113&hid=110&bdata=JnNpdGU9ZWZwhvc3QtbGl2ZQ%3d%3d#db=fua&AN=77938343>

5.10 ANEXOS

A continuación se muestran una lista de anexos que constan en este documento y otros adicionales.

Anexos en el presente documento.

- ✓ **Anexo A:** Manual de Usuario.
- ✓ **Anexo B:** Manual Técnico.
- ✓ **Anexo C:** Manual de Instalación.
- ✓ **Anexo D:** Diagrama de Entidades del sistema para la “COOD”.
- ✓ **Anexo E:** Diagrama de casos de uso del sistema para la “COOD”.

Anexos en el CD

- ✓ **Anexo A:** Manual de Usuario.
- ✓ **Anexo B:** Manual Técnico.
- ✓ **Anexo C:** Artículo Científico
- ✓ **Entregables de Metodología RUP**

Actas de trabajo

Casos de uso módulo de Gestión de Clientes

Casos de uso módulo de Facturación

En el CD anexo en documento aparte

MANUAL DE USUARIO

1. Introducción

El presente manual ha sido elaborado para el uso apropiado del “SYS-OD”. Se presentan las pantallas correspondientes a cada menú, acompañado de la respectiva aclaración de sus funciones.

2. Ingreso al Sistema

Para iniciar es necesario un explorador de internet que tengan soporte para componentes JSF y java. Al iniciar el sistema tendremos la siguiente pantalla que el usuario podrá usar, esta página es para el ingreso de usuarios que han tienen su usuario y contraseña.



ADMINISTRADOR

Una vez ingresado como administrador ingresara a la página principal, tendrá la oportunidad de revisar y gestionar las opciones que se indican a continuación:



1. Gestión de Aplicación

5. Reportes

6. Ayuda

Además de esto también podrá revisar la información primordial de la clínica como la cantidad que tiene en caja, cuanto tiene por cobrar y total facturado, además de esto podrá ver los diferentes ingresos por mes y por día en los gráficos.

Además de esto el administrador podrá gestionar y controlar los diferentes atributos primordiales de la clínica como lo es los:

1. Roles

1. Nuevo

2. Gestionar

2. Usuarios

1. Nuevo

2. Gestionar

3. Servicios

1. Nuevo

2. Gestionar

4. Productos

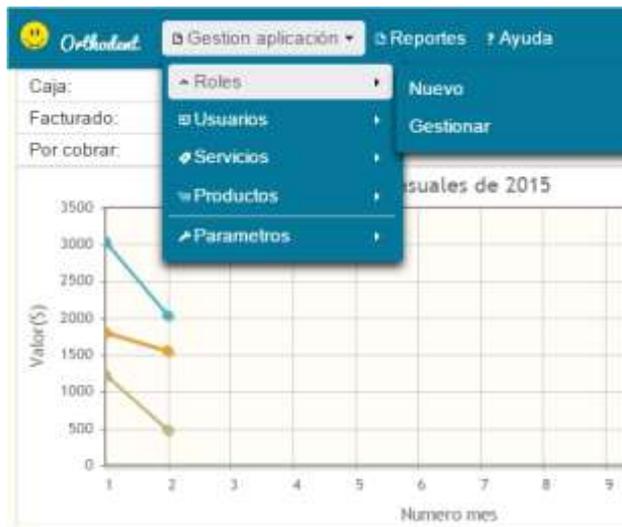
1. Nuevo

2. Gestionar

5. Parámetros

1. Nuevo

2. Gestionar



Agregar como editar la información de cada rol creado, además puede observar que usuario esta designado al rol.

The screenshot shows the 'Lista de usuarios' interface. It features a table with columns: 'Rol', 'Rol', 'Usuarios', and 'Acciones'. The table contains three rows of user data:

Rol	Rol	Usuarios	Acciones
CAJERO	administra recibos facturas	Ver	[Editar]
ADMIN	administra la informacion	Ver	[Editar]
AUDITOR	analiza la informacion	Ver	[Editar]

An 'Editar' modal window is open, showing a form with the following fields:

- Permiso: Cajero
- Descripción: administra recibos facturas

There is a 'Guardar' button at the bottom of the modal.

Agregar como editar la información de cada usuario creado, además puede observar el rol a cual usuario está ligado.

The screenshot shows the 'Lista de usuarios' interface. It features a table with columns: 'Usuario', 'Rol', 'Rol', and 'Acciones'. The table contains three rows of user data:

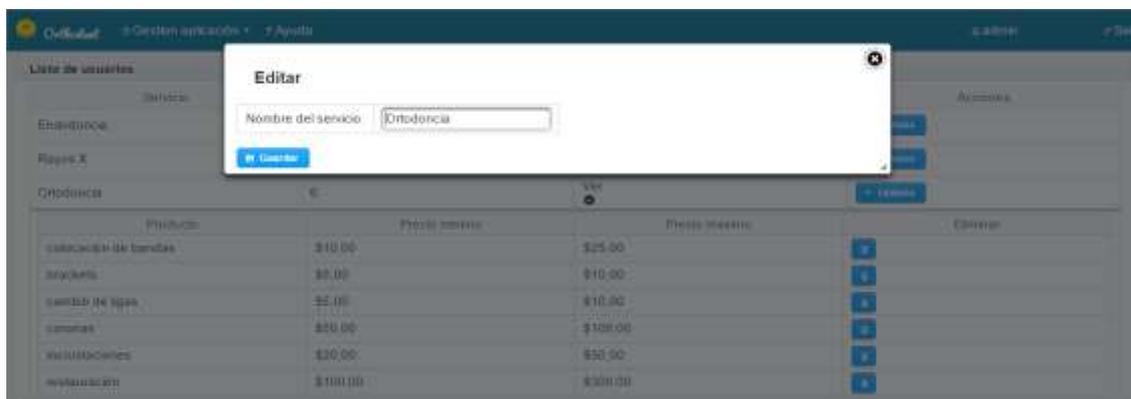
Usuario	Rol	Rol	Acciones
admin	ADMIN	Admin	[Editar]
auditor	AUDITOR	Admin	[Editar]
cajero	CAJERO	Admin	[Editar]

An 'Editar' modal window is open, showing a form with the following fields:

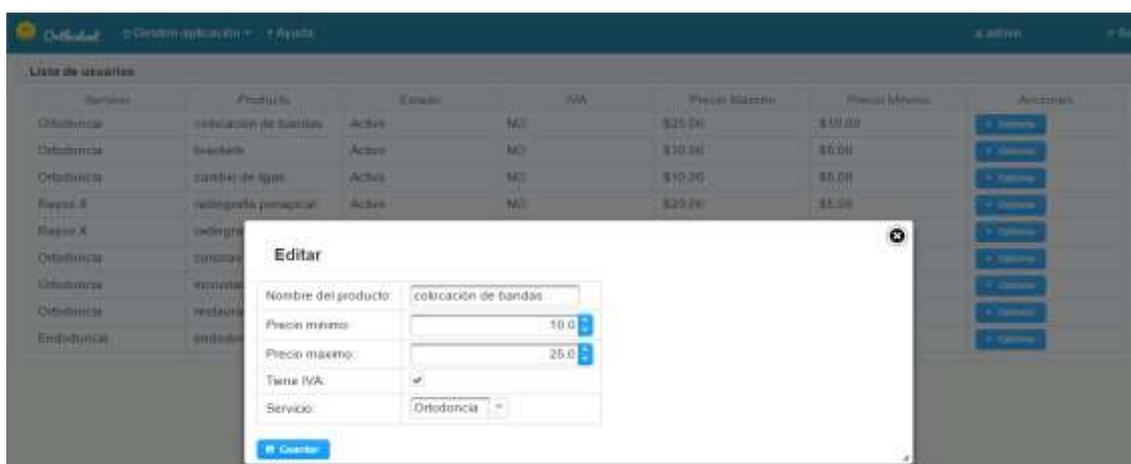
- Nombre de usuario: admin
- Contraseña: admin
- Nombre de usuario: ADMIN

There is a 'Guardar' button at the bottom of the modal.

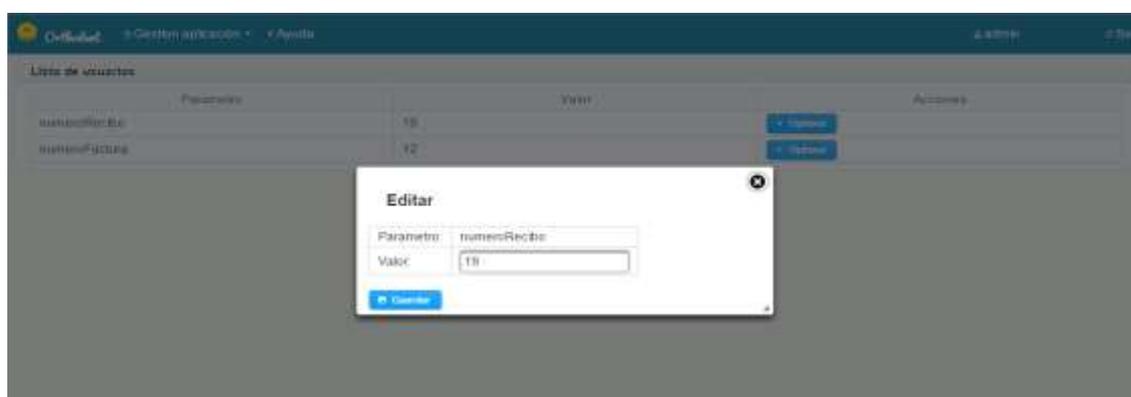
Agregar como editar la información de cada servicio creado, además puede observar los diferentes productos por cada servicio.



Agregar como editar la información de cada producto creado como su nombre, precio min y máx., IVA y su servicio ligado.



Agregar como editar parámetros de la empresa ligados a recibos como facturas e IVA.



En el indicador de los reportes, se mostrara la información primordial de la clínica con respecto a sus recibos como facturas y principalmente cliente, sus movimientos y tratamientos realizados, también se podrán realizar la respectiva descarga del mismo.

En la parte de clientes podremos observar la información del cliente, todos sus movimientos y tratamientos que ha realizado en la clínica.

Identificación	Número modelo	Nombres	Apellidos	Teléfonos	Recibos	Facturas
456788843	No tiene	vadin	vilota	1234556	Guardar PDF	Guardar PDF
1003646566	No tiene	vilota	vadin	245443434	Guardar PDF	Guardar PDF
1002646659	55	luis	correa	123456	Guardar PDF	Guardar PDF
1234567890	No tiene	grjalva	german	1234556	Guardar PDF	Guardar PDF
123456789	20	milton fabian	basidas	1233456	Guardar PDF	Guardar PDF
1003646271	144	Mauricio	Rea	0987654321	Guardar PDF	Guardar PDF

En la pestaña de recibos, se mostrara todos los recibos que se encuentren almacenados en el sistema indicando sus cuotas, que se abonó, cuanto falta de pagar, subtotal y total.

Numero c	Fecha	Cuotas Pagadas	Abonado	Faltante	Subtotal	Total
23	13/02/2015	8 de 8	\$145.00	\$0.00	\$145.00	\$145.00
21	05/02/2015	1 de 11	\$120.00	\$480.00	\$600.00	\$600.00
22	06/02/2015	1 de 1	\$40.00	\$0.00	\$40.00	\$40.00
20	02/02/2015	6 de 6	\$1,200.00	\$0.00	\$1,200.00	\$1,200.00
19	02/02/2015	3 de 3	\$50.00	\$0.00	\$50.00	\$50.00
17	28/1/2015	1 de 21	\$20.00	\$180.00	\$200.00	\$200.00
18	28/01/2015	1 de 1	\$1,000.00	\$0.00	\$1,000.00	\$1,000.00
12	26/01/2015	1 de 1	\$20.00	\$0.00	\$20.00	\$20.00
13	26/01/2015	1 de 1	\$30.00	\$0.00	\$30.00	\$30.00
14	26/01/2015	1 de 21	\$100.00	\$600.00	\$700.00	\$700.00
15	26/01/2015	1 de 1	\$20.00	\$0.00	\$20.00	\$20.00
16	26/01/2015	1 de 1	\$10.00	\$0.00	\$10.00	\$10.00
11	26/01/2015	3 de 18	\$260.00	\$450.00	\$710.00	\$710.00
10	26/01/2015	11 de 11	\$350.00	\$0.00	\$350.00	\$350.00

En la pestaña de facturas, se mostrara todas las diferentes facturas con sus respectivos recibos que se encuentren almacenados en el sistema indicando el número de la factura, la fecha de factura, subtotal, total y los recibos que se necesitaron para crear la factura.

Número	Fecha	Subtotal	Total	Recibos
10	26/01/2015	\$20.00	\$20.00	Ver
15	1	\$20.00	\$20.00	
11	28/01/2015	\$10.00	\$10.00	Ver
12	02/02/2015	\$10.00	\$10.00	Ver
13	10/02/2015	\$350.00	\$350.00	Ver
14	13/02/2015	\$145.00	\$145.00	Ver

La ayuda nos enviara a este documento que indica las diferentes funciones que tiene el sistema.

CAJERO

Una vez ingresado como cajero, tendrá la oportunidad de revisar y gestionar las diferentes opciones que se indican a continuación:

1. Clientes

2. Recibos

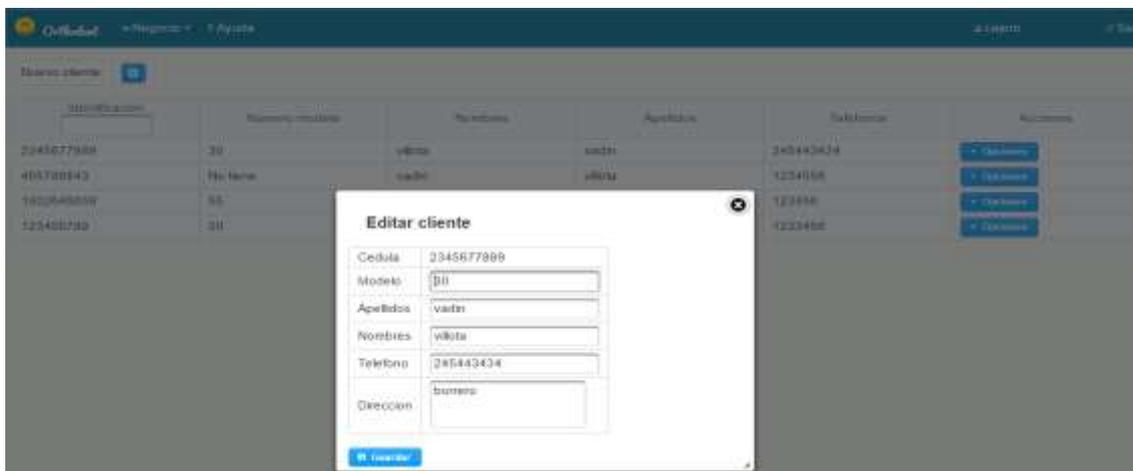
1. Agregar
2. Gestionar

3. Facturas

1. Agregar
2. Gestionar

Identificación	Modelo	Nombres	Apellidos	Teléfonos	Acciones
2345677889	30	vilota	vadin	245443434	Acciones
456788843	No tiene	vadin	vilota	1234556	Acciones
1002648659	55	luis	comea	123456	Acciones
123456789	20	milton fabian	barbóas	1233456	Acciones

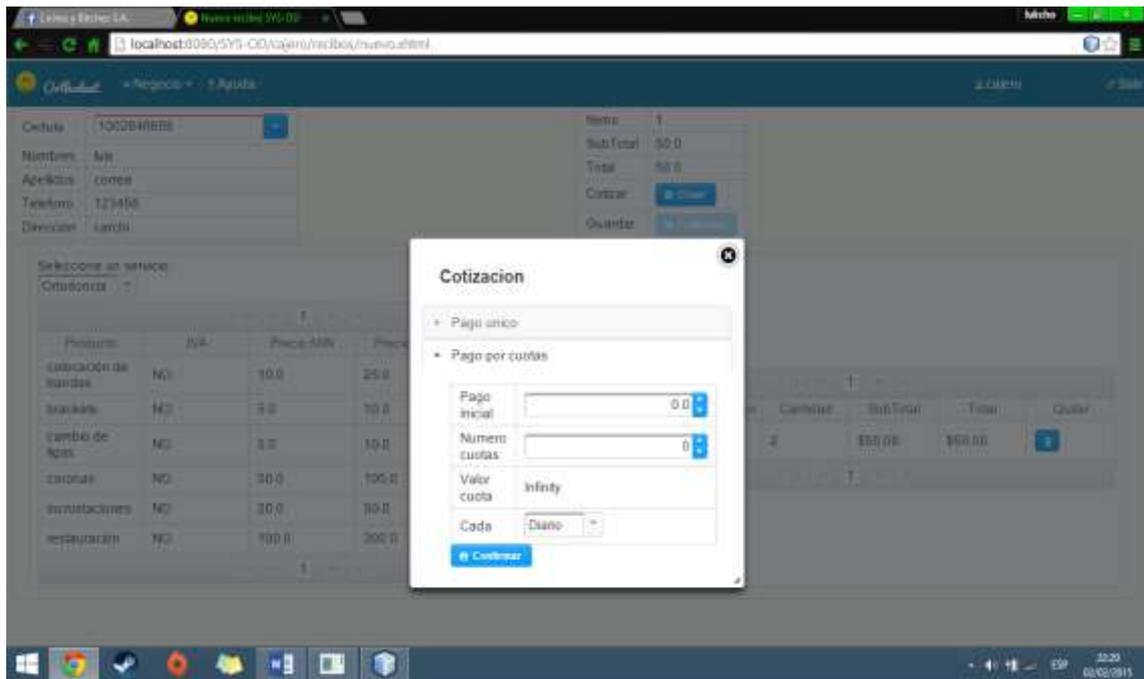
La página principal es la de clientes por lo que nos llevara al inicio de la página del cajero donde se podrá observar los diferentes clientes que pertenecen a la “COOD”, también se podrá editar como también ver tanto sus recibos como sus facturas dando clic en el botón opciones.



Al momento de agregar un nuevo recibo, pondremos primero el número de cedula del cliente que vamos a realizar un recibo, una vez encontrado el usuario, escogemos los diferentes tratamientos a través del servicio, agregamos la cantidad del producto solicitado y luego creamos el recibo.



En la parte de cotización tenemos dos opciones el pago único que es cuando se depositara en ese instante; caso contrario si se desea realizar cuotas, se deberá agregar el pago inicial, el número de cuotas, el valor cuota indicara el monto por el cual deberá pagar el cliente y si se desea pagar por meses o por días y se dará clic a confirmar.



Se confirmara el recibo por lo que se podrá observar datos del cliente y el servicio que usara, se podrá observar el número de cuotas a pagar, y se dará clic en guardar por lo que almacenara.



En la parte de gestión de recibos se podrá buscar por cedula de cliente, los recibos que se encuentren de cada cliente.



Si nos dirigimos a opciones por cada recibo se podrá observar los detalles como las cuotas del recibo, así como sus cuotas por pagar.

Confirmar

Cedula: 456788843
 Nombre: vadin
 Apellidos: vilota
 Items: 1
 SubTotal: \$30.00
 Total: \$30.00

Producto	Observacion	Cantidad	SubTotal	Total
radiografía panorámica	peza 2	2	\$30.00	\$30.00

Cuotas

Numero	Fecha estimada	Fecha de pago	Valor	Pagado	Pagar
0	02/02/2015	02/02/2015 10:32	20.0	SI	Pagar
1	02/03/2015		236.0	NO	Pagar
2	02/04/2015		236.0	NO	Pagar
3	02/05/2015		236.0	NO	Pagar
4	02/06/2015		236.0	NO	Pagar
5	02/07/2015		236.0	NO	Pagar

Cada vez que el cliente se acerque a pagar se descontara del saldo total y se guardara la información por lo que se podrá analizar después cuanto es lo que debe.

En Agregar una factura se podrá agregar a través de los diferentes recibos que se encuentren por cada cliente buscándolo por su cedula, se agregara uno o varios recibos y se confirmara la factura.

Orfebral - Registros - Ayuda

Cedula: 1002048659
 Nombre: luis
 Apellidos: conesa
 Telefono: 123456
 Direccion: carchi

Recibos: 0
 SubTotal: 0.0
 Total: 0.0
 Guardar [Guardar](#)

Recibos disponibles

Numero	Fecha	Items	Subtotal	Total	Detalles	Agregar
10	26/01/2015	1	\$10.00	\$10.00	-	+
15	26/01/2015	1	\$20.00	\$20.00	-	+

Factura

Numero	Items	Subtotal	Total	Detalles	Quitar
No records found.					

Por lo que se confirmaría la factura y se almacenara

Confirmar

Esta seguro que desea factura los siguientes recibos:

Numero	Items	Subtotal	Total	Detalles
10	1	\$10.00	\$10.00	-

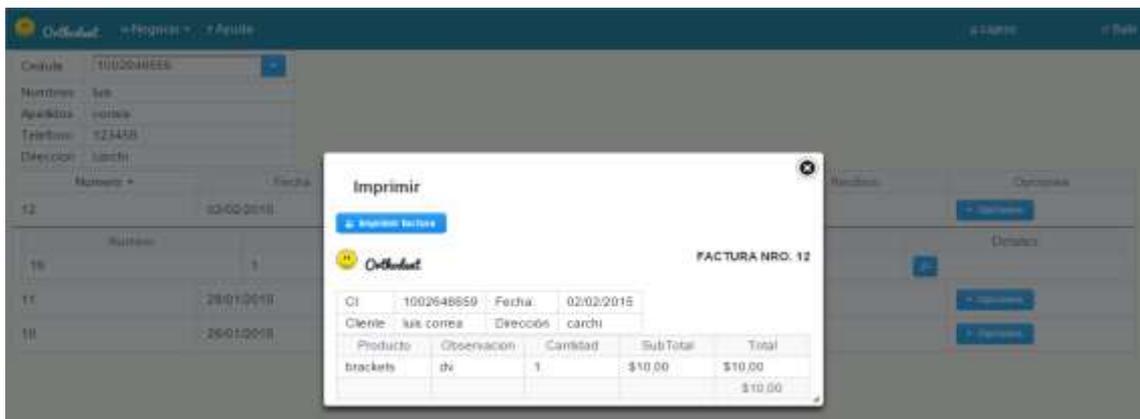
Recibos disponibles (background):

Numero	Fecha	Items	Subtotal	Total	Detalles	Quitar
15	26/01/2015	1	\$20.00	\$20.00	-	+
10		1	\$10.00	\$10.00	-	+

En gestionar facturas se podrá observar por cada cliente sus facturas desglosándolo con sus recibos respectivos por cada factura que se creó, además de esto podrá observar con detalle que recibos fueron necesarios para la creación de facturas y además, se podrá imprimir la factura.



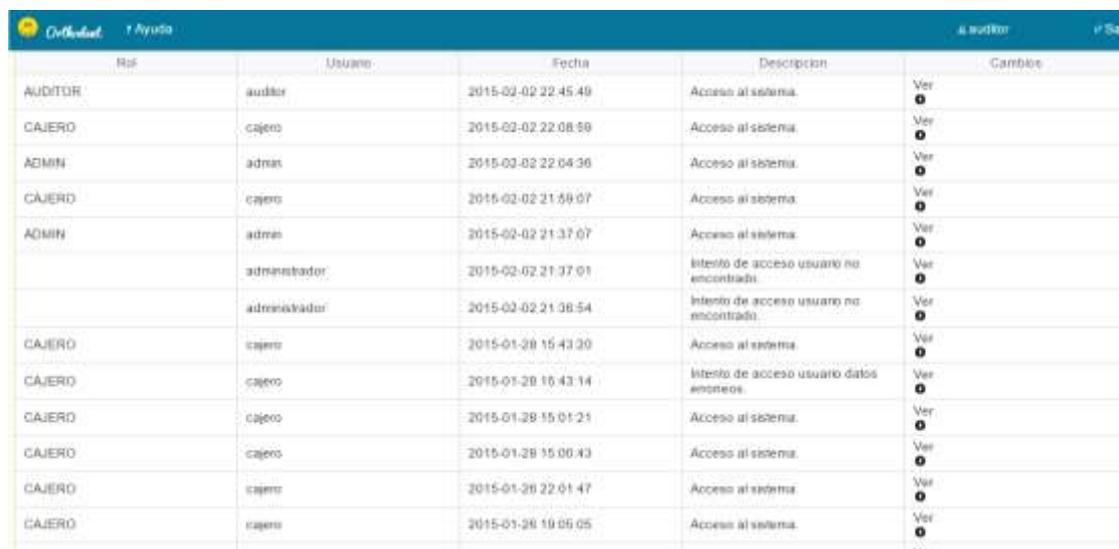
El botón imprimir si se lo da clic se enviara a imprimir la factura respectiva.



El botón que se encuentra de ayuda, llevara a este documento donde se encuentra la información importante por parte del cajero.

Auditor

Una vez ingresado como auditor, tendrá la oportunidad de revisar y analizar información pertinente acerca de los movimientos que se realizaron en el tiempo.

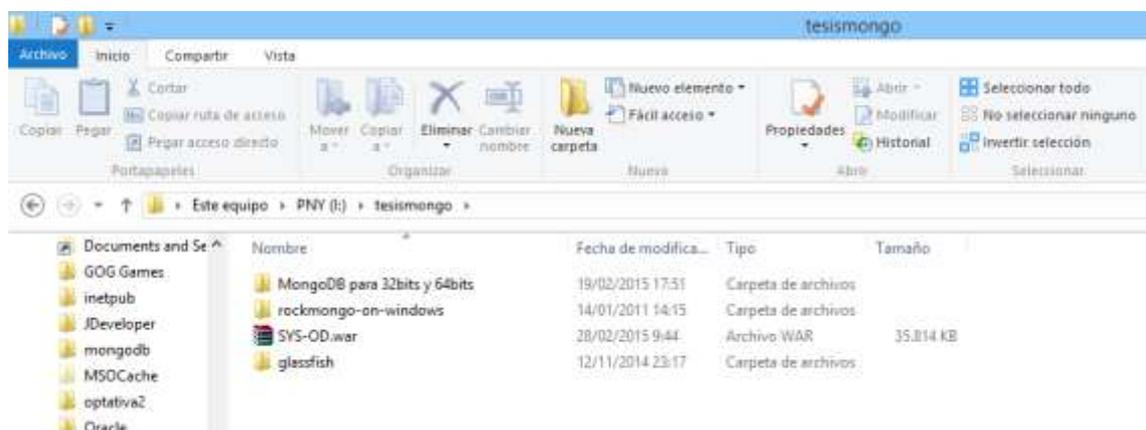


Ref	Usuario	Fecha	Descripción	Cambios
AUDITOR	auditor	2015-02-02 22:45:49	Acceso al sistema.	Ver
CAJERO	cajero	2015-02-02 22:08:59	Acceso al sistema.	Ver
ADMIN	admin	2015-02-02 22:04:36	Acceso al sistema.	Ver
CAJERO	cajero	2015-02-02 21:59:07	Acceso al sistema.	Ver
ADMIN	admin	2015-02-02 21:37:07	Acceso al sistema.	Ver
	administrador	2015-02-02 21:37:01	Intento de acceso usuario no encontrado.	Ver
	administrador	2015-02-02 21:38:54	Intento de acceso usuario no encontrado.	Ver
CAJERO	cajero	2015-01-29 15:43:30	Acceso al sistema.	Ver
CAJERO	cajero	2015-01-29 15:43:14	Intento de acceso usuario datos erróneos.	Ver
CAJERO	cajero	2015-01-29 15:01:21	Acceso al sistema.	Ver
CAJERO	cajero	2015-01-29 15:00:43	Acceso al sistema.	Ver
CAJERO	cajero	2015-01-29 22:01:47	Acceso al sistema.	Ver
CAJERO	cajero	2015-01-29 19:05:05	Acceso al sistema.	Ver

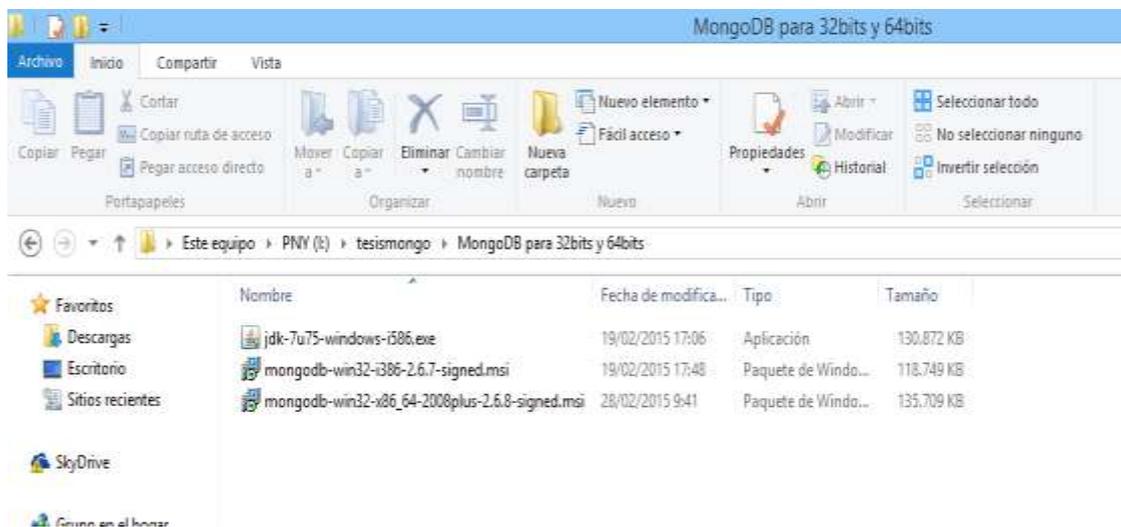
A continuación se detalla cómo instalar las herramientas que se necesitaran para el sistema en cuestión.

Copiar el Software del Sistema

En el CD que se entrega a la Clínica se encuentra con una carpeta con todo el software que necesita para hacer uso del sistema, por lo que tiene que copiarlos y pegarlos al escritorio, estas dos carpetas incluyendo el sistema SYS-OD.



En la carpeta de MongoDB para 32 bits y 64 bits se encuentra el instalador de la base de datos y del complemento JDK que se necesitara para ejecutarlo. Por lo que se debe instalar dependiendo del sistema operativo instalado.

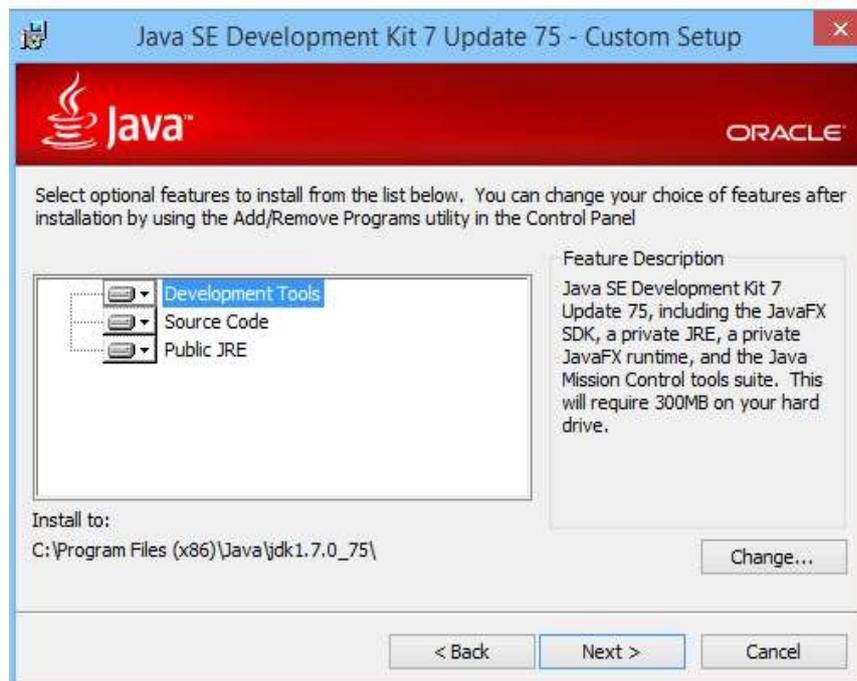


INSTALACIÓN DEL COMPLEMENTO JDK

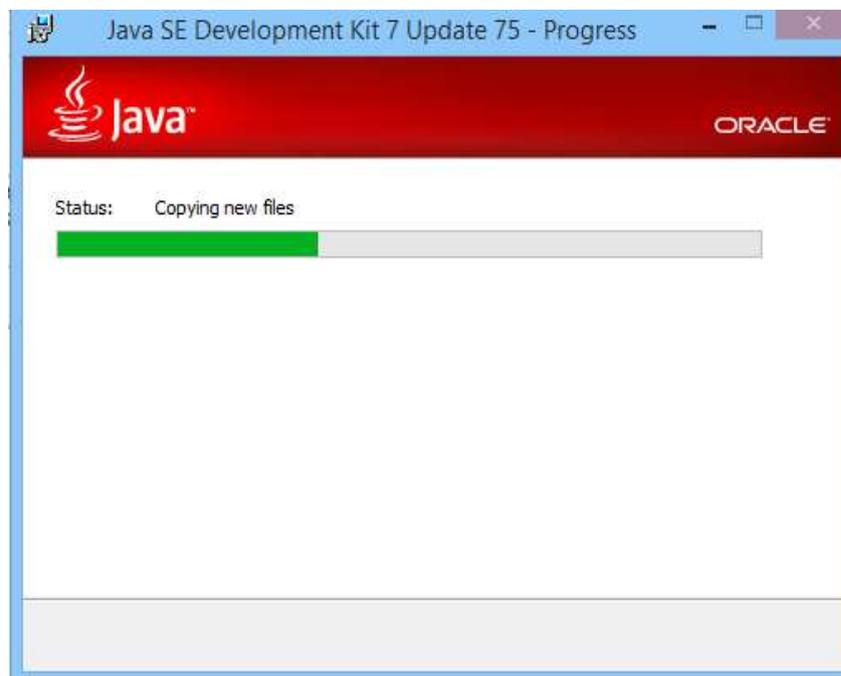
Para instalar el JDK, se le da doble clic y se acepta la instalación, se pone en continuar (Next).



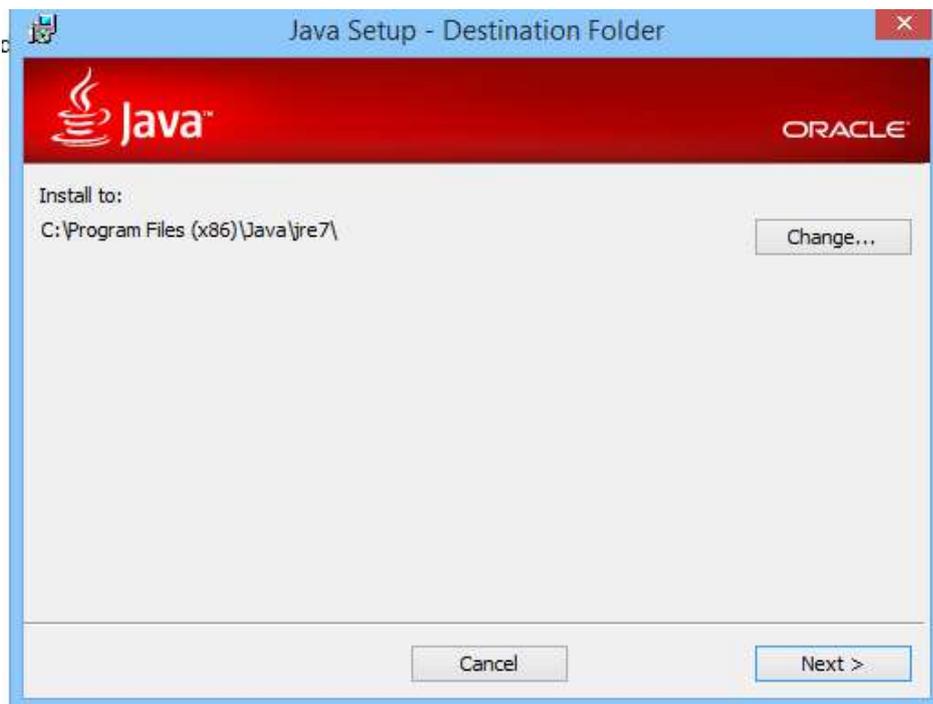
Se indica dónde se va a instalar por lo que se deja como defecto y da clic a continuar (Next)



Se deja instalar automáticamente los paquetes necesarios para poder usar el sistema.



Se deja por defecto el lugar de instalación y se pone en continuar (Next).



Se deja instalar automáticamente el complemento.



Se termina la instalación y se da clic a cerrar (close).



Instalación Base de Datos MongoDB

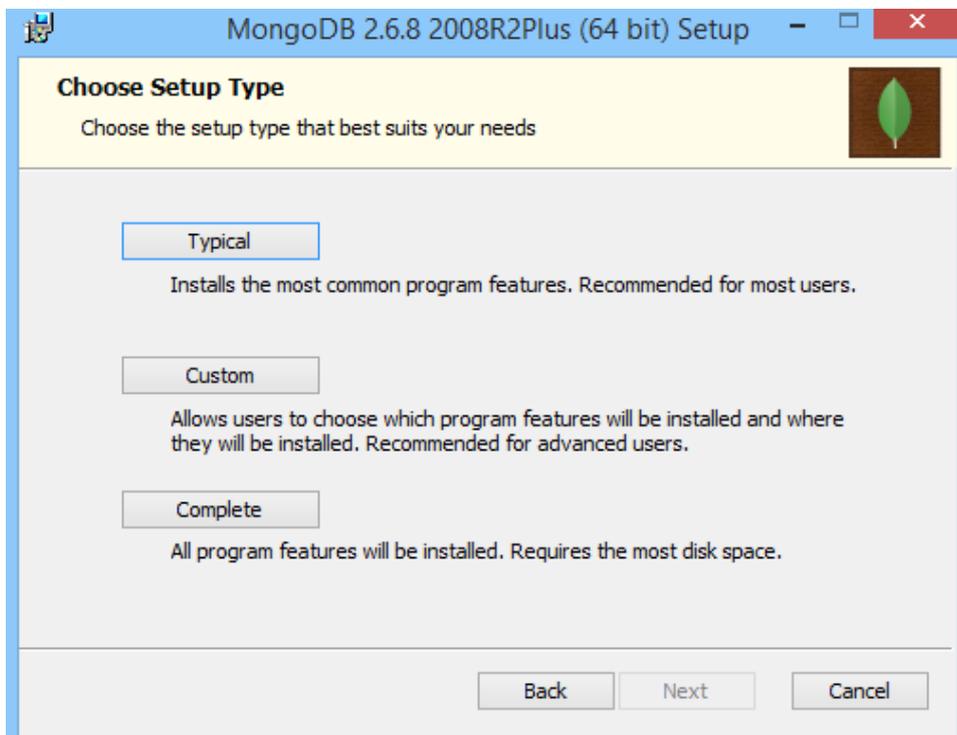
Para la instalación de la base de datos MONGODB se da doble clic a ejecutable .msi que se encuentra en la carpeta de los instaladores dependiendo del sistema operativo se encontrara para 32 bits y 64 bits por lo que se le da doble clic para iniciar el instalador por lo que se le da clic en siguiente (Next).



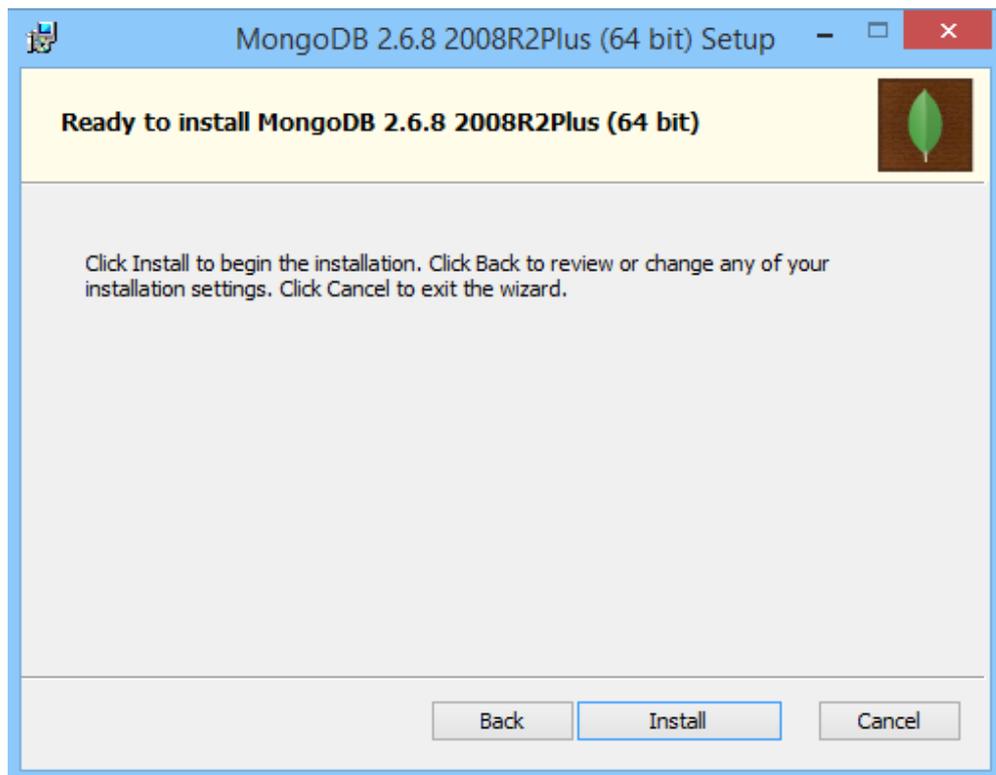
Se acepta la licencia libre de MongoDB dando un clic en el recuadro y se continuará (Next)



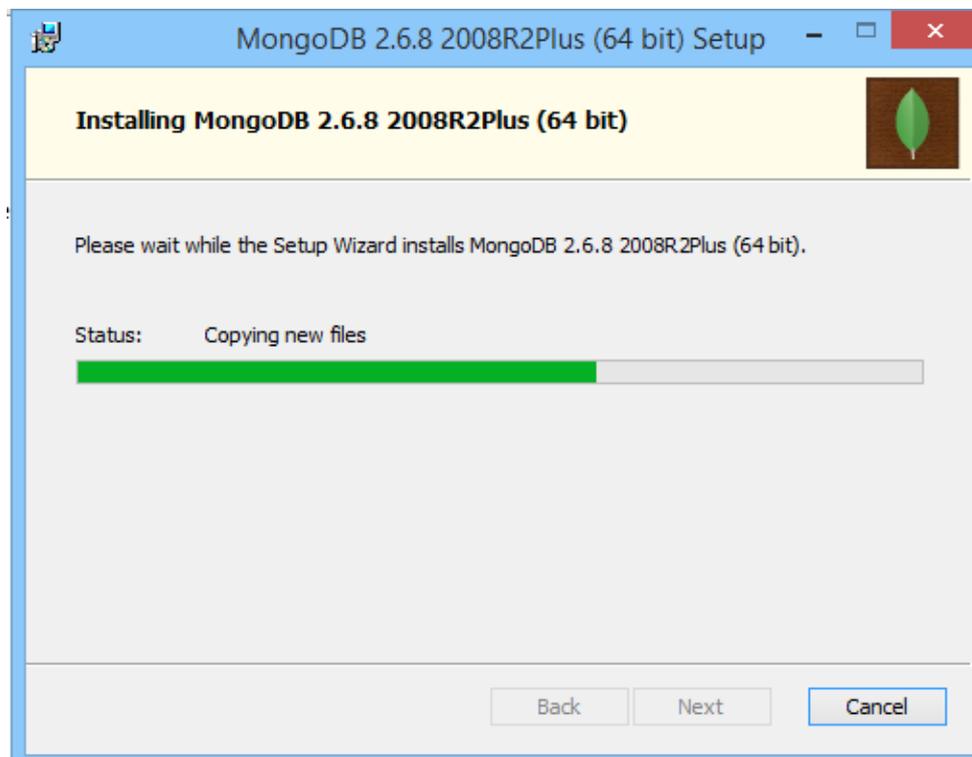
Se dará clic en el modo de instalación típica (Typical) de la base de datos MongoDB



Se dará clic en instalar (Install) por lo que se instalara la base de datos MongoDB



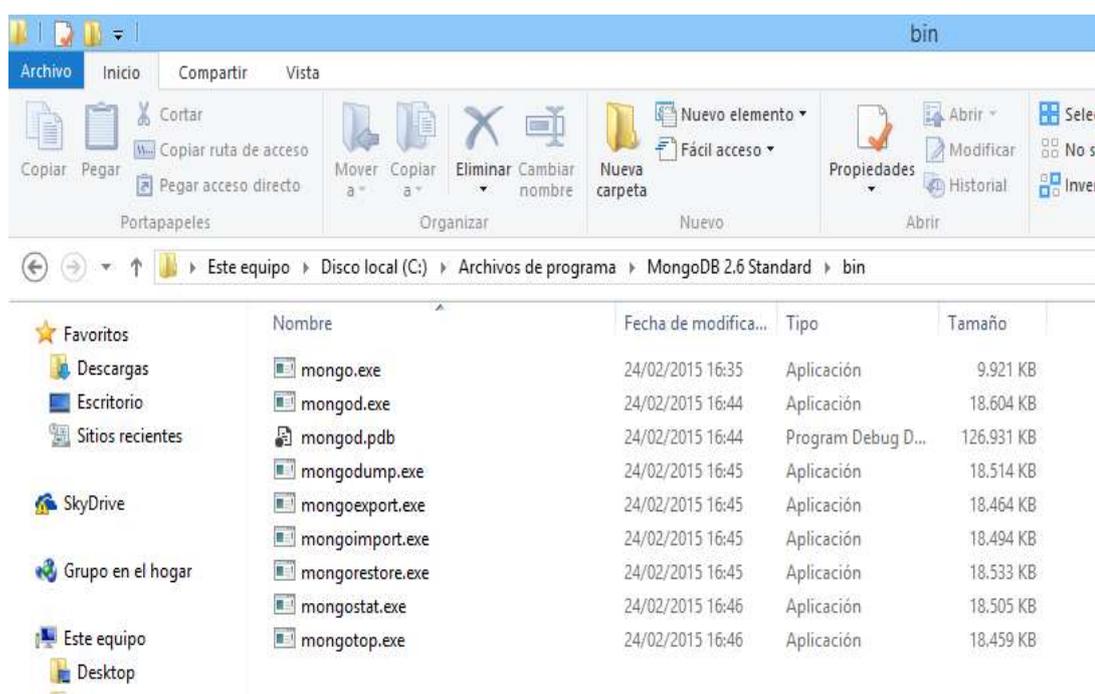
Se espera a que la instalación se complete



Una vez finalizada se da clic en terminar (Finish)



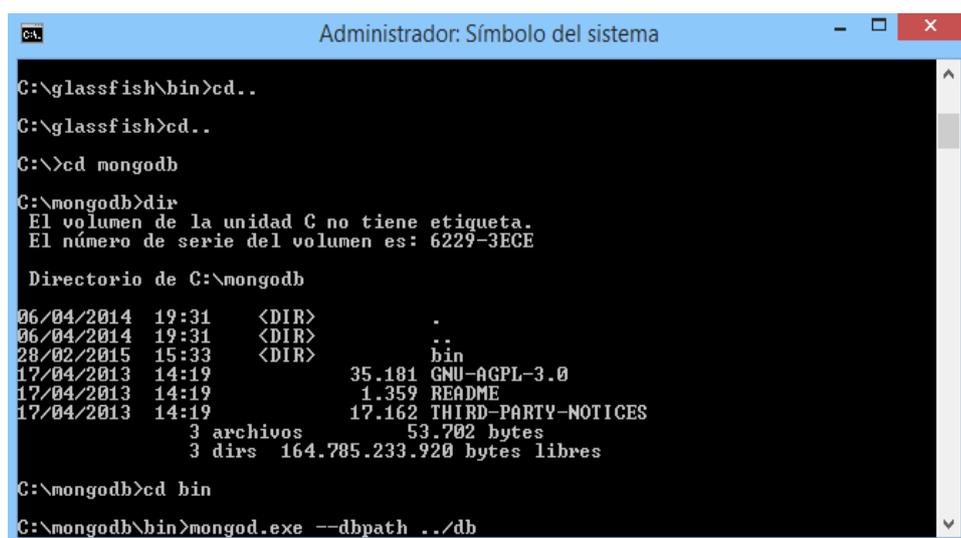
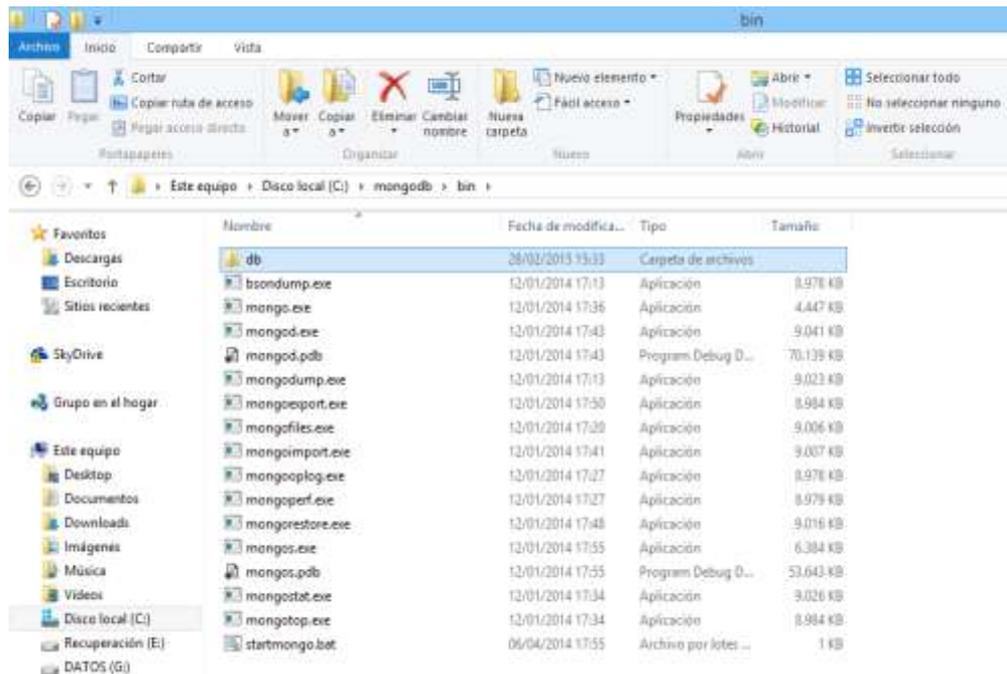
Una vez instalada se la debe ejecutar para que inicie la base de datos MongoDB por lo que hay que dirigirse a la dirección C:\Program Files\MongoDB 2.6 Standard\bin.



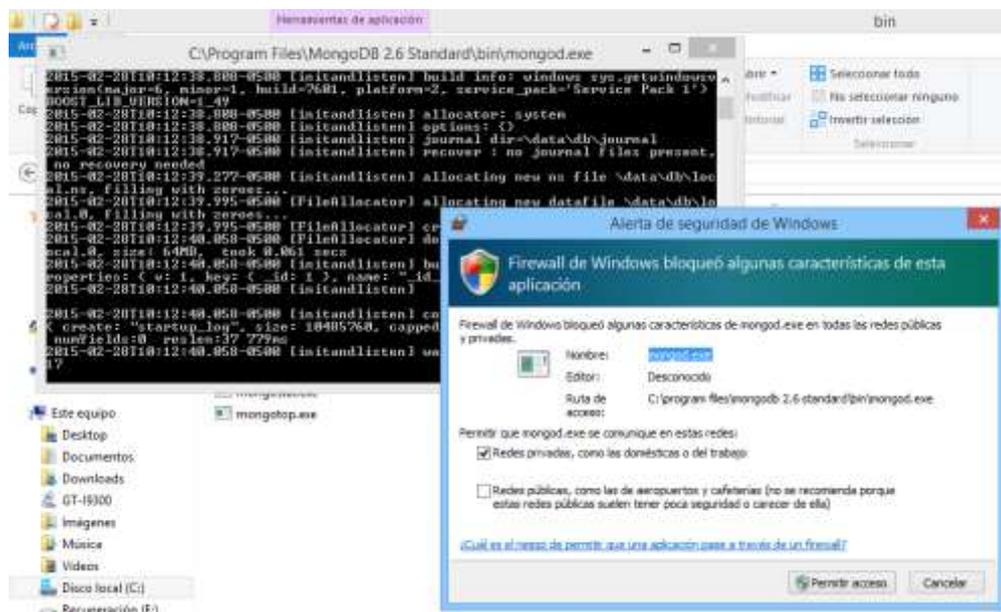
En esta dirección se encontraran los ejecutables para la base de datos MongoDB por lo que solo se debe crear una carpeta en la carpeta bin llamada db y luego de esto abrir el un cmd para agregar la siguiente línea de comandos en la dirección

C:\mongodb\bin\

mongod.exe --dbpath ../db

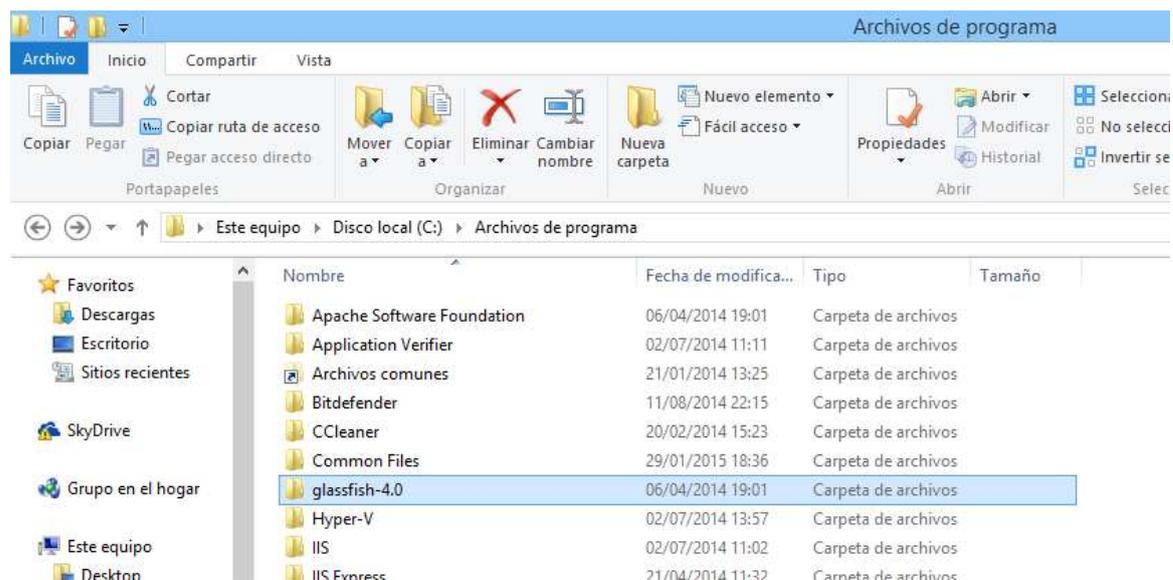


Al firewall o cortafuegos aparecerá indicando una alerta por lo que solo e debe dar clic en permitir acceso para que se ejecute normalmente.

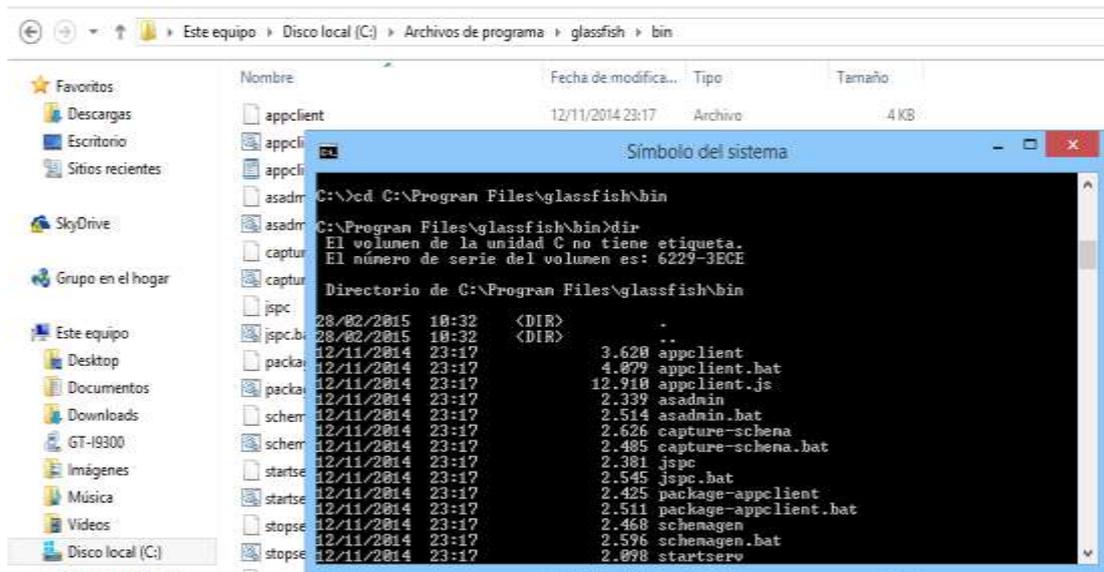


Instalación del Servidor Glassfish

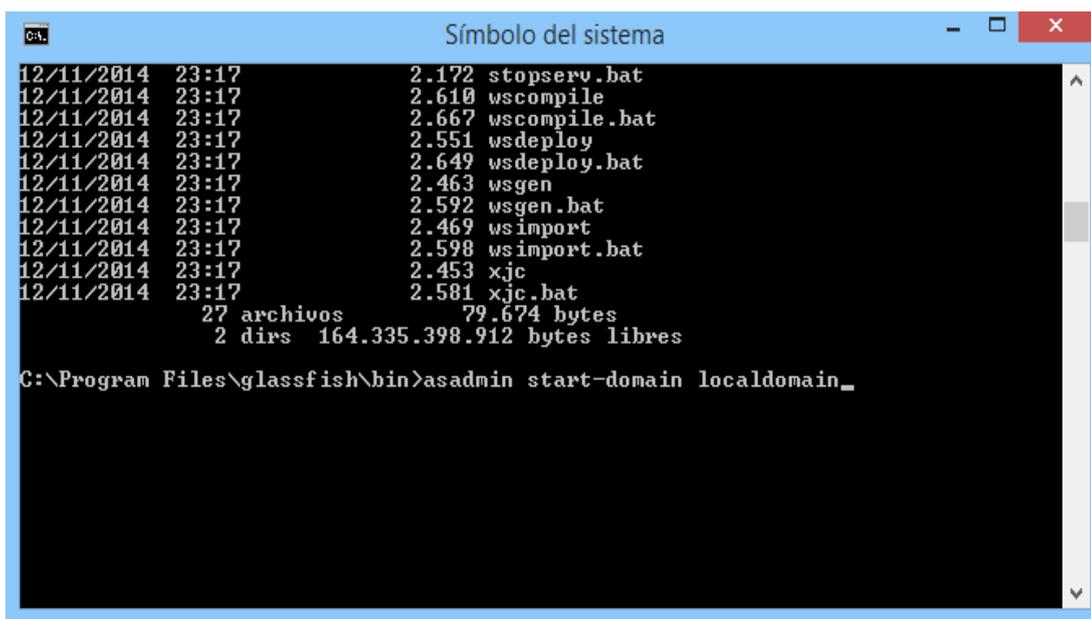
Ya ejecutado la base de datos MongoDB como el complemento de java JDK, procedemos a instalar el servidor de aplicaciones de Glassfish por lo que en la carpeta Glassfish se encuentra el servidor por lo que se lo debe copiar y pegar en C:/Program Files (C:/Archivos de programa)



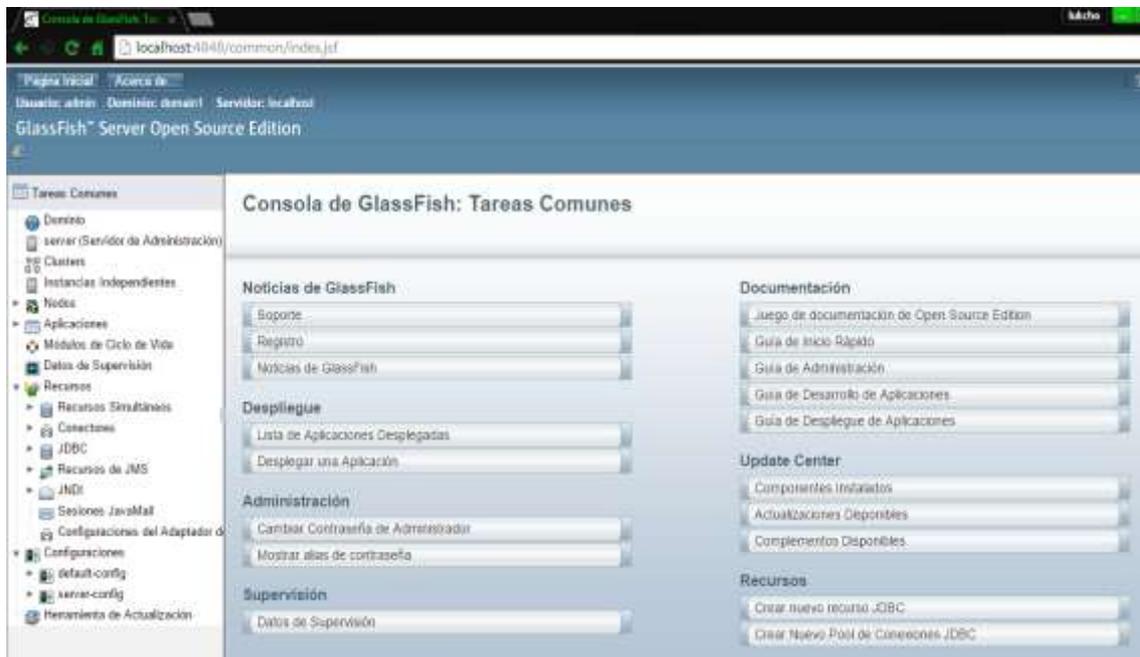
Hecho esto se lo debe inicializar, para esto se debe ingresar al símbolo del sistema (cmd) para poder ejecutarlo; por lo que nos dirigiremos a la dirección donde se instaló Glassfish por lo que debemos ejecutar el siguiente comando `cd C:\Program Files\glassfish\bin`



Luego de encontrarnos en la carpeta de Glassfish debemos ejecutar el comando siguiente `asadmin start-domain localdomain` para que se ejecute servidor de Glassfish



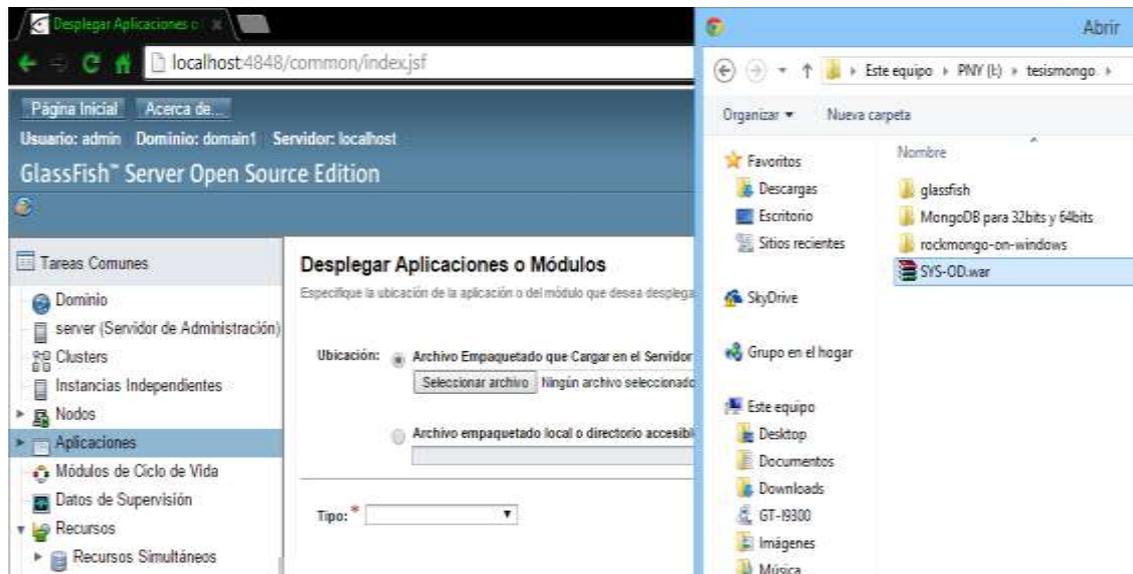
Una vez ejecutado el GlassFish debemos ir a un navegador de preferencia Google Chrome e ir a la siguiente dirección localhost: 4848 por lo que nos trasladara al servidor de aplicaciones de Glassfish



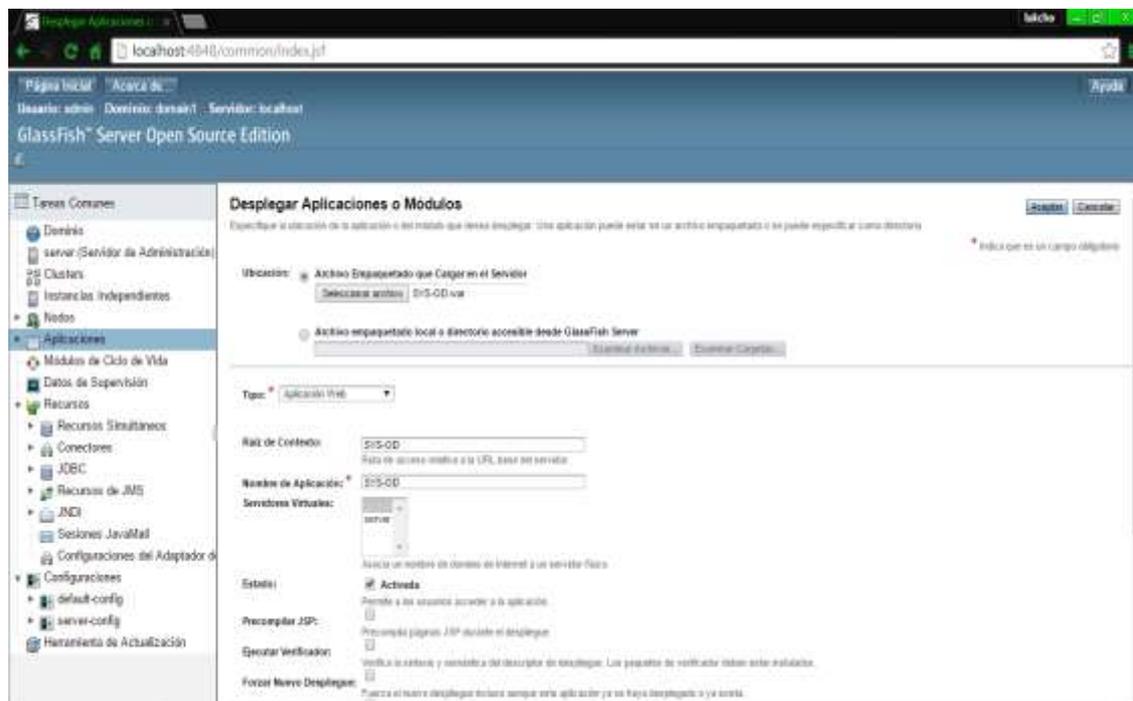
Nos dirigiremos a aplicaciones q se encuentra en las tareas comunes de Glassfish luego de esto daremos clic a desplegar (Deploy)



Debemos en ubicación, seleccionar el archivo .war del sistema SYS-OD por lo que se le da clic y buscar el archivo del sistema en los archivos entregados por lo que lo seleccionamos.



Puesto que se ha seleccionado, se dará clic a aceptar y el sistema estará completamente listo para ser usado.



Para estar seguros que se encuentra en el servidor se puede observar como esta creado en aplicaciones, luego de esto se encuentra distintas acciones por lo que se le da clic en Iniciar.



Después de esto se observara dos links que nos dirigirán al sistema SYS-OD por lo que se le da clic solo al primer enlace.

[Server] <http://luchito:8080/sys-od>



Después de esto se ingresara a la página principal del sistema

