

INSTITUTO TECNOLOGICO
"CORDILLERA"

CARRERA DE ANALISIS DE SISTEMAS

CONTROLAR LOS PROCESOS DE TELEMERCADEO Y VENTAS, A TRAVÉS DE UN SISTEMA INFORMÁTICO WEB DE GESTIÓN Y DESPACHO PARA LA COMPAÑÍA ATIEMPOFFICE.

Proyecto de investigación previo a la obtención del título de Analista de Sistemas.

Autor: Punguil Paucar José Ignacio.

Tutor: Ing. Johnny Coronel

Quito, Noviembre 2013

DECLARATORIA

Declaro que la investigación es absolutamente original, autentica, personal, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes. Las ideas, doctrinas resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

Punguil Paucar José Ignacio

CC: 172352131-4

CONTRATO DE CESIÓN SOBRE DERECHOS PROPIED AD INTELECTUAL

Comparecen a la celebración del presente contrato de cesión y transferencia de derechos de propiedad intelectual, por una parte, el estudiante José Ignacio Punguil Paucar, por sus propios y personales derechos, a quien en lo posterior se le denominará el "CEDENTE"; y, por otra parte, el INSTITUTO SUPERIOR TECNOLÓGICO CORDILLERA, representado por su Rector el Ingeniero Ernesto Flores Córdova, a quien en lo posterior se lo denominará el "CESIONARIO". Los comparecientes son mayores de edad, domiciliados en esta ciudad de Quito Distrito Metropolitano, hábiles y capaces para contraer derechos y obligaciones, quienes acuerdan al tenor de las siguientes cláusulas:

PRIMERA: ANTECEDENTE.- a) El Cedente dentro del pensum de estudio en la carrera de análisis de sistemas que imparte el Instituto Superior Tecnológico Cordillera, y con el objeto de obtener el título de Tecnólogo en Análisis de Sistemas, el estudiante participa en el proyecto de grado denominado "controlar los procesos de telemercadeo y ventas, a través de un sistema informático web de gestión y despacho para la compañía Atiempoffice", el cual incluye la creación y desarrollo del programa de ordenador o software, para lo cual ha implementado los conocimientos adquiridos en su calidad de alumno. **b)** Por iniciativa y responsabilidad del Instituto Superior Tecnológico Cordillera se desarrolla la creación del programa de ordenador, motivo por el cual se regula de forma clara la cesión de los derechos de autor que genera la obra literaria y que es producto del proyecto de grado, el mismo que culminado es de plena aplicación técnica, administrativa y de reproducción.

SEGUNDA: CESIÓN Y TRANSFERENCIA.- Con el antecedente indicado, el Cedente libre y voluntariamente cede y transfiere de manera perpetua y gratuita

todos los derechos patrimoniales del programa de ordenador descrito en la clausula anterior a favor del Cesionario, sin reservarse para sí ningún privilegio especial (código fuente, código objeto, diagramas de flujo, planos, manuales de uso, etc.). El Cesionario podrá explotar el programa de ordenador por cualquier medio o procedimiento tal cual lo establece el Artículo 20 de la Ley de Propiedad Intelectual, esto es, realizar, autorizar o prohibir, entre otros: a) La reproducción del programa de ordenador por cualquier forma o procedimiento; b) La comunicación pública del software; c) La distribución pública de ejemplares o copias, la comercialización, arrendamiento o alquiler del programa de ordenador; d) Cualquier transformación o modificación del programa de ordenador; e) La protección y registro en el IEPI el programa de ordenador a nombre del Cesionario; f) Ejercer la protección jurídica del programa de ordenador; g) Los demás derechos establecidos en la Ley de Propiedad Intelectual y otros cuerpos legales que normen sobre la cesión de derechos de autor y derechos patrimoniales.

TERCERA: OBLIGACIÓN DEL CEDENTE.- El cedente no podrá transferir a ningún tercero los derechos que conforman la estructura, secuencia y organización del programa de ordenador que es objeto del presente contrato, como tampoco emplearlo o utilizarlo a título personal, ya que siempre se deberá guardar la exclusividad del programa de ordenador a favor del Cesionario.

CUARTA: CUANTIA.- La cesión objeto del presente contrato, se realiza a título gratuito y por ende el Cesionario ni sus administradores deben cancelar valor alguno o regalías por este contrato y por los derechos que se derivan del mismo.

QUINTA: PLAZO.- La vigencia del presente contrato es indefinida.

SEXTA: DOMICILIO, JURISDICCIÓN Y COMPETENCIA.- Las partes fijan como su domicilio la ciudad de Quito. Toda controversia o diferencia derivada de éste, será resuelta directamente entre las partes y, si esto no fuere factible, se

solicitará la asistencia de un Mediador del Centro de Arbitraje y Mediación de la Cámara de Comercio de Quito. En el evento que el conflicto no fuere resuelto mediante este procedimiento, en el plazo de diez días calendario desde su inicio, pudiendo prorrogarse por mutuo acuerdo este plazo, las partes someterán sus controversias a la resolución de un árbitro, que se sujetará a lo dispuesto en la Ley de Arbitraje y Mediación, al Reglamento del Centro de Arbitraje y Mediación de la Cámara de comercio de Quito, y a las siguientes normas: a) El árbitro será seleccionado conforme a lo establecido en la Ley de Arbitraje y Mediación; b) Las partes renuncian a la jurisdicción ordinaria, se obligan a acatar el laudo arbitral y se comprometen a no interponer ningún tipo de recurso en contra del laudo arbitral; c) Para la ejecución de medidas cautelares, el árbitro está facultado para solicitar el auxilio de los funcionarios públicos, judiciales, policiales y administrativos, sin que sea necesario recurrir a juez ordinario alguno; d) El procedimiento será confidencial y en derecho; e) El lugar de arbitraje serán las instalaciones del centro de arbitraje y mediación de la Cámara de Comercio de Quito; f) El idioma del arbitraje será el español; y, g) La reconvención, caso de haberla, seguirá los mismos procedimientos antes indicados para el juicio principal.

SÉPTIMA: ACEPTACIÓN.- Las partes contratantes aceptan el contenido del presente contrato, por ser hecho en seguridad de sus respectivos intereses.

En aceptación firman a los 12 días del mes de Noviembre del dos mil doce.

f)_____ f)_____

C.C. Nº 172352131-4 Instituto Superior Tecnológico Cordillera

CEDENTE **CESIONARIO**

AGRADECIMIENTO

Gracias a todo el personal docente del Instituto Tecnológico Superior Cordillera, por sus enseñanzas ya que eso fue lo que me ayudo a poder realizar y terminar el presente proyecto.

DEDICATORIA

A mi hermano por su gran apoyo incondicional, por hacerme quien soy en estos momentos y por lo más sagrado que tengo en la vida.

ÍNDICE GENERAL

CONTENIDO **PÁGINAS**

- Declaración de aprobación del Tutor y Lector…….....…...I
- Declaración de autoría……...……....……II
- Declaración de Cesión de Derechos a la Institución……....……...III
- Agradecimiento……..…....…..…VI
- Dedicatoria……..……...….VII
- Índice General…….....…VIII
- Índice de Tablas…….....XIII
- Índice de Figuras…….……...XIV
- Resumen Ejecutivo…….....XVIII
- Abstract……...……...……..XX

Capítulo I: Antecedentes

1.01 Contexto.....1

1.02 Justificación.....2

1.03 Definición del problema central matriz T.....3

Capítulo II: Involucrados

2.01 Involucrados e intereses.....4

2.02 Mapeo de involucrados.....5

2.03 Matriz de análisis de involucrados.....5

2.04 Clasificación de involucrados.....6

Capítulo III: Árbol de problemas - Objetivos

3.01 Árbol de problemas.....7

3.02 Análisis árbol de problemas.....8

3.03 Árbol de objetivos.....9

3.04 Análisis árbol de objetivos.....10

Capítulo IV: Análisis de alternativas

4.01 Matriz de análisis de alternativas.....11

4.02 Matriz de análisis de impacto de objetivos.....12

4.03 Diagrama de estrategias.....13

4.04 Matriz marco lógico.....14

Capítulo V: Desarrollo del sistema

5.01 Justificación técnica.....15

5.02 Análisis y diseño.....16

5.02.01 Diagrama de casos de uso.....16

5.02.02 Diagrama de realización registro cliente.....17

5.02.03 Diagrama de realización registro producto.....18

5.02.04 Diagrama de realización entrega información de clientes.....19

5.02.05 Diagrama de realización gestión telemercadeo.....20

5.02.06 Diagrama de realización proforma.....21

5.02.07 Diagrama de realización facturar.....22

5.02.08 Diagrama de realización genera venta.....23

5.02.09 Diagrama de realización reportes.....24

5.02.10 Diagrama de realización registro nuevo usuario.....25

5.02.11 Diagrama de secuencia cliente.....26

5.02.12 Diagrama de secuencia productos.....27

5.02.13 Diagrama de secuencia información cliente.....28

5.02.14 Diagrama de secuencia telemercadeo.....29

5.02.15 Diagrama de secuencia proformas.....30

5.02.16 Diagrama de secuencia facturas.....31
5.02.17 Diagrama de secuencia venta.....32
5.02.18 Diagrama de secuencia reportes.....33
5.02.19 Diagrama de secuencia usuarios.....34
5.02.20 Diagrama de colaboración cliente.....35
5.02.21 Diagrama de colaboración telemercadeo.....36
5.02.22 Diagrama de colaboración proforma.....37
5.02.23 Diagrama de colaboración factura.....38
5.02.24 Diagrama de colaboración venta.....39
5.02.25 Diagrama de colaboración reporte.....40
5.02.26 Diagrama de colaboración usuario.....41
5.02.27 Diagrama de componentes.....42
5.02.28 Modelo lógico.....43
5.02.29 Modelo físico.....44
5.03 Desarrollo.....45
5.03.01 Arquitectura de desarrollo.....45
5.03.01.01 Capa de presentación.....45
5.03.01.02 Capa lógica de negocio.....45
5.03.01.03 Capa acceso a datos.....45
5.04 Estándares de diseño UML.....46
5.05 Estándares de programación.....47
5.05.01 Nombres para módulos de formulario y de clase.....47
5.05.02 Nombre para los controles.....47
5.05.03 Estándares de objetos .NET.....49
5.05.04 Estándares y prefijos de variables.....49

5.05.05 Estándares tipo de datos.....50

5.05.06 Estándares de base de datos.....52

5.06 Diseño e interfaces.....55

5.06.01 Interfaz inicio de sesión(login) 55

5.06.02 Interfaz usuario administrador.....56

5.06.03 Interfaz usuario vendedor.....57

5.06.04 Interfaz mantenimiento de productos.....58

5.06.05 Interfaz mantenimiento de clientes.....59

5.06.06 Interfaz creación de proformas.....60

5.06.07 Interfaz creación de usuarios.....61

5.06.08 Interfaz creación de Facturas.....62

5.07 Casos de prueba.....63

5.07.01 Técnicas de prueba.....63

5.07.02 Fundamentos de la prueba de software.....63

5.07.03 Objetivos de las pruebas de software.....63

5.07.04 Pruebas registro de nuevos usuarios.....64

5.07.05 Pruebas registro de nuevos clientes.....65

5.07.06 Pruebas asignación de clientes a vendedor.....66

5.07.07 Pruebas gestión de telemercadeo.....67

5.07.08 Pruebas generar proformas.....68

5.07.09 Pruebas generar facturas.....69

5.07.10 Pruebas registro nuevos usuarios.....70

5.07.11 Pruebas generar reportes.....71

Capítulo VI: Aspectos administrativos

6.01 Recursos.....72

6.02 Presupuesto.....72

6.03 Cronograma.....72

Capítulo VII: Conclusiones y recomendaciones

7.01 Conclusiones.....73

7.02 Recomendaciones.....74

Anexos

Anexo 1 Manual técnico.....76

Anexo 1.01 Instalación SQL Server 2008.....76

Anexo 1.02 Instalación Visual Studio 2010.....88

Anexo 2 Código Fuente.....97

Anexo 3 Cronograma de actividades.....112

Anexo 4 Manual de Usuario.....113

ÍNDICE DE TABLAS

TABLAS **PAGINAS**

Tabla 1 Análisis fuerzas T.....3
Tabla 2 Involucrados e intereses.....4
Tabla 3 Matriz de análisis de involucrados.....5
Tabla 4 Matriz de análisis de alternativas.....11
Tabla 5 Matriz de análisis de impacto.....12
Tabla 6 Matriz de marco lógica.....14
Tabla 7 Descripción objetos UML.....46
Tabla 8 Descripción nombres de controladores.....48
Tabla 9 Descripción objetos ADO.....49
Tabla 10 Descripción estándares de variables.....49
Tabla 11 Tipos de datos.....50
Tabla 12 Descripción de variables especiales.....51
Tabla 13 Características tipos de datos.....52
Tabla 14 Caso de pruebas registro de nuevo clientes.....64
Tabla 15 Caso de prueba registro de nuevos productos.....65
Tabla 16 Asignación de clientes a vendedores.....66
Tabla 17 Caso de prueba gestión de telemercadeo..... 67
Tabla 18 Caso de prueba generar proformas.....68
Tabla 19 Caso de prueba generar facturas.....69
Tabla 20Caso de prueba registrar nuevos usuarios.....70
Tabla 21 Caso de prueba generar reportes.....71
Tabla 22 Descripción detalle presupuesto.....72

ÍNDICE DE FIGURAS

FIGURAS **PÁGINAS**

Figura 1.....5
Figura 2.....6
Figura 3.....7
Figura 4.....9
Figura 5.....13
Figura 6.....16
Figura 7.....17
Figura 8.....18
Figura 9.....19
Figura 10.....20
Figura 11.....21
Figura 12.....22
Figura 13.....23
Figura 14.....24
Figura 15.....25
Figura 16.....26
Figura 17.....27
Figura 18.....28
Figura 19.....29
Figura 20.....30
Figura 21.....31
Figura 22.....32

Figura 23.....33
Figura 24.....34
Figura 25.....35
Figura 26.....36
Figura 27.....37
Figura 28.....38
Figura 29.....39
Figura 30.....40
Figura 31.....41
Figura 32.....42
Figura 33.....43
Figura 34.....44
Figura 35.....55
Figura 36.....56
Figura 37.....57
Figura 38.....58
Figura 39.....59
Figura 40.....60
Figura 41.....61
Figura 42.....62
Figura 43.....76
Figura 44.....77
Figura 45.....77
Figura 46.....78
Figura 47.....78

Figura 48.....79
Figura 49.....79
Figura 50.....80
Figura 51.....80
Figura 52.....81
Figura 53.....82
Figura 54.....82
Figura 55.....83
Figura 56.....84
Figura 57.....84
Figura 58.....85
Figura 59.....85
Figura 60.....86
Figura 61.....86
Figura 62.....87
Figura 63.....88
Figura 64.....89
Figura 65.....89
Figura 66.....90
Figura 67.....90
Figura 68.....91
Figura 69.....92
Figura 70.....93
Figura 71.....93
Figura 72.....94

Figura 73.....94
Figura 74.....95
Figura 75.....96
Figura 76.....112
Figura 77.....113
Figura 78.....114
Figura 79.....114
Figura 80.....115
Figura 81.....115
Figura 82.....116
Figura 83.....117
Figura 84.....118
Figura 86.....120
Figura 87.....120
Figura 88.....122
Figura 89.....122

Resumen Ejecutivo

El presente proyecto, esta oriento a satisfacer las necesidades de la empresa Atiempoffice y a los procesos de ventas referentes, los cuales presentan fallas tales como: procesos lentos, falta de integración, falta de información oportuna para toma de decisiones y la obtención de reportes, información no confiable, lo que originó un inadecuado control de procesos de telemercadeo y ventas, mala atención al cliente y bajos ingresos económicos para la empresa.

En el primer capítulo contiene lo relacionado con el problema como: el contexto, Justificación y la definición del problema, en la justificación se analiza el problema de la empresa, causas, efectos; razón que se realiza la investigación para encontrar soluciones al problema planteado.

En el segundo capítulo se analiza los posibles involucrados en el proyecto ya que de una forma directa o indirecta pueden verse afectados.

En el tercer capítulo se estudia los problemas centrales que afectan directamente a la empresa para poder obtener una conclusión de los mismos.

En el cuarto comprende una mayor perspicacia de los problemas que se plantea analizando la relación causa efecto y los posibles impactos de objetivos.

En el quinto capítulo se presenta una justificación técnica del proyecto, a su vez se muestra análisis y diseño del sistema con sus respectivos diagramas e interfaces que ayudaran a dar solución a la necesidad presente.

En el capítulo sexto muestra detalladamente el recurso, presupuesto y cronograma de actividades que se utilizó para el desarrollo e implementación del sistema.

En el capítulo séptimo se presentan las conclusiones y recomendación desde el punto de vista del autor, con el propósito de cubrir de manera eficiente y eficaz las necesidades y problemas planteados en el presente proyecto.

Abstract

This project , this oriented to meet the needs of the company Atiempoffice and sales processes relating, which have faults such as slow processes , lack of integration , lack of timely information for decision making and obtaining reporting, unreliable information, resulting in inadequate control of telemarketing and sales processes, poor customer service and low income for the company.

The main objective and scope of this project is to propose a control system web telemarketing and sales, in order to overcome the deficiencies generated by the current manual control of the company.

The expected outcomes of this project are: the generation of proformas, send proformas, invoice generation, reporting and indicators to facilitate decision making within the administrative área, for the project definition was based on a previous study of the present situation and the needs presented in the study area of the company.

CAPÍTULO I

1.01. Contexto

El telemercadeo es un instrumento de la mercadotecnia directa que ha venido cobrando mayor relevancia en los últimos años, debido a que aprovecha las nuevas tecnologías de la información y las aplica al desarrollo empresarial, el telemercadeo ahorra costosas visitas personales, se contactan más prospectos al día y se mantiene satisfechos a los clientes, con base en una buena atención, y la oportunidad de respuesta ante la competencia.

La compañía Atiempoffice ubicada en la provincia de pichincha, en la zona 9 dentro del plan nacional de desarrollo del buen vivir desarrolla su actividad comercial a la industria de tecnología en relojes biométricos de huella digital y electromecánicos, direccionada dentro da las normas, reglamentos, procedimientos, organizacionales y gubernamentales.

El control inadecuado en los procesos de telemercadeo y ventas por parte de los diferentes entes de la compañía no es eficiente ni eficaz produciendo bajo rendimiento laboral, es necesario añadir el desconocimiento de los empleados sobre los procesos a seguir para cumplir con el trabajo asignado.

Con dichos antecedentes que de alguna manera requieren el sustento tecnológico para mejorar el control de los procesos de ya mencionados, para ello es indispensable proporcionar una herramienta que facilite y garantice e l cumplimiento del trabajo asignado al personal, reduciendo la problemática, reduciendo el bajo desempeño laboral y controlando los procesos de telemercadeo y ventas.

1.02. Justificación

Actualmente el telemercadeo es una de las herramientas más prácticas que puede tener una organización para tener contacto con el cliente, interesarlo, motivarlo y atraerlo, dejando una buena imagen de la empresa, es importante tener presente que para tener un buen resultado se necesita un buen equipo de trabajo comprometido, que asuma retos y sobre todo que exista un control de los procesos de telemercadeo y ventas para tener a nuestro alcance y de manera ordenada toda la información que el usuario requiera, sin que este proceso signifique para las personas una actividad tediosa, con pérdida de tiempo, a su vez evaluar el desempeño laboral de los empleados mediante resultados obtenidos de acuerdo a las diferentes reglas del negocio y reportes que se podrán obtener de manera rápida, sencilla con información verdadera y confiable.

1.03. Definición del problema central matriz T

Tabla 1.

Análisis Fuerzas T

Escenario empeorado	Situación actual				Escenario mejorado
Empresa en quiebra por falta de ingresos económicos	Inadecuado de control de procesos de telemercadeo y ventas				Mayor control en los procesos de telemercadeo y ventas
	I	**P**	**I**	**P**	
Fuerzas impulsadoras		**C**		**C**	Fuerzas Bloqueadoras
Capacitaciones sobre telemercadeo y ventas	3	3	4	4	La empresa no interviene en capacitaciones a los empleados
Experiencia en telemercadeo y ventas	3	3	4	4	Desconocimiento en el área
Cumplir con las reglas establecidas	3	4	5	4	No cumplir con el trabajo asignado
Reportes en tiempo real	3	3	4	4	Información errónea de los procesos de telemercadeo y ventas
Optimizar tiempos de trabajo	3	3	4	4	Pérdida de tiempo de trabajo

CAPÍTULO II

2.01. Listado de involucrados e intereses

Tabla 2.

Involucrados e intereses.

No.	Involucrado	Interés
1	Empleados 1	En desacuerdo por la exigencia a cumplir con su trabajo
2	Clientes	Mejorar la atención
3	Propietarios	Controlar el cumplimiento de labores asignadas y generar más ingresos
4	Gerentes	Controlar los procesos de telemercadeo y ventas
5	Socios	Controlar el cumplimiento de labores asignadas y generar más ingresos
6	Supervisores	Automatizar los procesos de telemercadeo y ventas
7	Empleados 2	Apoyo por automatizar procesos y reducción de tiempo en las mismas

2.02. Mapeo de involucrados

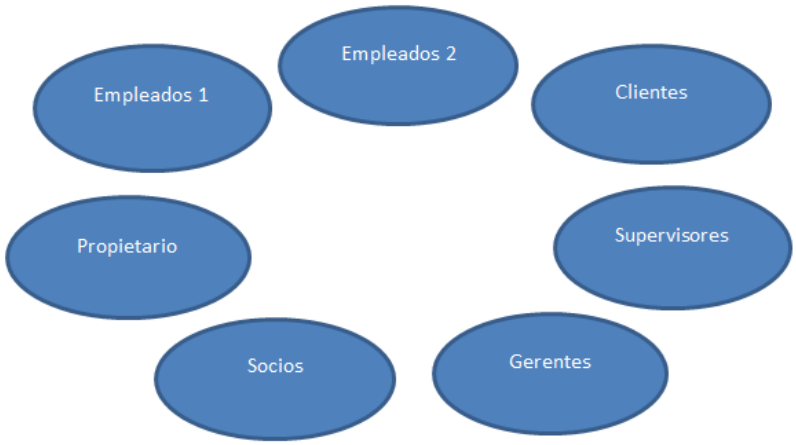


Figura 1. *Mapeo de involucrados*

Diagrama que representa de forma gráfica el mapeo de involucrados en el proyecto a realizar.

2.03. Matriz de análisis de involucrados

Tabla 3.

Matriz de análisis de involucrados

No.	Involucrados	Expectativa	Fuerza	Resultante	Posición P
1	Propietario	4	5	20	Favorecedores
2	Socios	4	5	20	Favorecedores
3	Gerentes	3	4	12	Favorecedores
4	Supervisores	3	3	9	Favorecedores
5	Clientes	2	2	4	Neutrales
6	Empleados 2	3	2	6	Neutrales
7	Empleados 1	-3	2	-6	Opositores

2.04. Clasificación de involucrados

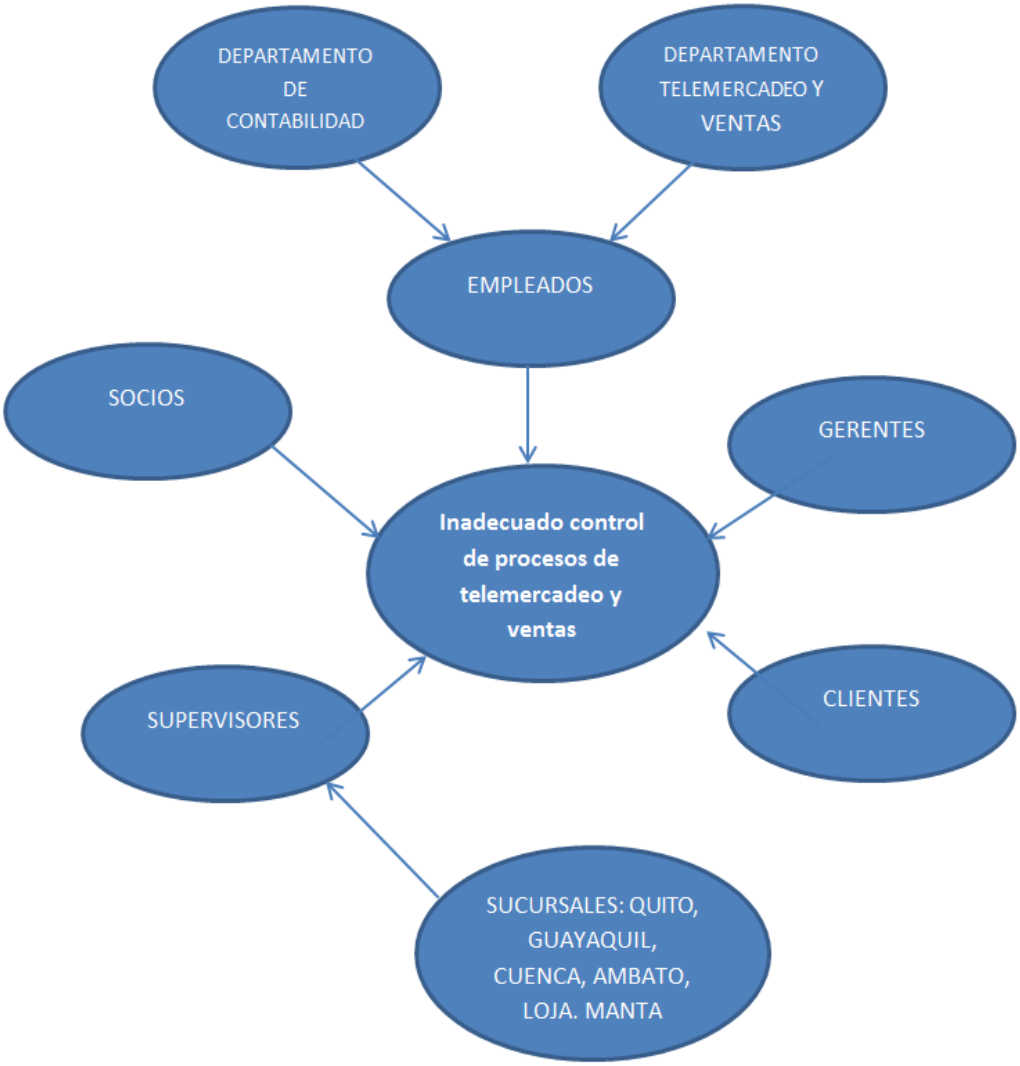


Figura 2. *Clasificación de involucrados*

Describe el mapeo de involucrados en forma clasificada de acuerda a las actividades o procesos que realiza cada involucrado.

CAPÍTULO III

3.01. Análisis árbol de problemas

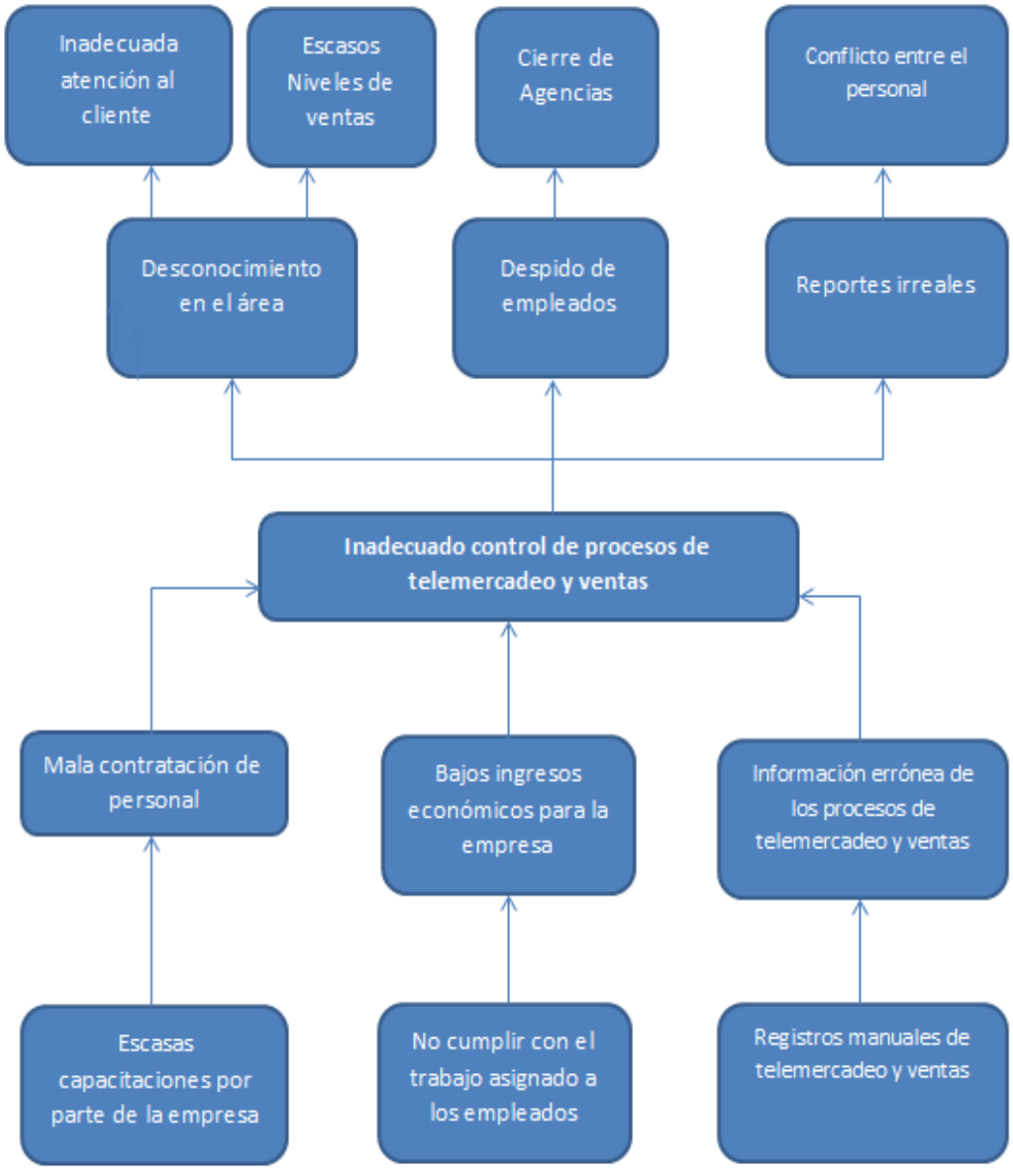


Figura 3. *Árbol de problemas*

Describe en forma gráfica los problemas, causas y efectos.

3.02. Análisis

El inadecuado control de los procesos de telemercadeo y ventas, la mala administración del departamento, también por la información errónea de todos los procesos registrados manualmente, provocan que el personal no cumpla con sus labores asignadas, que se generen bajos ingresos económicos para la empresa, impidiendo que la empresa continúe y que posiblemente haya despidos de empleados o hasta incluso cierre de sucursales.

3.03. Análisis árbol de objetivos

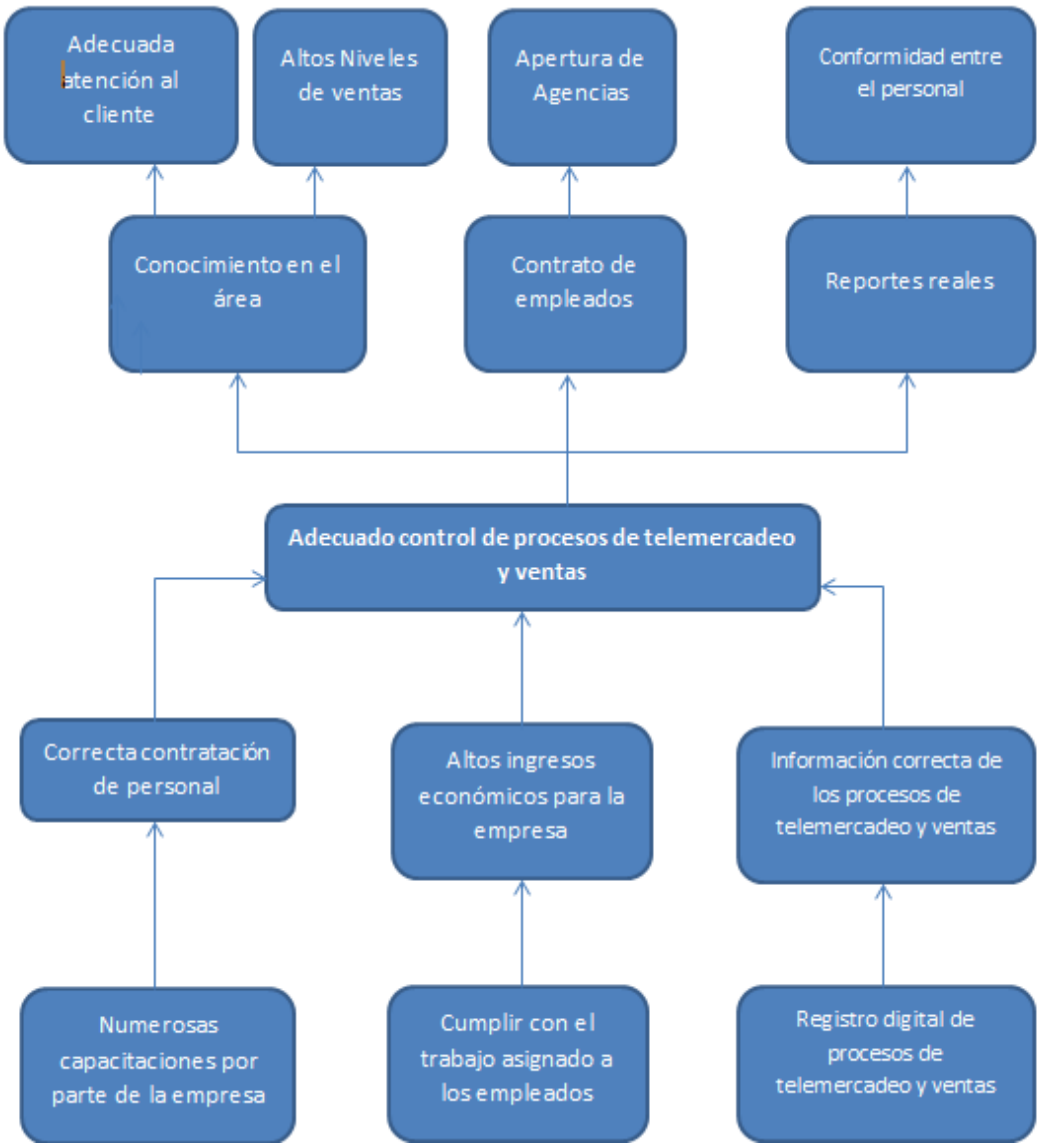


Figura 4. *Árbol de objetivos*

Describe de forma gráfica el árbol de problemas convertido en objetivos o los problemas que se encontraban en forma negativa se transformaron en positiva.

3.04. Análisis

El tener un adecuado control de los procesos de telemercadeo y ventas conlleva a que existan altos ingresos económicos en la empresa, contratación de personal y hasta incluso poder liderar el mercado de la comercialización en lo que la empresa desempeña sus labores.

CAPÍTULO IV

4.01. Matriz de análisis de alternativas

Escala de evaluación:

1 = Bajo, 2 = Medio Bajo, 3 = Medio Alta, 4 = Alta

Tabla 4.

Matriz de análisis de alternativas

Matriz de análisis de alternativas

Objetivos	Impacto propósito	Factibilidad técnica	Factibilidad financiera	Factibilidad social	T	Categoría
Altos ingresos económicos para la empresa	5	5	5	4	19	Alta
Información correcta de los procesos de telemercadeo y ventas	5	5	4	4	18	Alta
Conocimiento en el área	4	4	4	4	17	Media alta
Contratación de empleados	4	4	4	5	18	Alta
Reportes en tiempo real	4	5	5	4	19	Alta
Total	25	23	22	21	91	

4.02. Matriz de análisis impacto de objetivos

Tabla 5.

Matriz de análisis de impacto

Análisis impacto de objetivos

	Fact. de lograse (4 - 2 - 1)	Imp. Género (4 - 2 - 1)	Imp. ambiental (4 - 2 - 1)	Relevancia (4 - 2 - 1)	Sostenibilidad (4 - 2 - 1)	T
Generación de altos ingresos económicos para la empresa	Se cumplirá con el trabajo asignado a los empleados (4)	Incremento de eficiencia y eficacia de los trabajadores (4)	Contribución a la preservación del medio ambiente (4)	Cumple con las expectativas (4)	Fortalece el crecimiento de la organización (4)	20
Información correcta de los procesos de telemerca-deo y ventas	Se llevara un registro digital de los procesos de telemercadeo y ventas (4)	Incremento correcta verificación del grado de trabajo de los vendedo-res(4)	Baja utilización de papel (2)	Control interno de procesos de telemerca-deo y ventas (4)		14
Conocimiento en el área	Numerosas capacitaciones por parte de la empresa (4)	Personal capacitado para desempeñar sus labores diarias (4)		Beneficia al personal de ventas (2)	Eficiencia y eficacia en el departamento de ventas (4)	14
Contratación de empleados	Generación de altos ingresos a la empresa (4)	Generación de fuente de trabajo (4)	Favorece la educación ambiental (4)	Beneficia a personal en general (4)	Crecimiento y apertura de sucursales (4)	20

4.03. Diagrama de estrategias

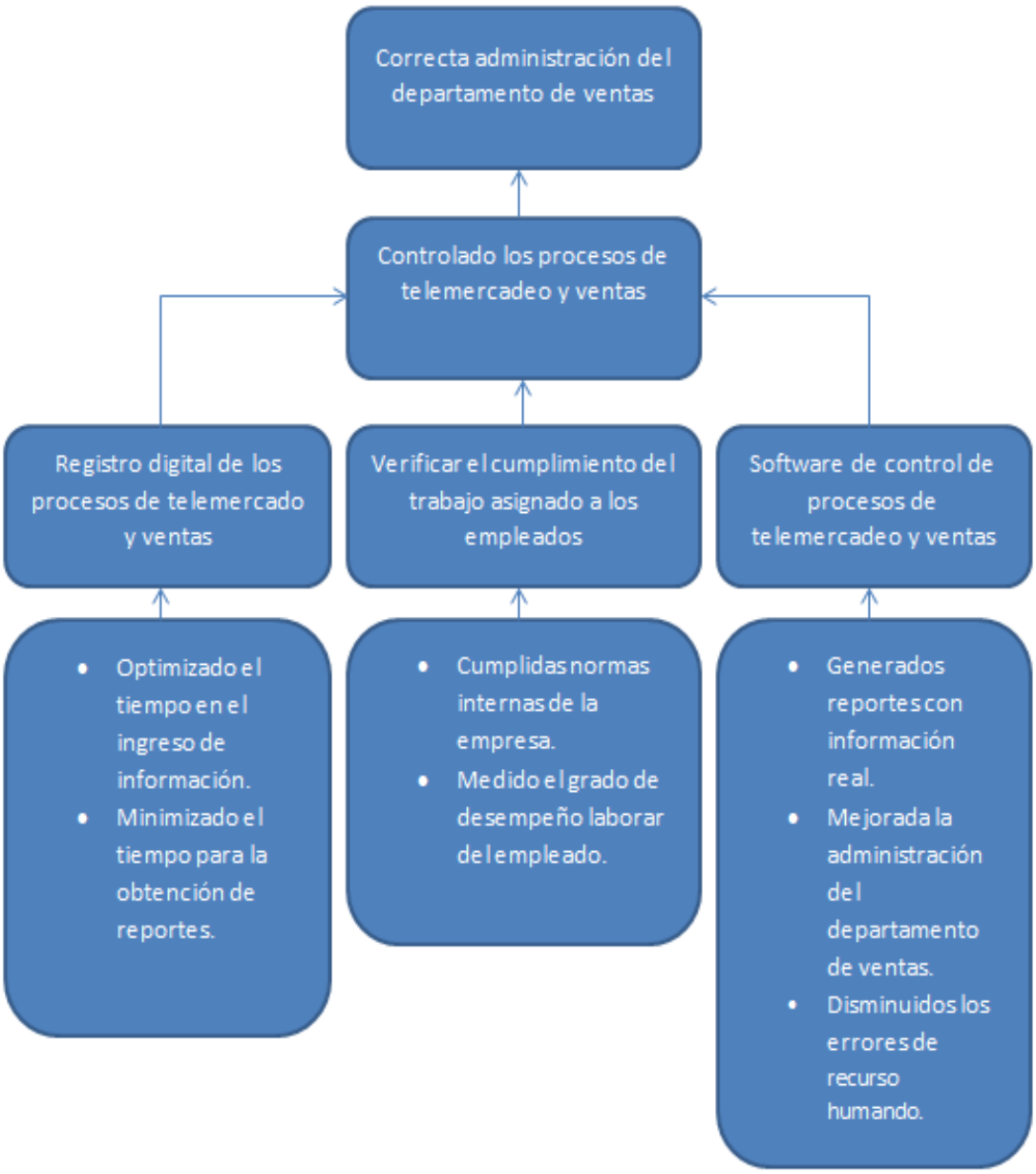


Figura 5. *Diagrama de estrategias*

En la figura actual describe gráficamente un diagrama de estrategias que se obtiene del árbol de objetivos.

4.04. Matriz de marco lógico

Tabla 6.

Matriz de marco lógico

Finalidad	Indicadores	Medios de verificación	Supuestos
Administración adecuada del departamento de ventas	Control en los procesos administrativos del departamento	Documentación digitalizada	
Propósito del proyecto control adecuado de los procesos de telemercadeo y ventas	Registro de información digital y confiable. Evitar ingreso erróneo de información	Documentación y reportes en tiempo real	
Componentes del proyecto adecuado control de procesos en el departamento de ventas	Información almacenada y clasificada	Documentación y reportes en tiempo real	

CAPÍTULO V

5.01. Justificación técnica

El presente proyecto está elaborado con el fin de controlar los procesos de telemercadeo y ventas, que se ejecutan de la compañía con el fin de optimizar los recursos, tiempos en la generación de reportes administrativos.

Para ello se aplica una metodología orientada a objetos que permita utilizar las características más importantes de POO como herencia, polimorfismo y encapsulamiento, el diseño está enfocado a un ambiente gráfico utilizando UML(Lenguaje de Modelamiento Unificado), lo cual ha permitido diagramar los procesos a automatizar de formas graficas haciendo que sea entendible para lectores, a su vez para modelar una base de datos orientada a objetos, para esto se ha utilizado diferentes tipos de diagrama los cuales se describen a continuación:

- Diagrama de Caso de Uso
- Diagrama de interacciones
- Diagrama de secuencias
- Diagrama de Colaboración
- Diagrama de componentes
- Modelo Lógico
- Modelo físico

Los diagramas anteriormente descritos ayudaran a visualizar las necesidades, lógica del negocio, así como las diferentes especificaciones para el desarrollo del sistema.

5.02. Análisis y diseño

5.02.01. Diagramas de casos de uso

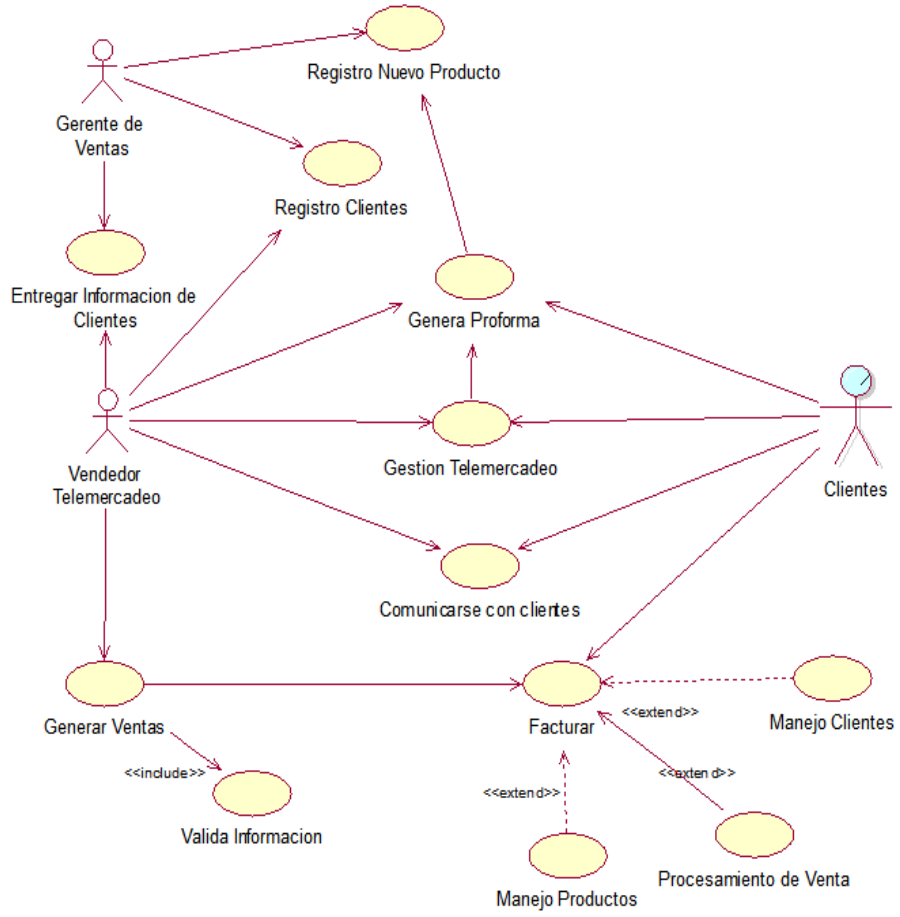


Figura 6. *Diagrama general de caso de uso*

Describe los procesos generales que se desea automatizar con el desarrollo del sistema.

5.02.02. Diagrama de realización Registro clientes

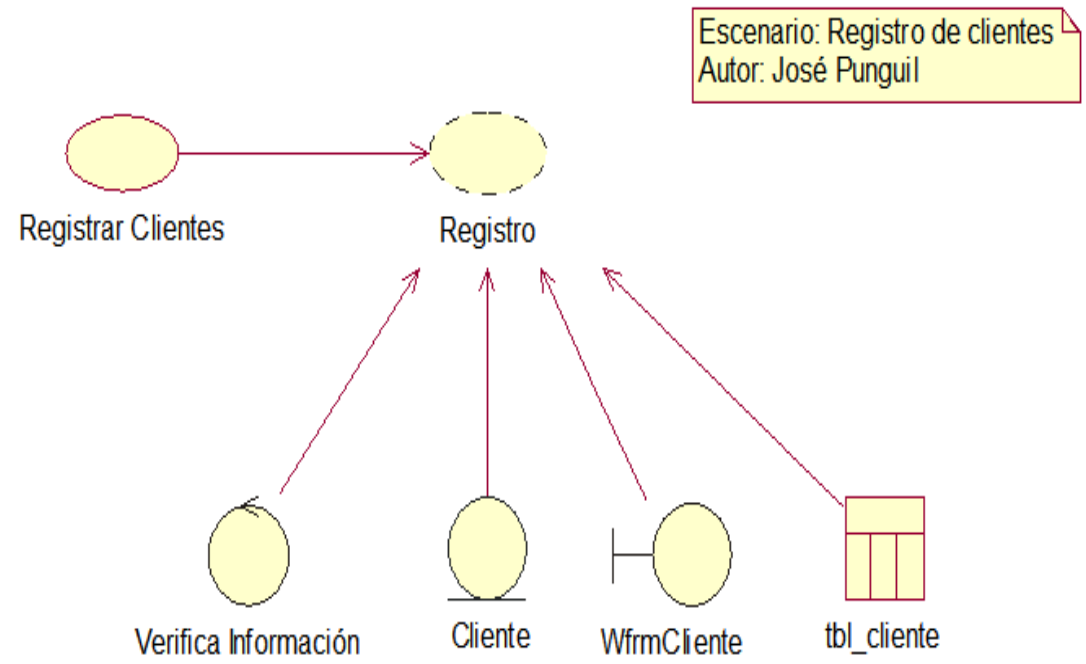


Figura 7. *Diagrama Registro de clientes*

En el presente diagrama interviene un actor el cual es el Gerente de ventas donde el objetivo es registrar un nuevo cliente. Para esto el gerente de ventas deberá estar logeado en el sistema, e ingresar en el formulario de clientes donde el software validara la información y guardara si todo es correcto de lo contrario nos mostrara un mensaje de error al guardar la información.

5.02.03. Diagrama de realización Registro de productos

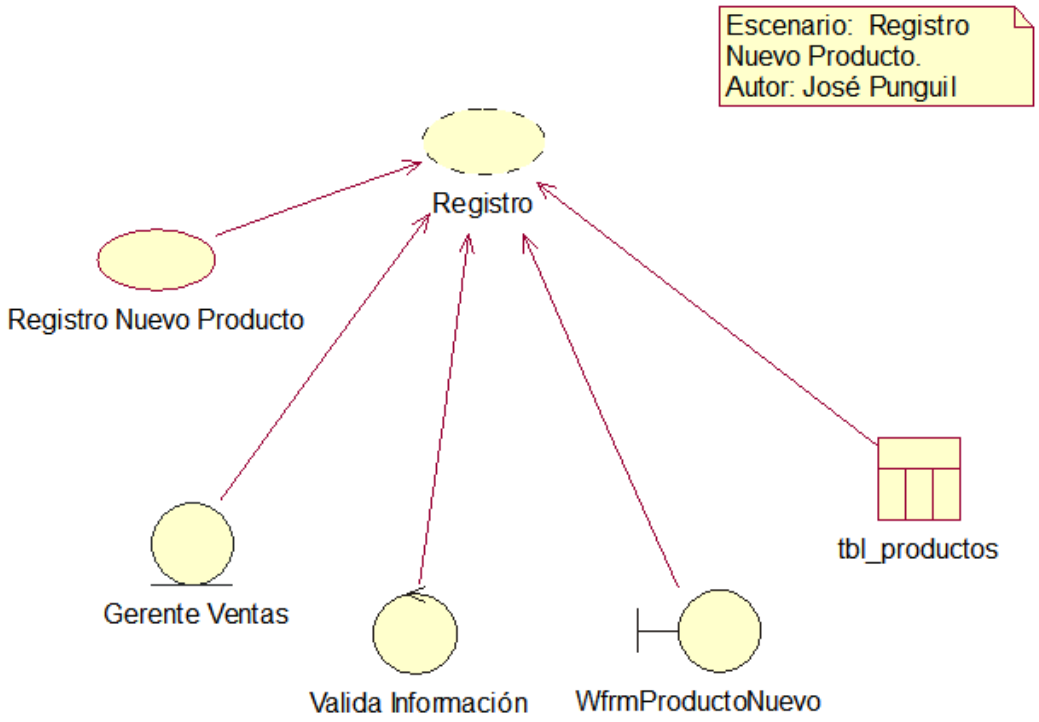


Figura 8. *Diagrama Registro de Productos*

En el presente diagrama interviene un actor el cual es el Gerente de ventas donde el objetivo es registrar un nuevo producto al sistema. Para esto el gerente de ventas deberá estar logeado en el sistema, e ingresar en el formulario de registro de nuevos productos, donde el software validara la información y guardara si todo es correcto de lo contrario nos mostrara un mensaje de error al guardar la información del nuevo producto.

5.02.04. Diagrama de realización Entrega información de clientes

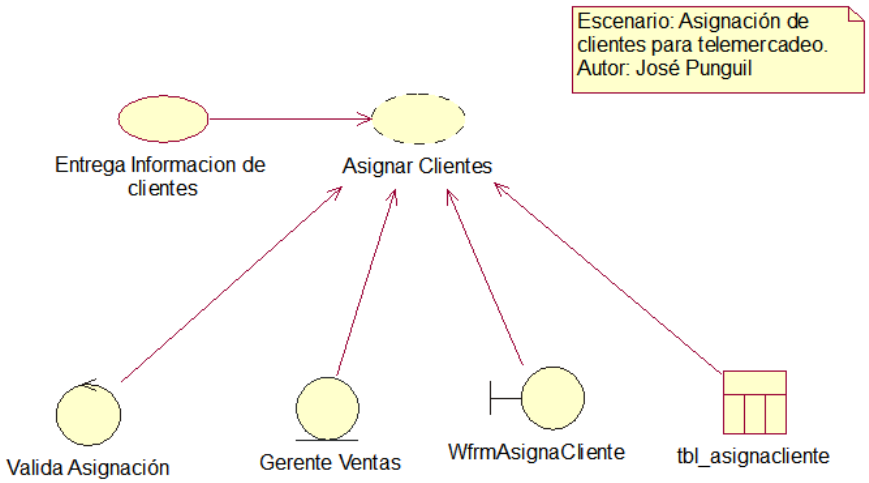


Figura 9. *Diagrama de entrega de información de clientes*

En este diagrama intervienen dos actores los cuales son el Gerente de ventas y el vendedor, en el cual el objetivo es asignar la información de un número determinado de clientes a los vendedores para tener un control de telemercadeo y poder cubrir la mayor demanda posible de los productos. Para esto el gerente de ventas deberá estar logeado en el sistema, e ingresar en el formulario de asignación de clientes para poder seleccionar los clientes y asignar a un determinado vendedor, el software validara la información asignada y mostrara un mensaje que informara que la asignación fue correcta de lo contrario nos mostrara un mensaje de error al realizar la asignación.

5.02.05. Diagrama de realización Gestión telemercadeo (llamar clientes)

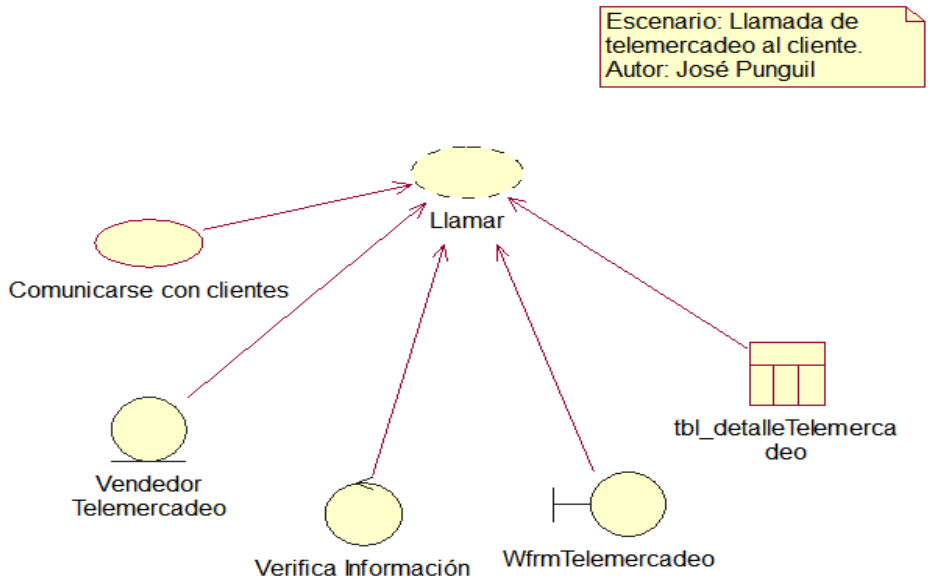


Figura 10. *Diagrama de gestión de telemercadeo*

En este diagrama intervienen dos actores los cuales son el vendedor y el cliente, en el cual el objetivo principal es comunicarse con el cliente previamente registrado en el sistema y ofrecer los productos que la empresa comercializa y a su vez tratar de obtener una cita para una demostración y poder cerrar una venta. Para esto el vendedor tendrá que verificar si la información del cliente asignada es verdadera. Una vez terminada la comunicación con el cliente, la información proporcionada será registrada en el sistema para posteriormente obtener diferentes reportes sobre telemercadeo, esto se realizara en la interfaz de telemercadeo.

5.02.06. Diagrama de realización Proforma

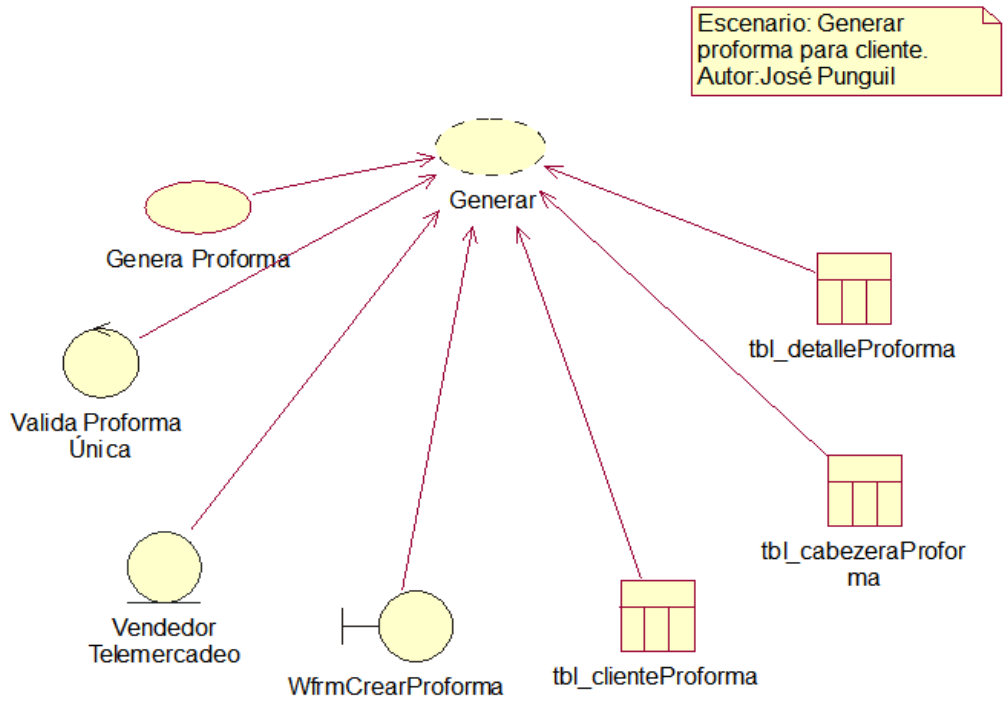


Figura 11. *Diagrama generación de proformas*

Este proceso nos ayuda a la elaboración de proformas en tiempo real, esto quiere decir que se podrá realizar mientras el vendedor se comunica con el cliente, esto ayudara a mostrar una eficiencia en el departamento de ventas cuando el vendedor envié la proforma al cliente, para realizar este proceso el vendedor deberá encontrarse logueado en el sistema dentro del formulario de elaboración de proformas, una vez llena la información de la proforma el software validara la información ingresada y guardara cuando todos los datos sean reales y correctos, de lo contrario mostrara un mensaje de error al guardar la información.

5.02.07. Diagrama de realización facturar

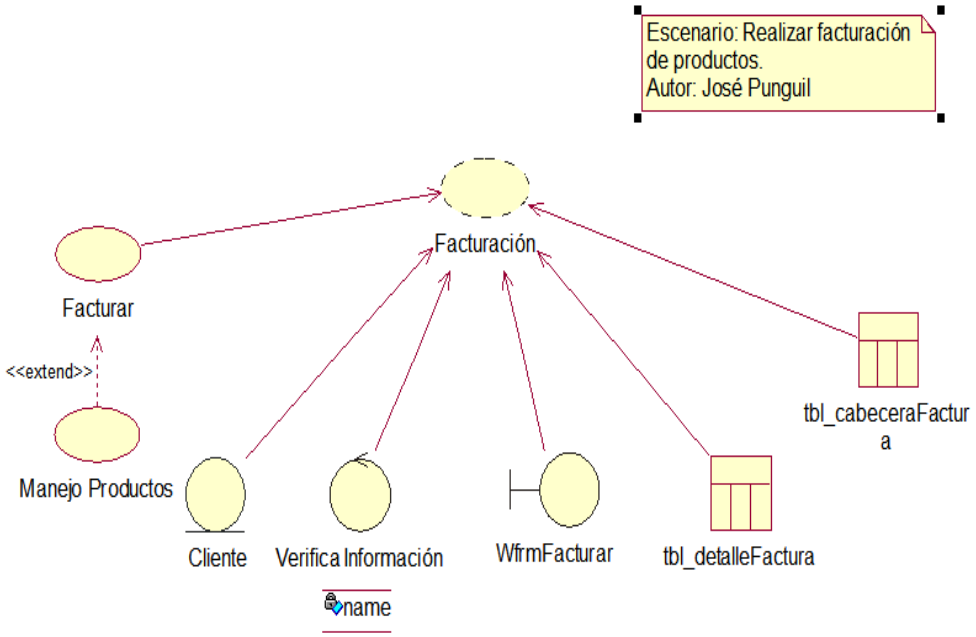


Figura 12. *Diagrama generación de facturas*

Este diagrama nos muestra el proceso de facturación que lo realizara cada vendedor dependiendo sus ventas, para esto los vendedores estarán logeados en el sistema donde utilizaran el formulario de facturación, en esto el sistema realizara la validación de información respectiva y se podar dar paso a guardar la factura y poderla imprimir, registrando todo este proceso en la base de datos para posteriormente poder utilizar dicha información.

5.02.08. Diagrama de realización generar venta

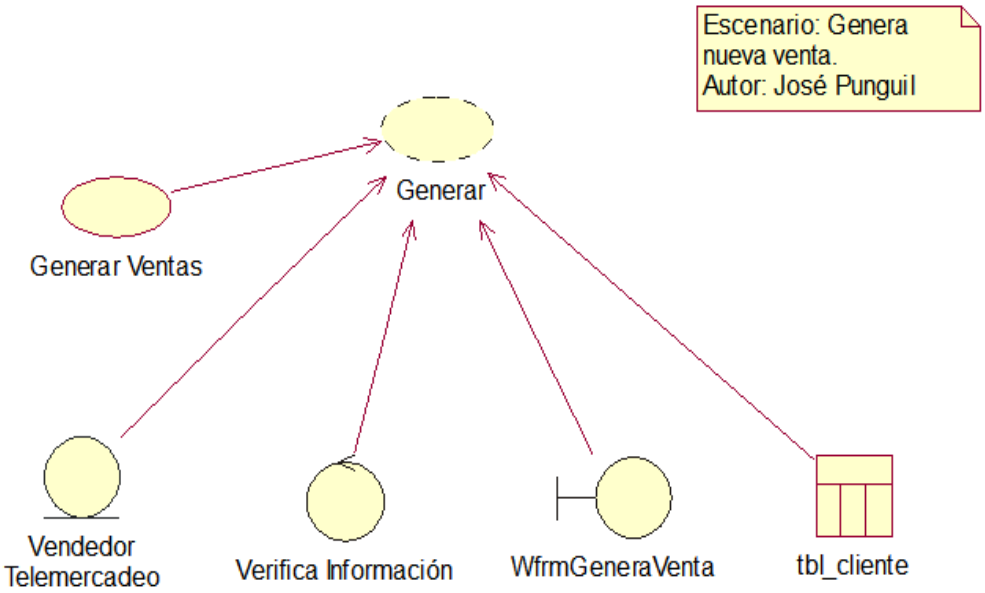


Figura 13. *Diagrama generación de venta*

Este diagrama representa la generación de la venta en el cual se utilizara la interfaz de generación de ventas donde el vendedor ingresara la información pertinente en el sistema la cual será validada, en este proceso el vendedor previo a un telemercadeo realizado será considerado como una llamada saliente y generadora de una venta, de lo contrario no se considerara como venta generada, esta información será utilizada para fines administrativos.

5.02.09. Diagrama de realización reportes

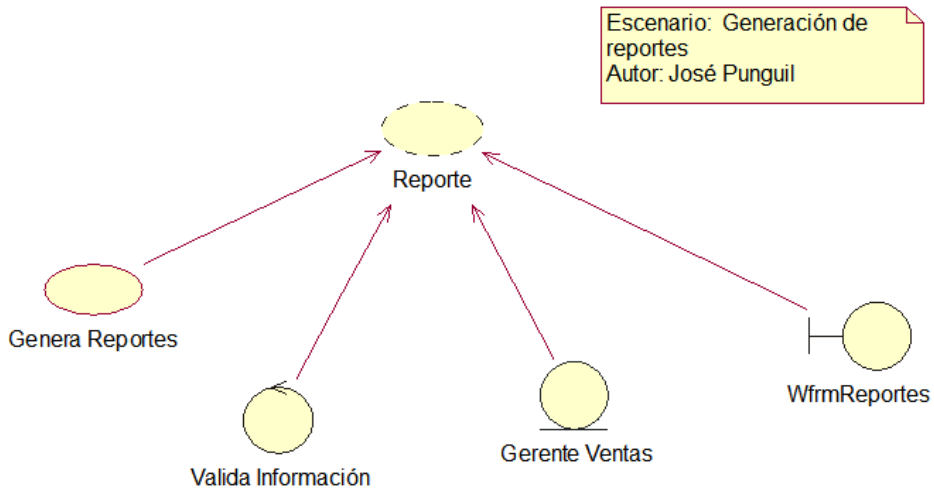


Figura 14. *Diagrama generar reporte*

Este diagrama muestra el proceso para generar reportes administrativos que ayudara a la toma de decisiones y se podrá verificar la eficiencia del sistema implementado ya que en base a toda la información ingresada anteriormente se podrá obtener reportes de diferente tipo como: ventas realizadas por sucursales, ventas realizadas por vendedores, reporte de cumplimiento del telemercadeo ingresado por el gerente, reportes por periodos de fechas, reporte de comisiones, reportes de clientes, etc.

Esto a su vez mostrara la utilidad y el control de los procesos que anteriormente se lo hacía manualmente o sin control alguno.

5.02.10. Diagrama de realización registro nuevo usuario

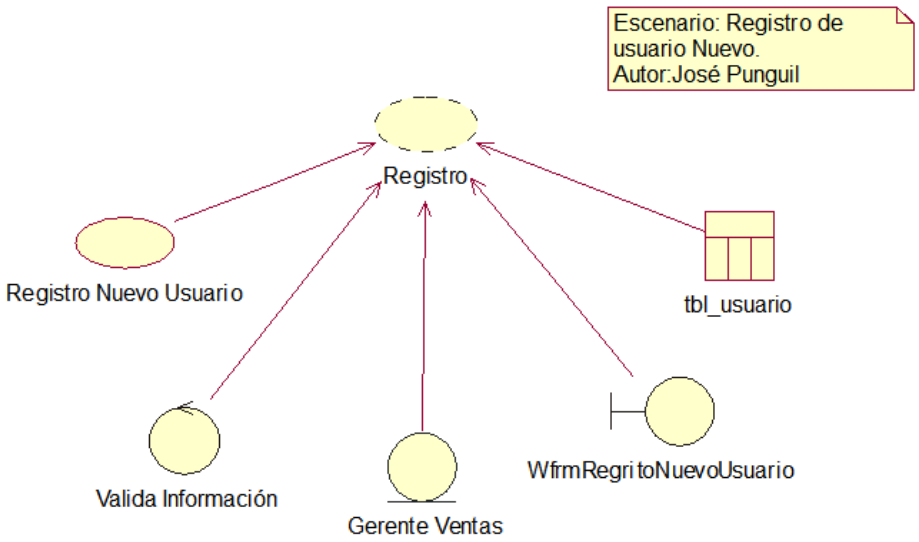


Figura 15. *Diagrama registro de nuevo usuario*

En el presente diagrama interviene un actor el cual es el Gerente de ventas donde el objetivo es registrar un nuevo usuario. Para esto el gerente de ventas deberá estar logeado en el sistema, e ingresar en el formulario de registro de usuarios donde el software validara la información y guardara si todo es correcto de lo contrario nos mostrara un mensaje de error al guardar la información.

5.02.11. Diagrama de secuencia Cliente

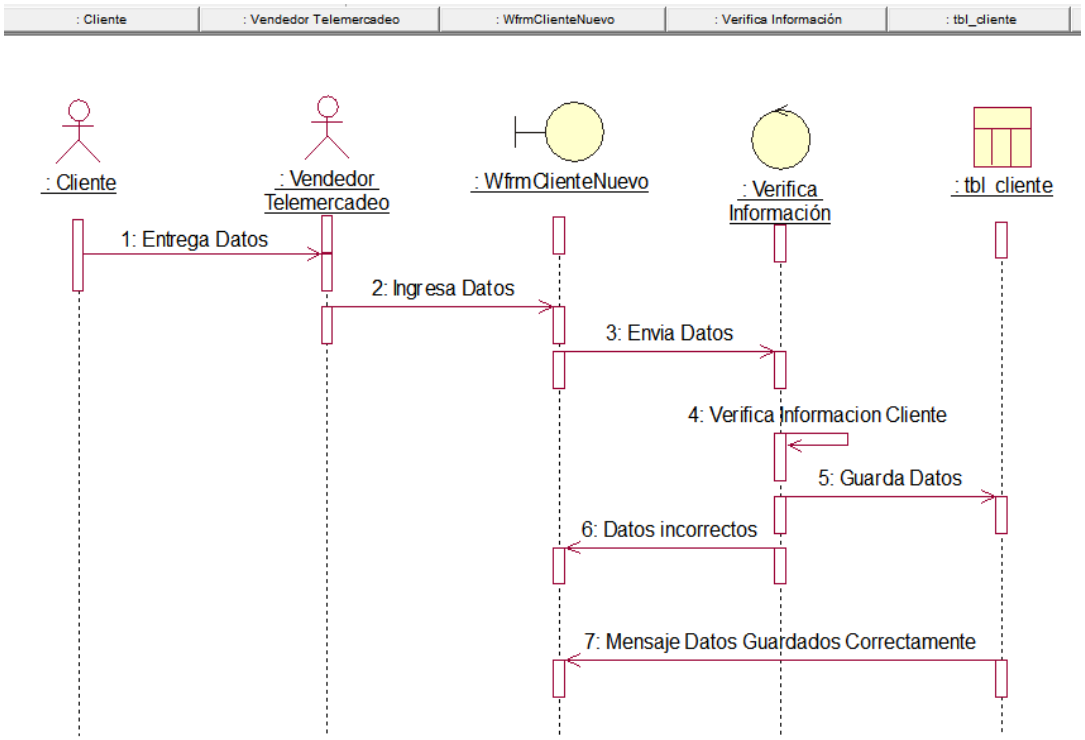


Figura 16. *Diagrama de secuencia registro cliente*

En este diagrama de secuencia intervienen dos actores los cuales son el cliente y el vendedor de la compañía en que va a estar logeado en la interfaz cliente nuevo, lo primero a realizar es que el cliente entrega sus datos al vendedor, luego dicha información es digitada en la interfaz donde el software validara la información y guardara si todo es correcto de lo contrario muestra un mensaje de error.

5.02.12. Diagrama de secuencia Productos

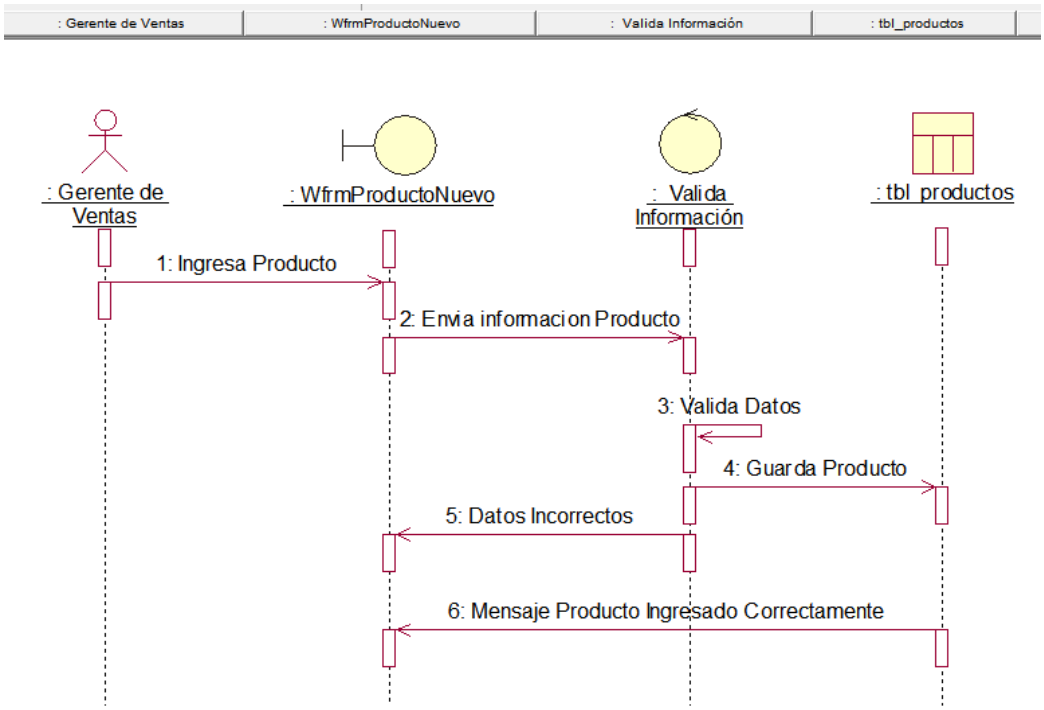


Figura 17. *Diagrama de Secuencia registro productos*

En este diagrama de secuencia intervienen el actor el cual es el gerente de ventas de la compañía el que va a estar logeado en la interfaz cliente nuevo, lo primero a realizar es digitar la información, luego dicha información es validara en el software y guardara la información si todo es correcto de lo contrario muestra un mensaje de error.

5.02.13. Diagrama de secuencia Información Cliente

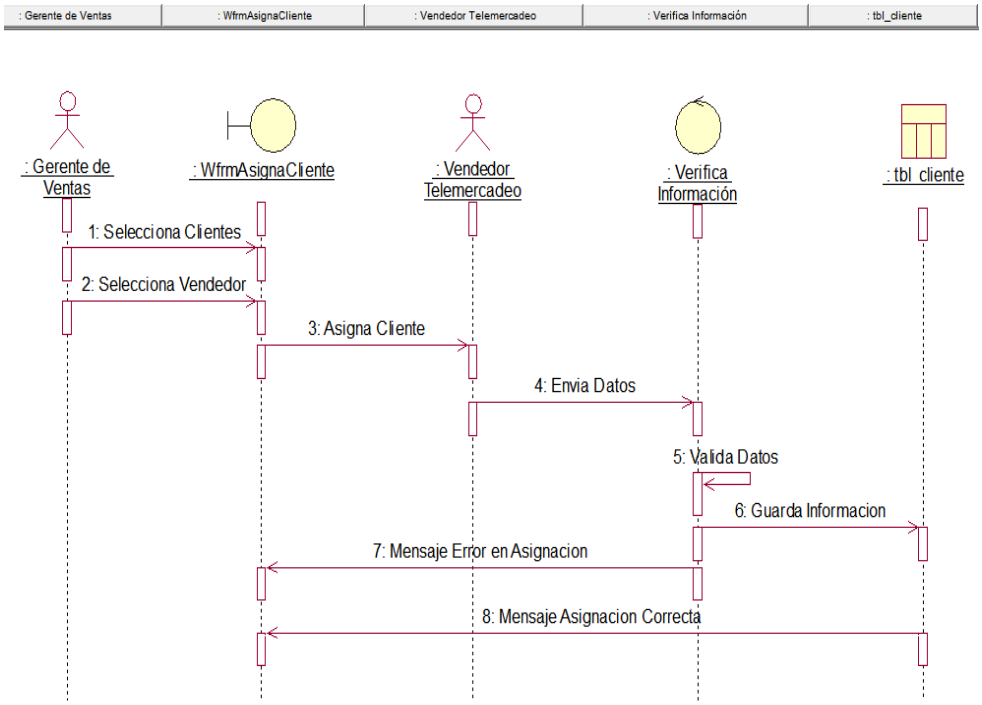


Figura 18. *Diagrama de secuencia asigna información cliente*

En este diagrama de secuencia se muestra el proceso a seguir para asignar un grupo de clientes a un vendedor, con el propósito de controlar el telemercadeo, para esto el gerente seleccionara los clientes y escogerá a un vendedor para asignar la información con el envio de datos y validación respectiva del software, si toda la información es correcta el registro se guarda de lo contrario mostrara un mensaje de error.

5.02.14. Diagrama de secuencia Telemercadeo

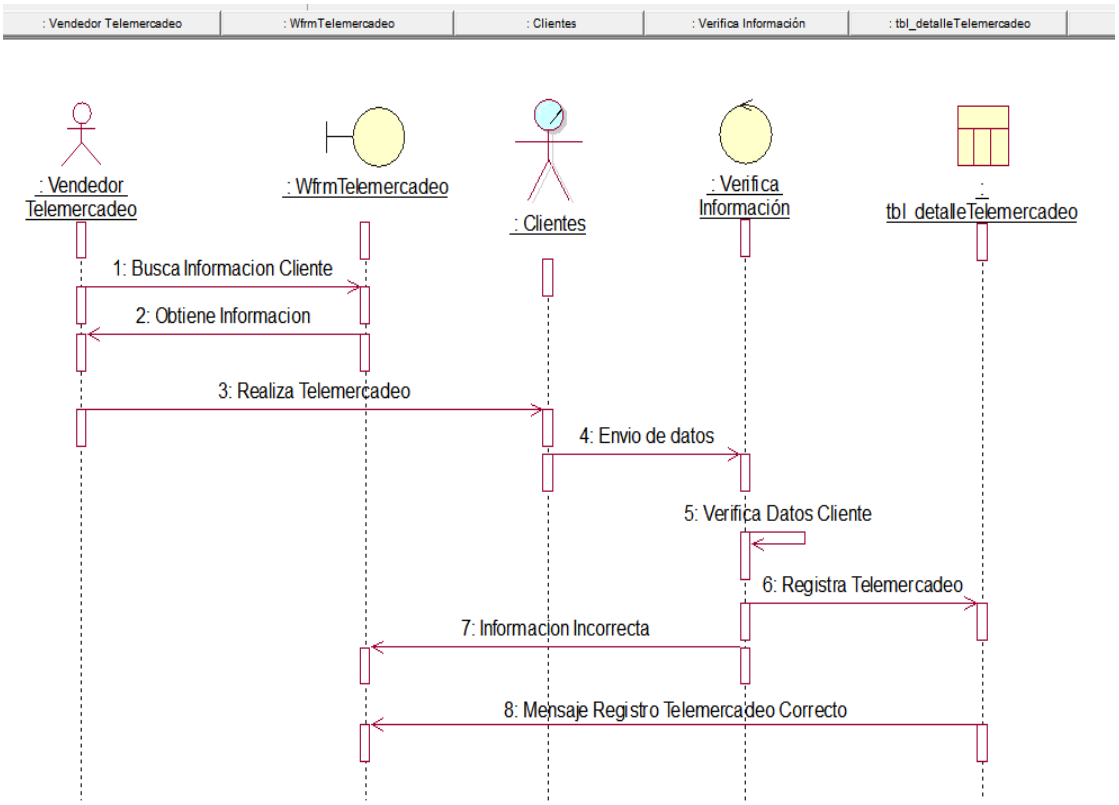


Figura 19. *Diagrama de secuencia realiza telemercadeo*

En este diagrama el vendedor busca la informacion del cliente, una ver obtenida realiza el telemercadeo es decir se comunica con el cliente, para verificar si la informacion del cliente es correcta, luego ofrece los productos que la empresa comercializa para tratar de obtener un damostracion, esa informacion se almacenara en la base para su posterior uso.

5.02.15. Diagrama de secuencia Proformas

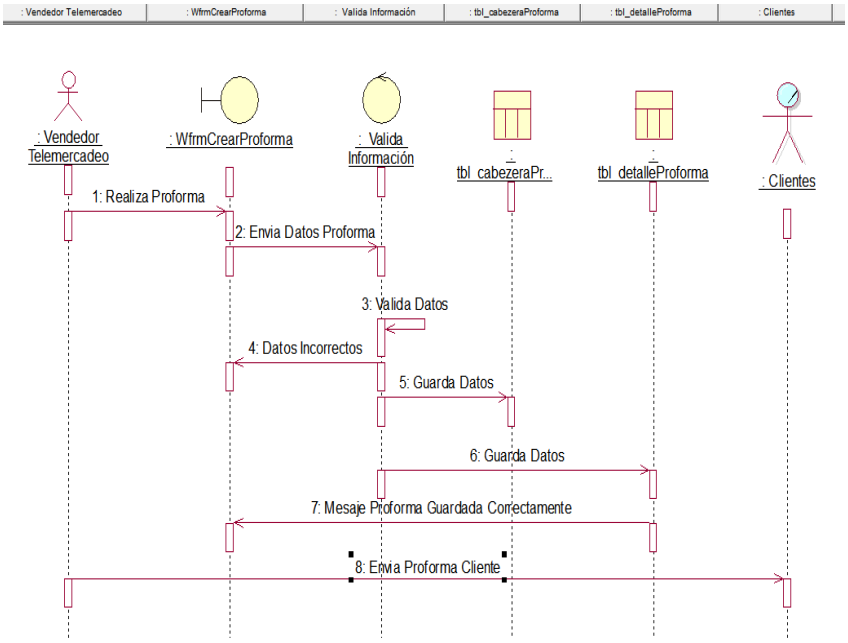


Figura 20. *Diagrama de Secuencia proforma*

En el presente diagrama de secuencia se explica el proceso a seguir para la generación de una proforma, el cual es, ingresar al sistema al módulo de proformas donde el vendedor realizara la búsqueda del cliente asi como la búsqueda del producto, una vez con la información llena en el formulario el vendedor envía los datos para que el software valide dicha información, si toda la información ingresada es correcta guardara en la base de datos y mostrara un mensaje de datos guardados de lo contrario envía un error.

5.02.16. Diagrama de secuencia Facturar

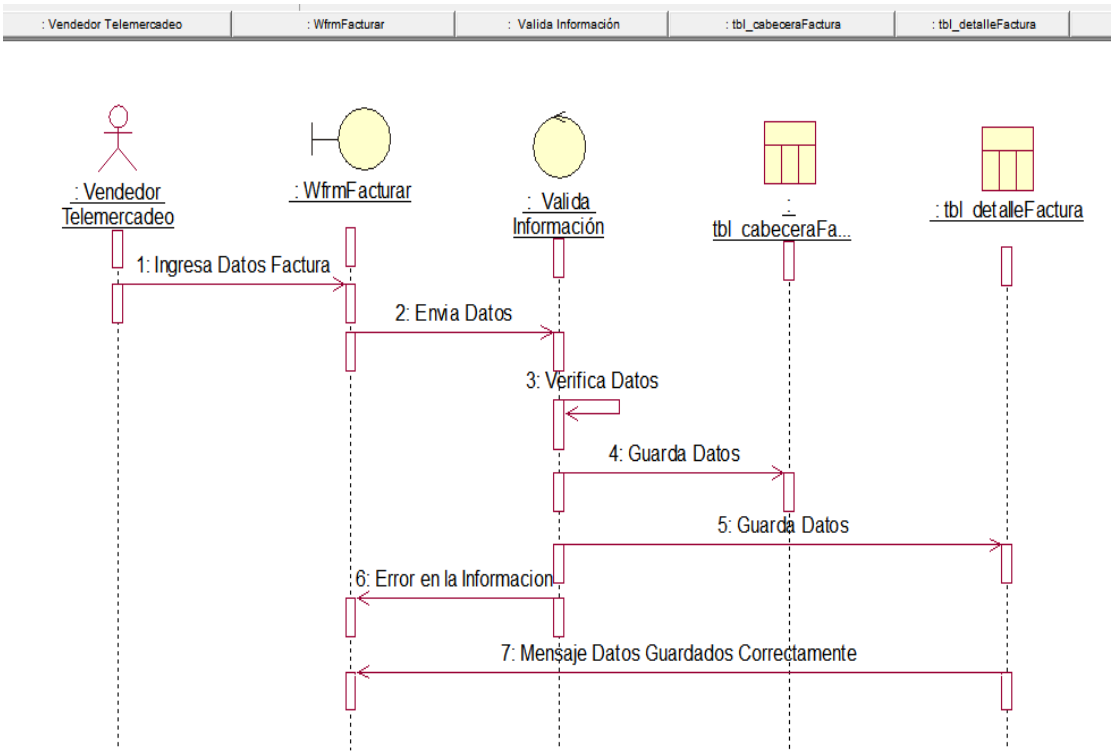


Figura 21. *Diagrama de Secuencia Facturar*

En el presente diagrama de secuencia se explica el proceso a seguir para la generación de una factura, el cual es, ingresar al sistema al módulo de facturas donde el vendedor realizara la búsqueda del cliente así como la búsqueda del producto, una vez con la información llena en el formulario el vendedor envía los datos para que el software valide dicha información, si toda la información ingresada es correcta guardara en la base de datos y mostrara un mensaje de datos guardados de lo contrario envía un error.

5.02.17. Diagrama de secuencia Ventas

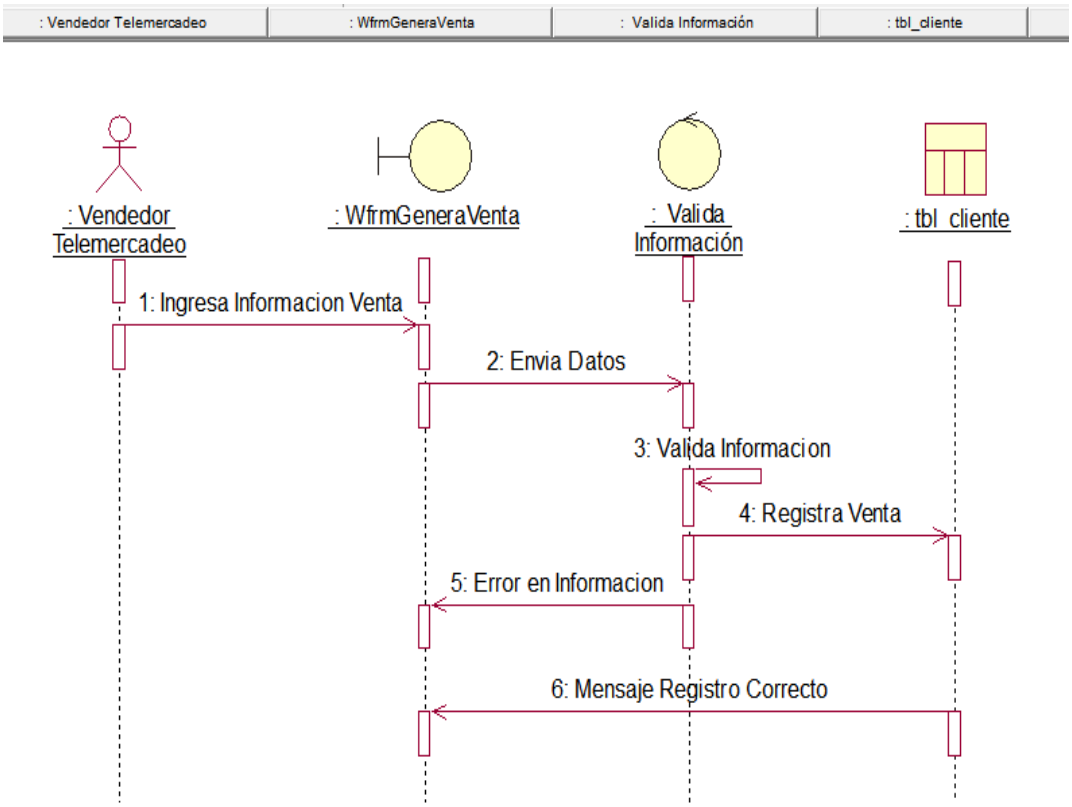


Figura 22. *Diagrama de Secuencia genera ventas*

En este diagrama de secuencia se diagrama el proceso para la generación de una venta, el cual es que el vendedor mediante un seguimiento previo a un telemercadeo realizado pueda generar una venta, es decir que se considera como venta a los clientes a una factura que se generó de una llamada saliente de lo contario si es de una llamada entrante no será considerado como venta.

5.02.18. Diagrama de secuencia Reportes

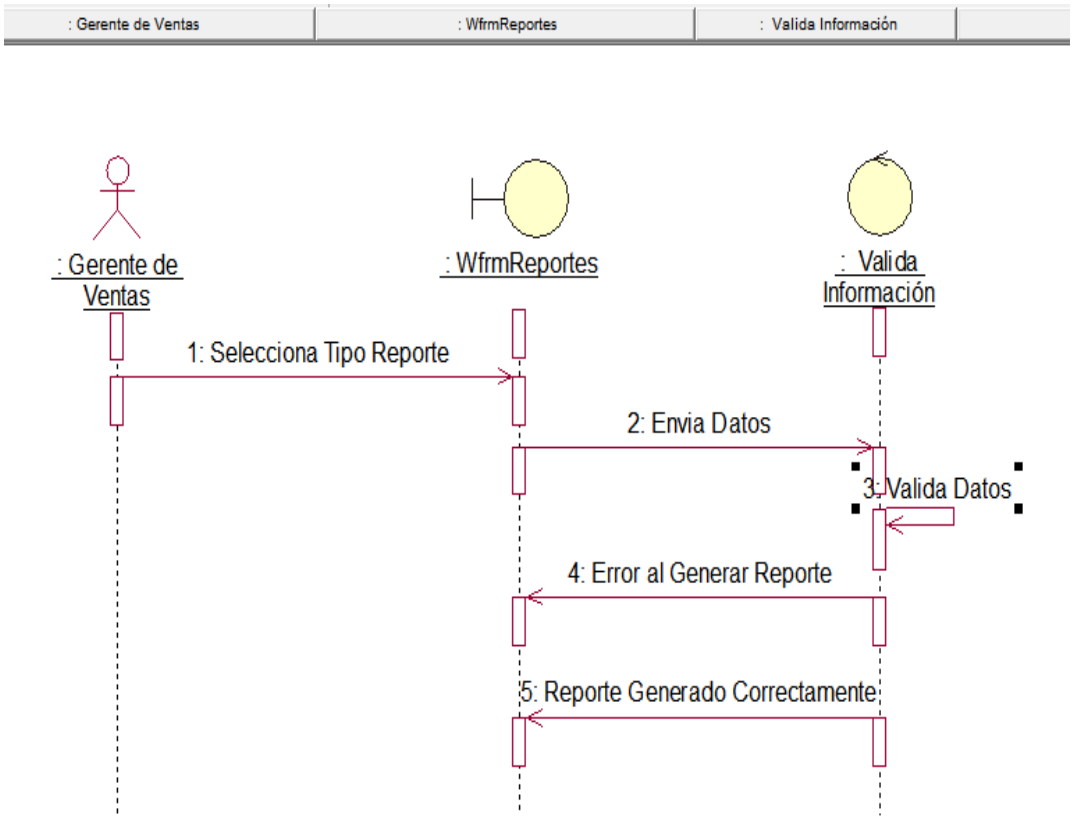


Figura 23. *Diagrama de Secuencia genera reportes*

El diagrama de secuencia muestra el proceso para poder generar reportes del sistema, el cual es: el gerente de ventas ingresa al sistema al formulario de reportes donde el gerente podrá escoger los diferentes tipos de reportes y hasta incluso por periodos de tiempo si toda la información es correcta mostrara los reportes de lo contrario mostrara un mensaje de error identificando la posible causa.

5.02.19. Diagrama de secuencia Usuario

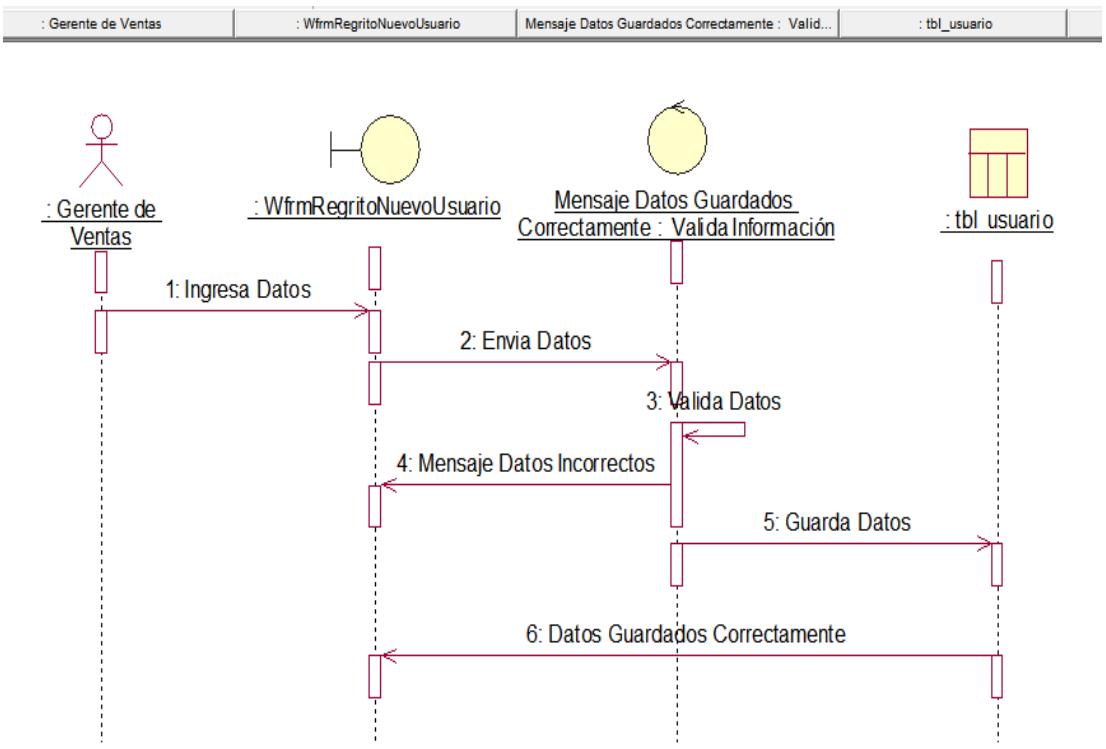


Figura 24. *Diagrama de Secuencia nuevo usuario*

En este diagrama de secuencia intervienen el actor el cual es el gerente de ventas de la compañía el que va a estar logeado en la interfaz cliente nuevo, lo primero a realizar es digitar la información, luego dicha información es validara en el software y guardara la información si todo es correcto de lo contrario muestra un mensaje de error.

5.02.20. Diagrama de colaboración cliente

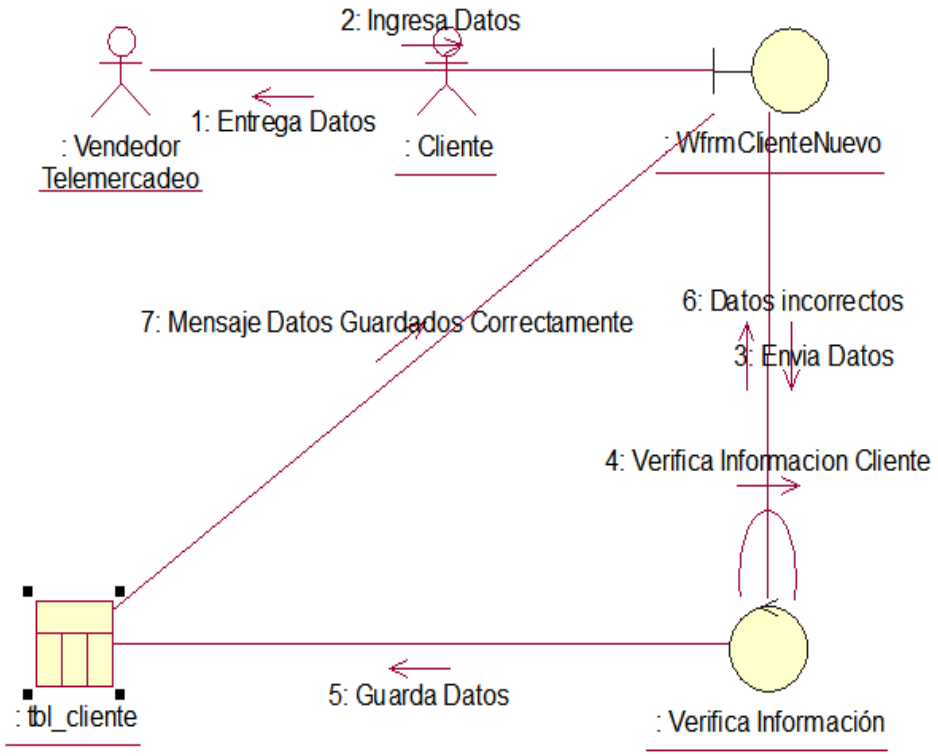


Figura 25. *Diagrama de colaboración registro cliente*

Diagrama que muestra la interacción entre diferentes roles, en este caso muestra la interacción del cliente y el vendedor para realizar el registro de un nuevo cliente.

5.0.2.21. Diagrama de colaboración telemercadeo

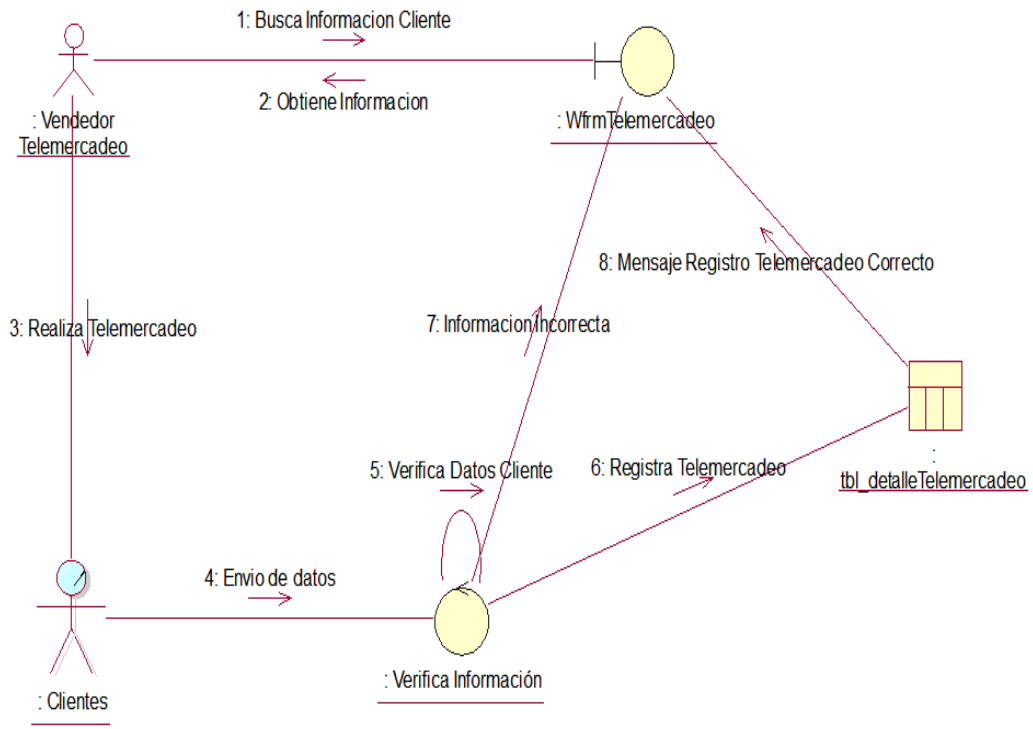


Figura 26. *Diagrama de Colaboración realizar telemercadeo*

Diagrama que muestra la interacción entre diferentes roles, en este caso muestra la interacción del cliente y el vendedor para realizar la gestión de telemercadeo la cual consiste en comunicarse y ofrecer los servicios con los que cuenta la empresa.

5.0.2.22. Diagrama de colaboración Proforma

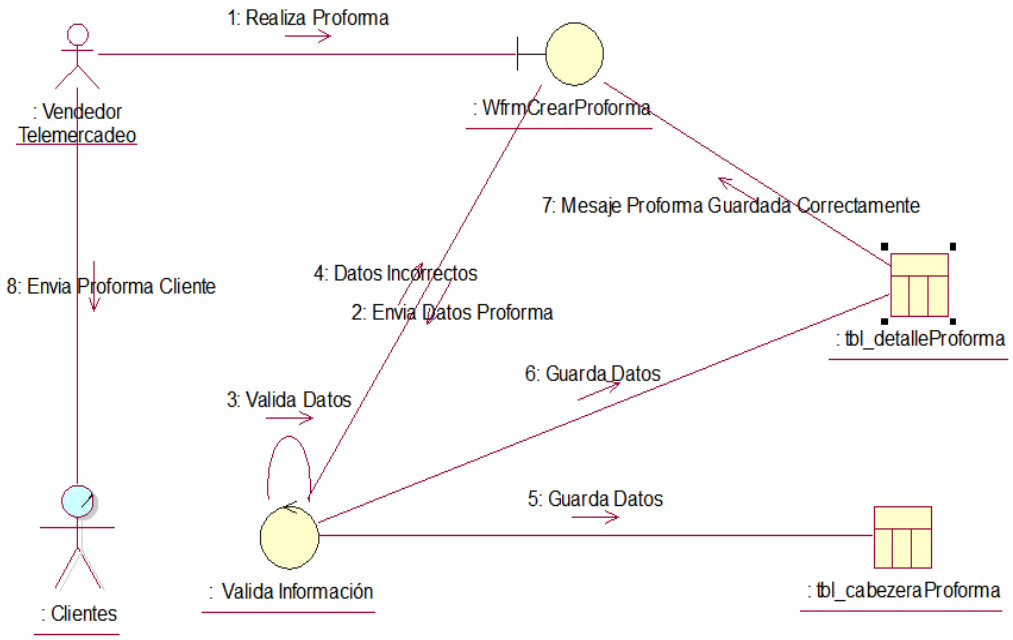


Figura 27. *Diagrama de Colaboración realizar proforma*

Diagrama que muestra la interacción entre diferentes roles, en este caso muestra la interacción del cliente y el vendedor para realizar una nueva proforma, en si muestra la relación que existe entre el cliente y el vendedor.

5.0.2.23. Diagrama de colaboración factura

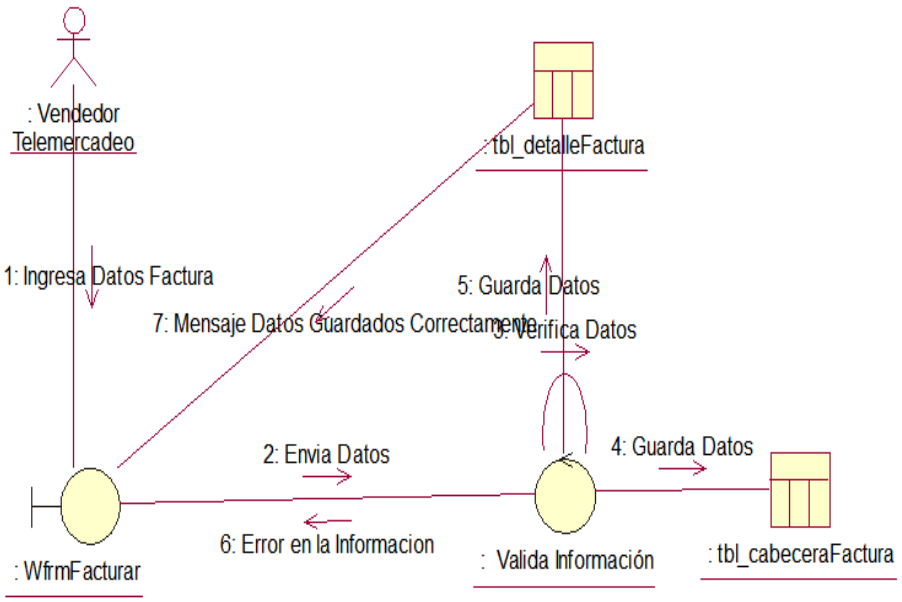


Figura 28. *Diagrama de Colaboración facturar*

Diagrama que muestra la interacción entre el vendedor y la interfaz de facturación en la cual seleccionara los clientes y los productos a facturar.

5.02.24. Diagrama de colaboración Ventas

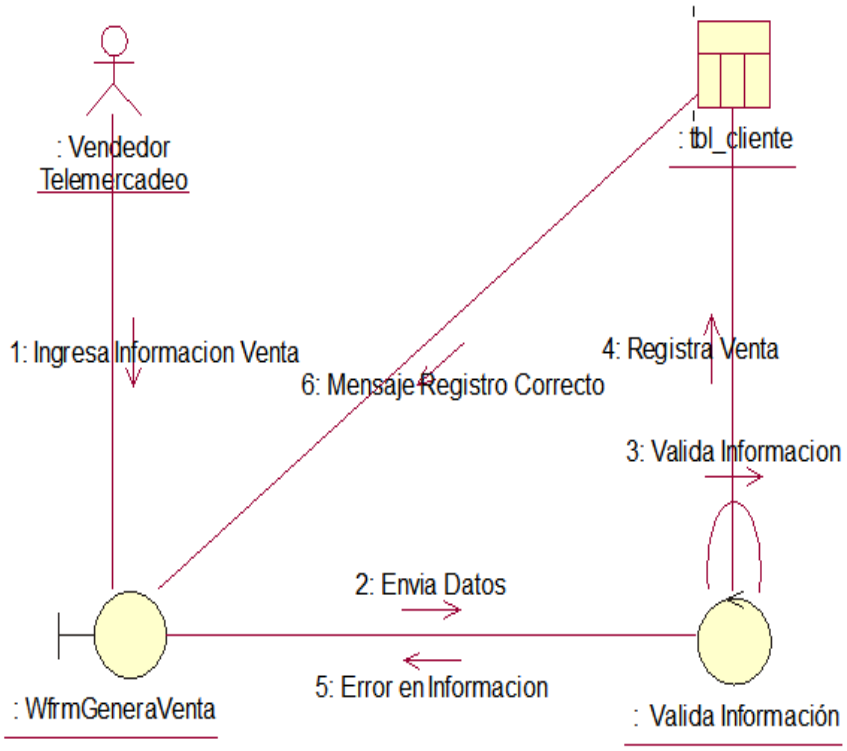


Figura 29. *Diagrama de Colaboración generar ventas*

Este diagrama permite visualizar la interacción con los diferentes elementos con los cuales el vendedor interactúa para poder generar una venta.

5.02.25. Diagrama de colaboración Reportes

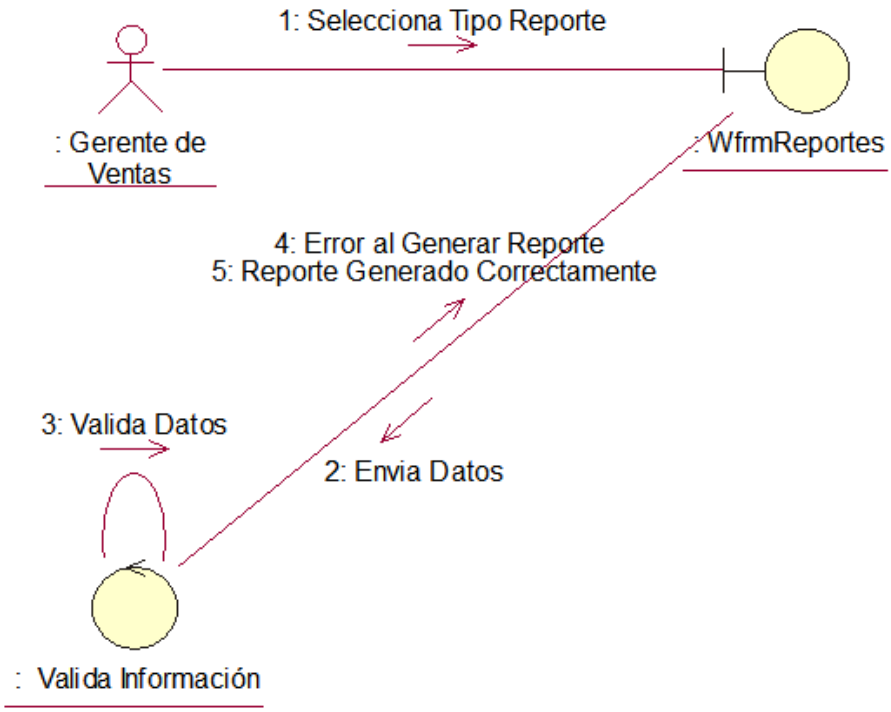


Figura 30. *Diagrama de Colaboración reportes*

En el presente diagrama podemos observar los procesos a seguir y las interacciones que el Gerente debe realizar para poder obtener los diferentes reportes que genera el sistema los cuales ayudaran en el área administrativa y también para la toma de decisiones.

5.02.26. Diagrama de colaboración Usuarios

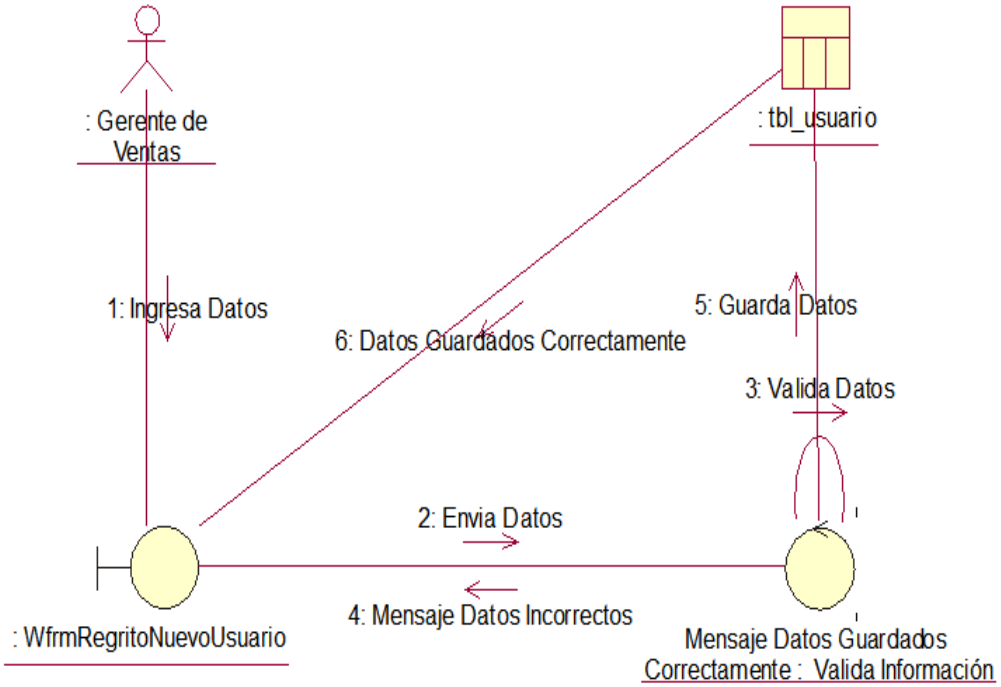


Figura 31. *Diagrama de Colaboración registro de usuarios*

En este diagrama muestra detalladamente todos los procesos e interacciones que realizara el gerente para poder ingresar un nuevo usuario al sistema.

5.02.27. Diagrama de componentes

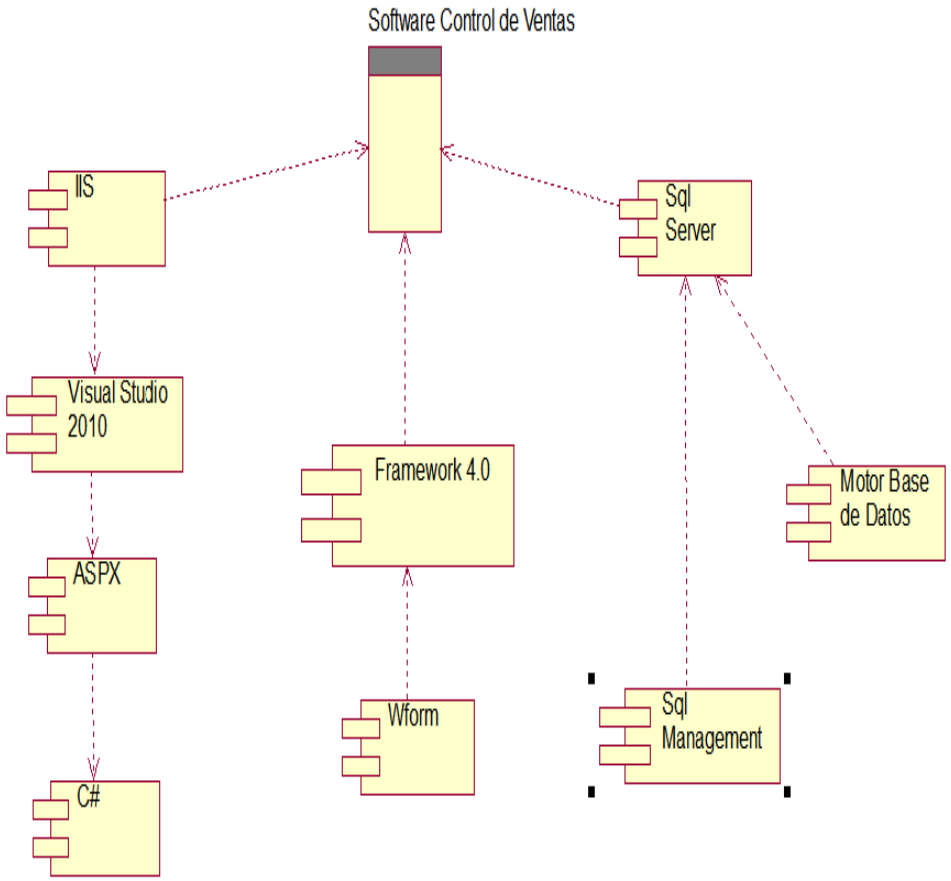


Figura 32. *Diagrama de componentes*

Muestra todos los componentes de software y desarrollo utilizados para la creación e implementación del sistema control de procesos de ventas.

5.02.28. Modelo lógico

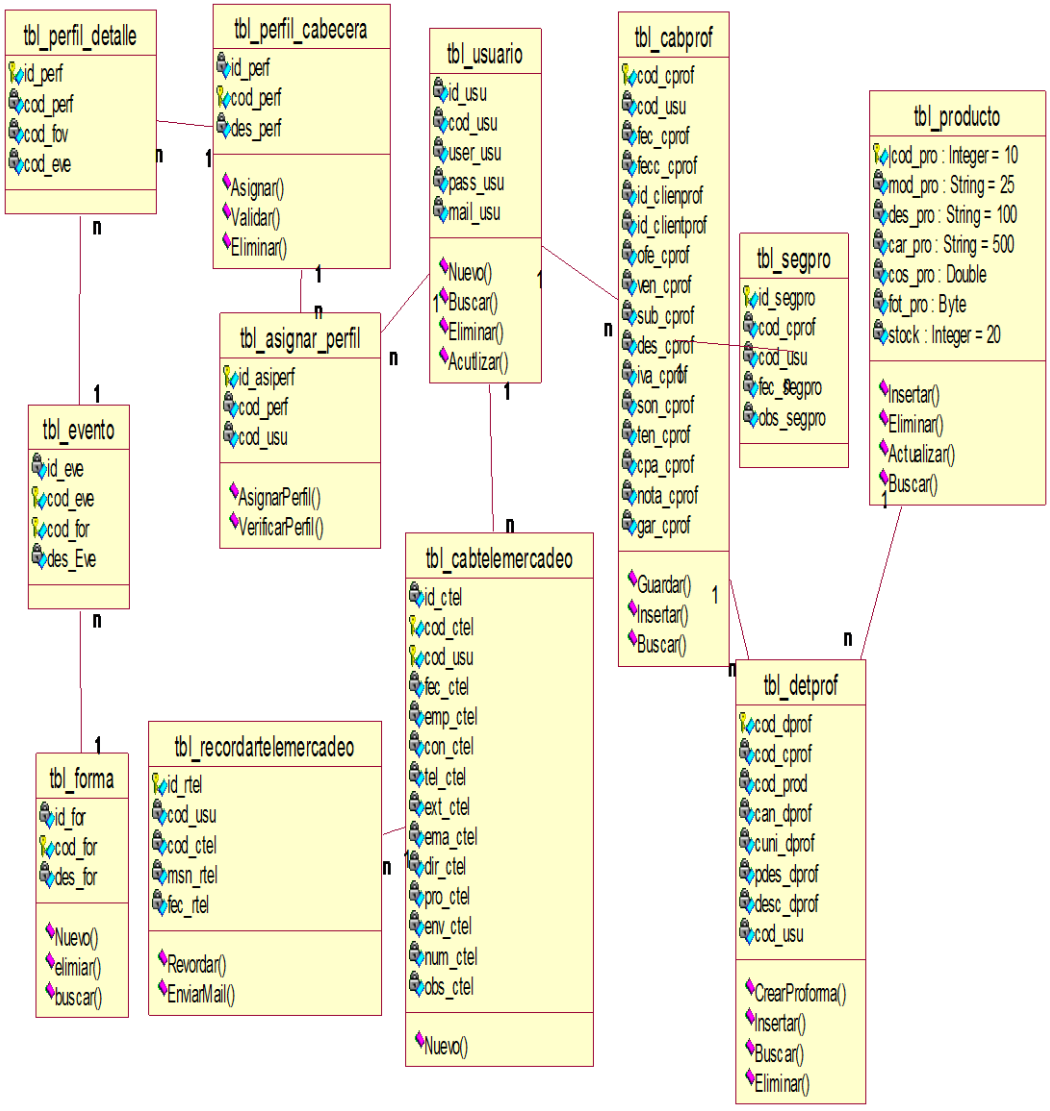


Figura 33. *Modelo lógico*

El presente modelo lógico presenta modelado los elementos estructurales estáticos que se usa a demás captura y define los objetos, entidades y bloques de construcción de un sistema.

5.02.29. Modelo físico

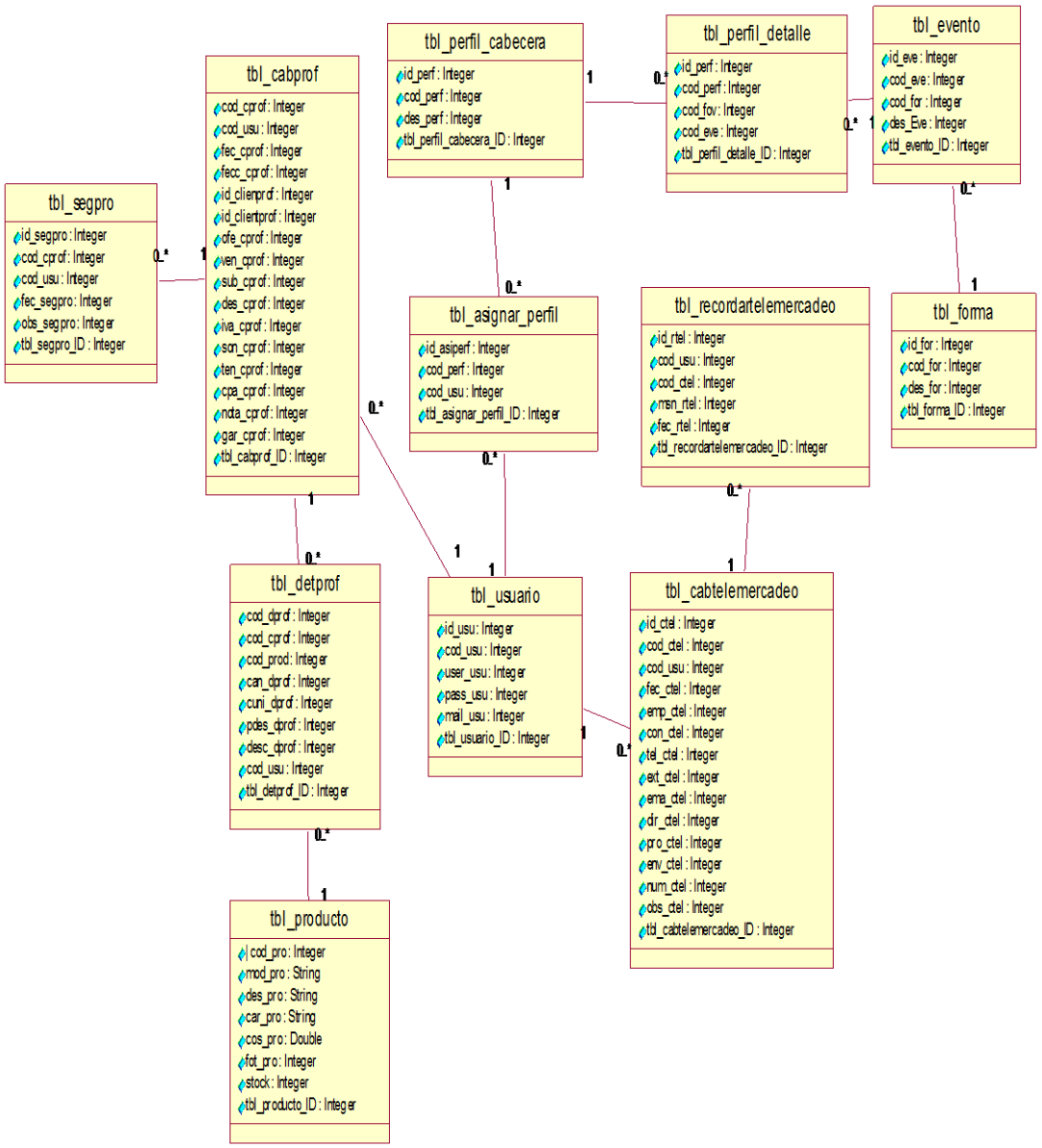


Figura 34. *Modelo físico*

El modelo físico es el cual describe los componentes, de hardware y de software, que se
Utilizaran durante el desarrollo de software. Describe elementos tales como plataformas de hardware.

5.03. Desarrollo

5.03.01. Arquitectura de desarrollo

Para el desarrollo de esta aplicación se utilizó una arquitectura de 3 capas, la cual tiene por objetivo separar las capas de negocio, capa de datos y capa de presentación.

5.03.01.01. Capa presentación

Esta capa es la que ve el usuario, presenta el sistema o interfaz de usuario, le comunica la información y captura la información del usuario en un mínimo de proceso.

Esta capa también se comunica con la capa de negocio. También es conocida como interfaz gráfica y debe tener la característica de ser amigable para el usuario.

5.03.01.02. Capa lógica de negocios:

En la capa lógica de negocio, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se la conoce coma capa de negocio e incluso lógica del negocio, porque es aquí donde se establecen las reglas que deben cumplirse.

Esta capa es la que se comunica son la capa presentación, para almacenar o recuperar datos del él.

5.03.01.03. Capa acceso a datos

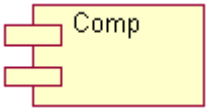
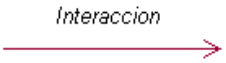
Provee el acceso a los repositorios de información persistente y/o sistemas externos. La tecnología utilizada para desarrollar esta capa es ado.net, y se utilizaron stored procedure.

5.04. Estándares de diseño UML

Para este proyecto se utilizó los siguientes estándares de diseño uml.

Tabla 7.

Descripción objetos UML

Nombre	Diagrama	Descripción
Clase		Describe un conjunto de objetos que comparten los mismos atributos, métodos, relaciones y semántica.
Caso de uso		Describe un conjunto de secuencias de acciones que un sistema ejecuta.
Componente		Parte física y por tanto reemplazable de un modelo, que agrupa un conjunto de interfaces, archivos de código fuente, clases, colaboraciones y proporciona la implementación de dichos elementos.
Interacción		Comprende un conjunto de mensajes que se intercambian entre un conjunto de objetos, para cumplir un objetivo específico.
Nota		Partes explicativa de UML, que puede describir textualmente cualquier aspecto del modelo

5.05. Estándares de programación

5.05.01. Nombres para módulos de formulario y de clase

Para el nombre de los formularios utilizados en el proyecto se utilizó el siguiente estándar Wfrmclientes donde todos los formularios iniciaran con la letra "w" seguido de frm y el nombre relacionado a la interfaz.

Para el caso de las clases utilizadas se definió de la siguiente forma logicaclientes, por añadidura, tal como lo ve, la denominación incluye el uso de altas y bajas (mayúsculas y minúsculas) para identificar las diferentes palabras que conforman el nombre ya que es más fácil leer wfrmventas, que: frmventasentotal o frmventasentotal (internamente), externa, con respecto al desarrollo de clases para los proyectos estas deben de ser creadas bajo el prefijo de "lógica", quedando de esta forma el nombre "logicanegocio".

5.05.02. Nombres para los controles

Se utiliza el nombre del resultado o dato que se desplegará en el control. Es decir, si al leer alguna información de una base de datos ésta se desplegará en algún cuadro de texto (ejemplo, en la suposición de que el dato sea un apellido paterno), el cuadro se podría

llamar apellidopaterno, con el prefijo "txt", quedando de esta forma "txtapellidopaterno".

En la utilización de webusercontrol o control de usuario web para configurar y crear controles programables y reutilizables se utilizará el nombre con el prefijo "wuc", quedando en forma "wuccontrolacceso".

Para el nombre de los controles se ha utilizado, los siguientes sufijos como se muestra en el siguiente tabla:

Tabla 8.

Descripción nombres de controles.

Control	Sufijo	Grupo de controles	Ejemplo
Textbox	Txt	Estándar	Txtbuscar
Label	Lbl	Estándar	Lblbuscar
Button	Btn	Estándar	Btnbuscar
Dropdownlist	Ddl	Estándar	Ddlcategoria
Checkbox	Chk	Estándar	Chkpreferencia
Checkboxlist	Cbl	Estándar	Cblpreferencia
Radiobutton	Rbt	Estándar	Rbtsexo
Radiobuttonlist	Rbl	Estándar	Rblsexo
Calendar	Cal	Estándar	Calfechainicio
Fileupload	Fup	Estándar	Fuparchivo
Image	Img	Estándar	Imgfoto
Linkbutton	Lbt	Estándar	Lbtsiguiente
Imagebutton	Ibt	Estándar	Ibtdetalle
Gridview	Gdv	Datos	Gdvlista
Requiredfieldvalidator	Rfv	Validación	Rfvnombre
Rangevalidator	Rav	Validación	Ravedad
Regularexpressionvalidator	Rev	Validación	Revdato
Comparevalidator	Cov	Validación	Covemail
Customvalidator	Cuv	Validación	Cuvdato
Scriptmanager	Scm	Extensiones ajax	Scmpagina
Calendarextender	Cae	Extensiones ajax	Caefechainicio

5.05.03. Estándares de objetos .net

Para todos los tipos de objetos utilizados en la programación se procedió a utilizar los siguientes estándares:

Tabla 9.

Descripción objetos ADO

Clase	Prefijo
Dataset	Ds
Datatable	Dt
Datarow	Drw
Connection	Cnn
Command	Cmd
Dataadapter	Da
Datareader	Dr

5.05.04. Estándares y prefijos de variables

Para la utilización de las variables en el proyecto se utilizó los siguientes estándares para poder identificar de manera rápida y sencilla el tipo de variable que se está utilizando:

Tabla 10.

Descripción estándares de variables

Alcance	Prefijo	Ejemplo
Global	G	gNombreUsuario
Nivel de módulo	M	mProgresoCálculo
Público	P	pCantidadUsuario

5.05.05. Estándares para la asignación de variables con tipo de dato.

Uno de los puntos más importantes para la utilización de variables con tipos de dato es que no deben contener espacios en blanco, a su vez se utilizó los prefijos seguidos de una letra en mayúscula de acuerdo a la siguiente tabla:

Tabla 11.

Tipos de datos

Tipo de dato	Prefijo
Boolean	Bln
Byte	Byt
Double	Dbl
Decimal	Dec
Date	Dtm
Float/single	Sng
Integer	Int
Long	Lng
Object	Obj
String	Str

Para el manejo y utilización de las variables de tipo contador y para un reconocimiento rápido y sencillo se utilizó las siguientes:

- X, y, cont.

Existen algunas variables especiales que siempre deben nombrarse de la misma manera, de acuerdo a la siguiente tabla:

Tabla 12.

Descripción variables especiales.

Nombre	Clase	Descripción
Ex	Exception	Manejador de errores de c#.net
Sex	Sqlexception	Manejador de errores de sql
Itm	Listitem	Item de un control dropdownlist
Conexión	Sqlconnection	Conexión a base de datos
Dataadapter	Sqldataadapter	Adaptador de datos
Dataset	Dataset	Contenedor de datos
Datatable	Datatable	Tabla de datos
Comando	Sqlcommand	Comando de datos
Con	Conexionbd	Procesa las solicitudes a la base de datos

En cuanto a las clases negocio no deben contener underline (_) ni espacios en blanco, seguido de la letra en mayúscula correspondiente al módulo y el nombre de la clase con la primera letra en mayúscula.

Las clases entidad no deben contener underline (_) ni espacios en blanco.

Los nombres de las clases deben ser sustantivos.

Ejemplo: Logicanegocio.cs

5.05.06. Estándares de base de datos

Los objetos de base de datos deberán estar debidamente documentados con las siguientes propiedades.

Tabla 13.

Descripción tipos de datos SQL Server

Tipo de dato	Tamaño	Descripción
Bit	1 byte	Valores sí y no, y campos que contienen solamente uno de dos valores.
Tinyint	1 byte	Un número entero entre 0 y 255.
Money	8 bytes	Un número entero comprendido entre: –922.337.203.685.477,5808 y 922.337.203.685.477,5807.
Datetime	8 bytes	Una valor de fecha u hora entre los años 100 y 9999
Decimal	17 bytes	Un tipo de datos numérico exacto con valores comprendidos entre 1028 - 1 y -1028 - 1. Puede definir la precisión (1 - 28) y la escala (0 - precisión definida).
Float	8 bytes	Un valor de coma flotante de precisión doble con un intervalo comprendido entre – 1,79769313486232e308 y – 4,94065645841247e-324
Integer	4 bytes	Un entero largo entre – 2.147.483.648 y 2.147.483.647.
Image	Lo que se requiera	Desde cero hasta un máximo de 2.14 gigabytes.

Las tablas representan una instancia de una entidad, y debe contener solo información que corresponde a la entidad que representa así pues el nombre de ésta debe ser alusivo a la información que contiene antecediendo el prefijo tbl_, por ejemplo:

Clientes => tbl_clientes.

Para el caso de las tablas que se generan por el rompimiento de una relación de varios a varios se utilizara las 3 primeras latras de la tabla principal seguido de (_) con las tres primeras letras de la segunda tabla, como se muestra en el siguiente ejemplo:

- *tbl_cliente*
- *tbl_producto*

Para pode definir el nombre de las columnas se deberá escribir con sentido común y de acuerdo con la información que se está guardando. Se utilizara las 3 primeras letras del nombre de la columna seguido del (_) más las 3 primeras letras que identifican a la tabla, de esta manera cuando se tenga una consulta compleja, como por ejemplo con un join no habrá conflicto de ambigüedad con nombres.

- Idi_usu
- Nom_usu
- Est_usu

Procedimientos almacenados

Los procedimientos almacenados siempre tiene una tarea específica y en base a estos se los nombrara, utilizando el prefijo sp_ mas el nombre de la tarea con las iniciales en mayúsculas.

Ejemplos:

- sp_InsertarCliente
- sp_ActualizarCliente
- sp_EliminarCliente

Índices y claves foráneas

El nombre del índice debe comenzar con un prefijo que indique el tipo de índice, nombre de la tabla en donde se creara, columna y si lo desea información adicional:

- Pk_idi_usu => clave primaria de la tabla usuario.
- Pk_idi_cli = > clave primaria de la tabla cliente.
- Fk_+tablaquereferencia +tablareferenciada.

5.06. Diseño de interfaces

5.06.01. Inicio de sesión (login.)



Figura 35. *Inicio de sesión (login.)*

En esta pantalla se muestra el inicio de sesión de los usuarios previamente registrados en el sistema, dependiendo del privilegio asignado a cada usuario tendrán acceso a los diferentes opciones que presenta el sistema.

5.06.02. Interfaz principal, usuario administrador



Figura 36. *Pantalla principal, usuario administrador*

En esta pantalla se muestra todas las opciones a las que tiene acceso el usuario de tipo administrador, las cuales son: productos, clientes, proformas, usuarios, facturación, reportes.

5.06.03. Interfaz principal, usuario vendedor



Figura 37. *Pantalla principal, usuario vendedor*

En esta pantalla se muestra todas las opciones a las que tiene acceso el usuario de tipo vendedor, las cuales son: clientes, proformas, facturación.

5.06.04. Pantalla para el mantenimiento de productos.

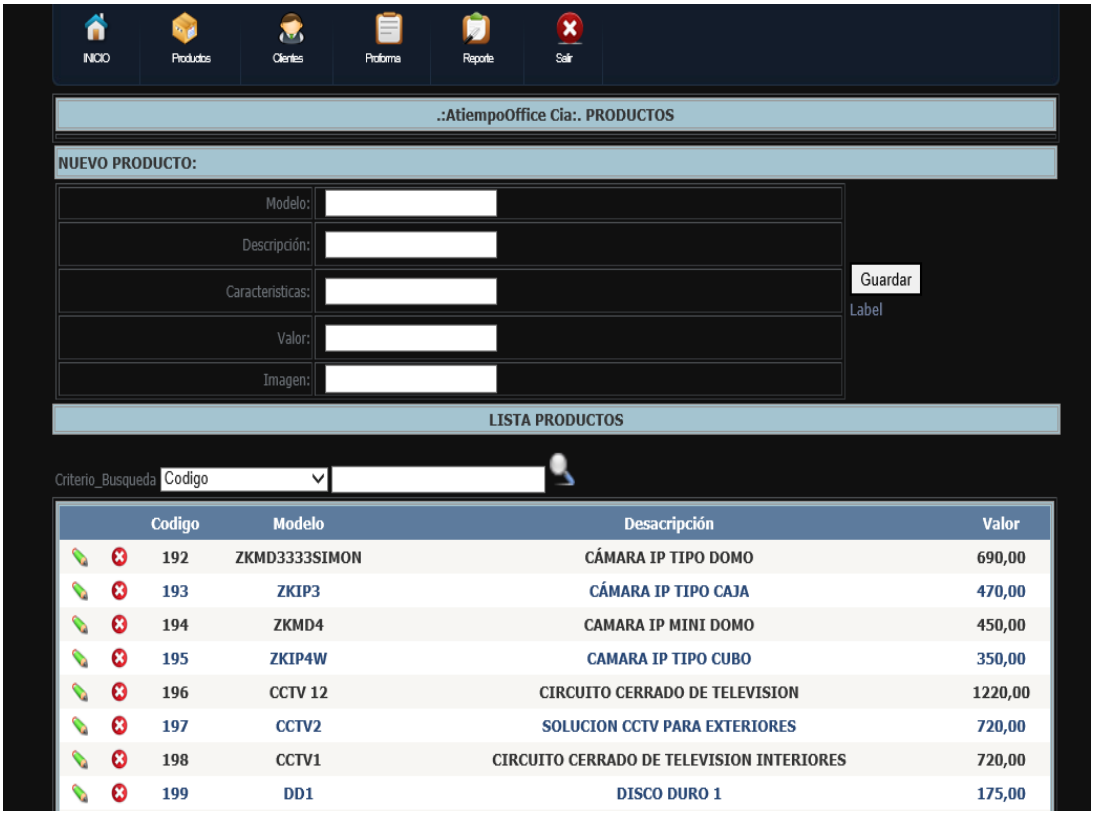


Figura 38. *Pantalla para el mantenimiento de productos.*

En esta interfaz se realiza el ingreso, modificación y eliminación de productos los cuales comercializa la empresa, a este módulo solo tiene acceso el usuario de tipo administrador.

5.06.05. Pantalla para el mantenimiento de clientes.

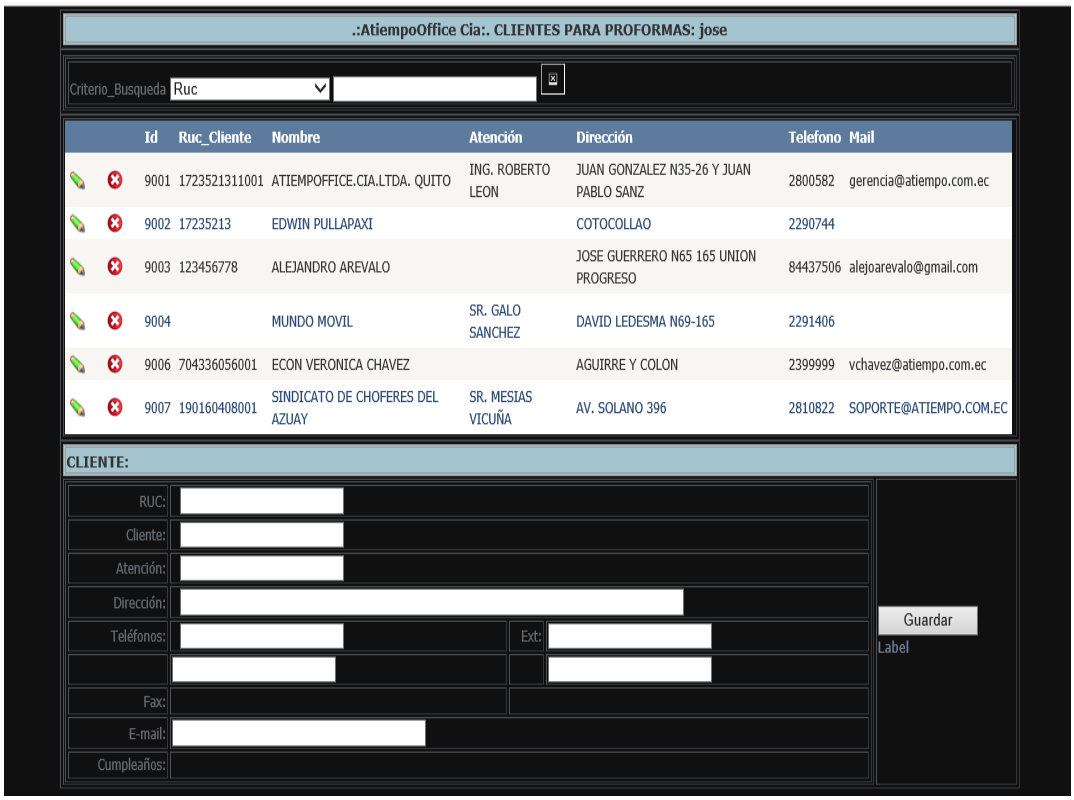


Figura 39. *Pantalla para el mantenimiento de clientes.*

En esta interfaz se realiza el ingreso, modificación y eliminación de clientes, a este módulo tiene acceso el usuario de tipo administrador y el usuario de tipo vendedor.

5.06.06. Interfaz para la creación de proformas.

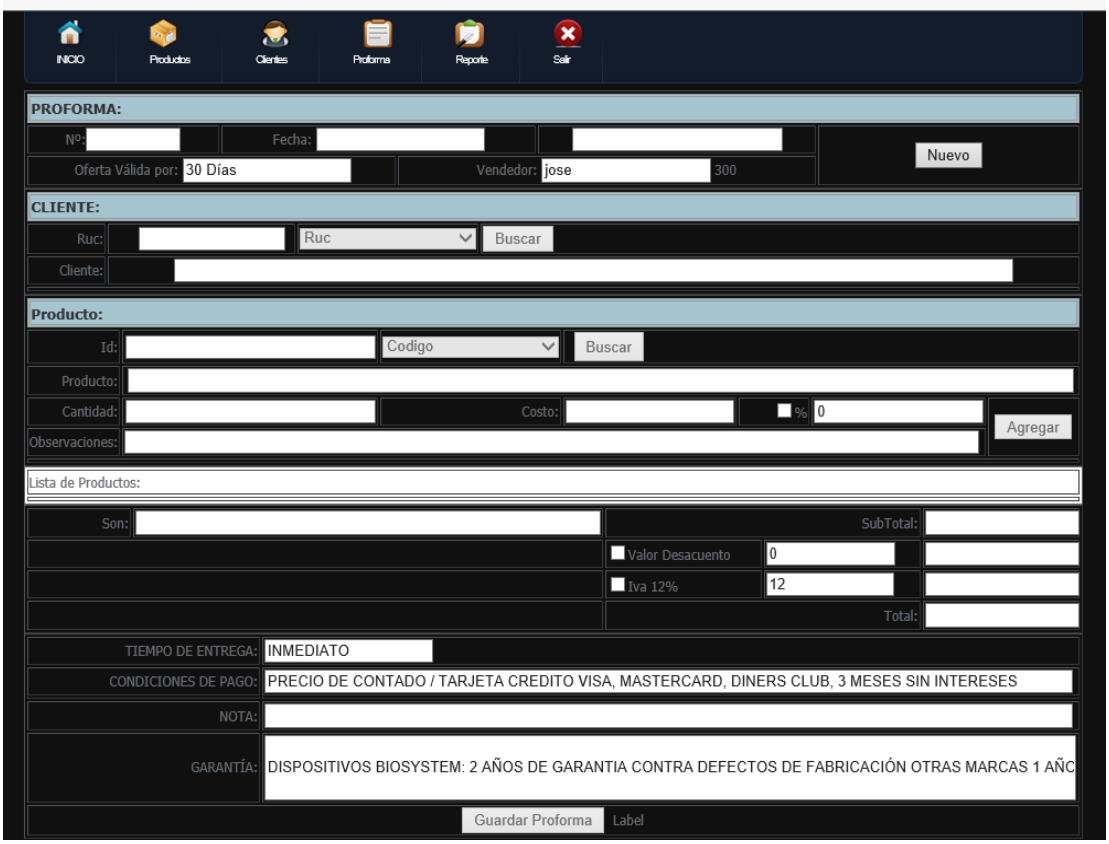


Figura 40. *Interfaz para la creación de proformas*

En esta interfaz los diferentes tipos de usuarios podrán elaborar las proformas de acuerdo a las necesidades del cliente, a su vez el guardado de dicha proforma para luego poder obtener reportes requeridos por la empresa.

5.06.07. Interfaz para la creación de usuarios.

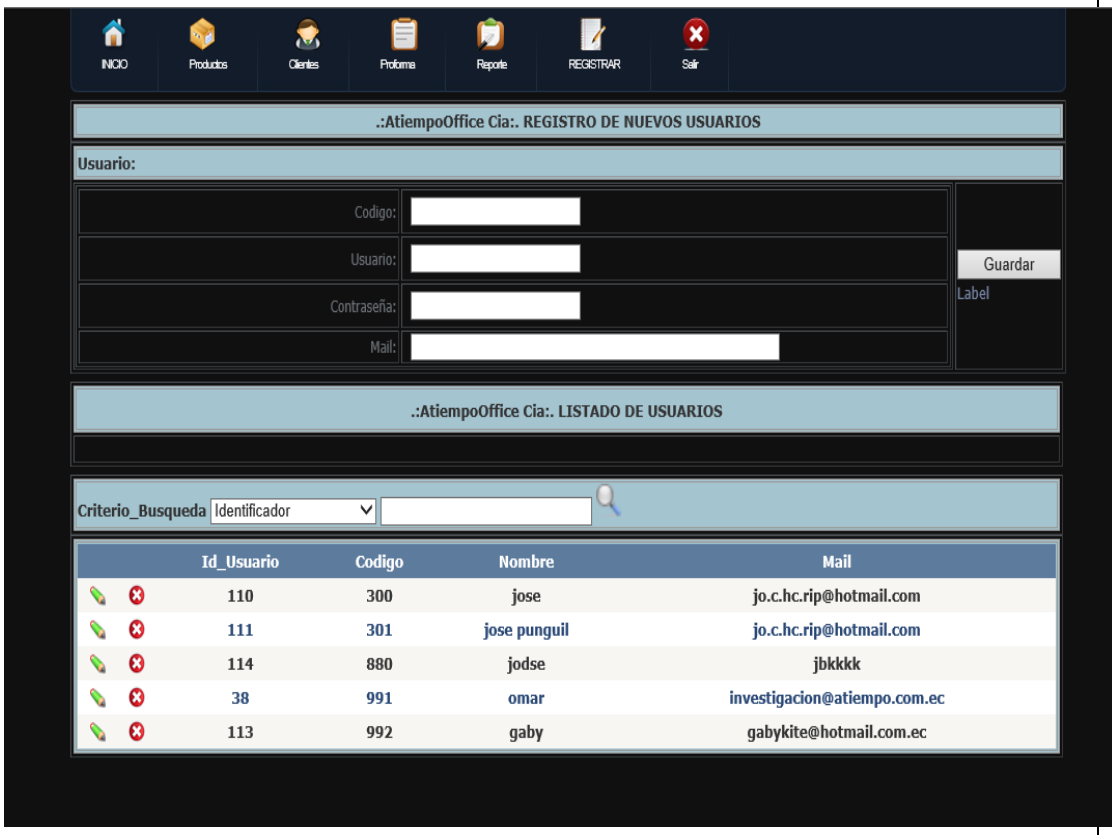


Figura 41. *Interfaz para la creación de usuarios*

En esta interfaz el usuario de tipo administrador podrá realizar modificación y eliminación de los usuarios que se encuentran registrados en el sistema y a su vez el ingreso de nuevos usuarios con los diferentes privilegios.

5.06.08. Interfaz para la creación de facturas.

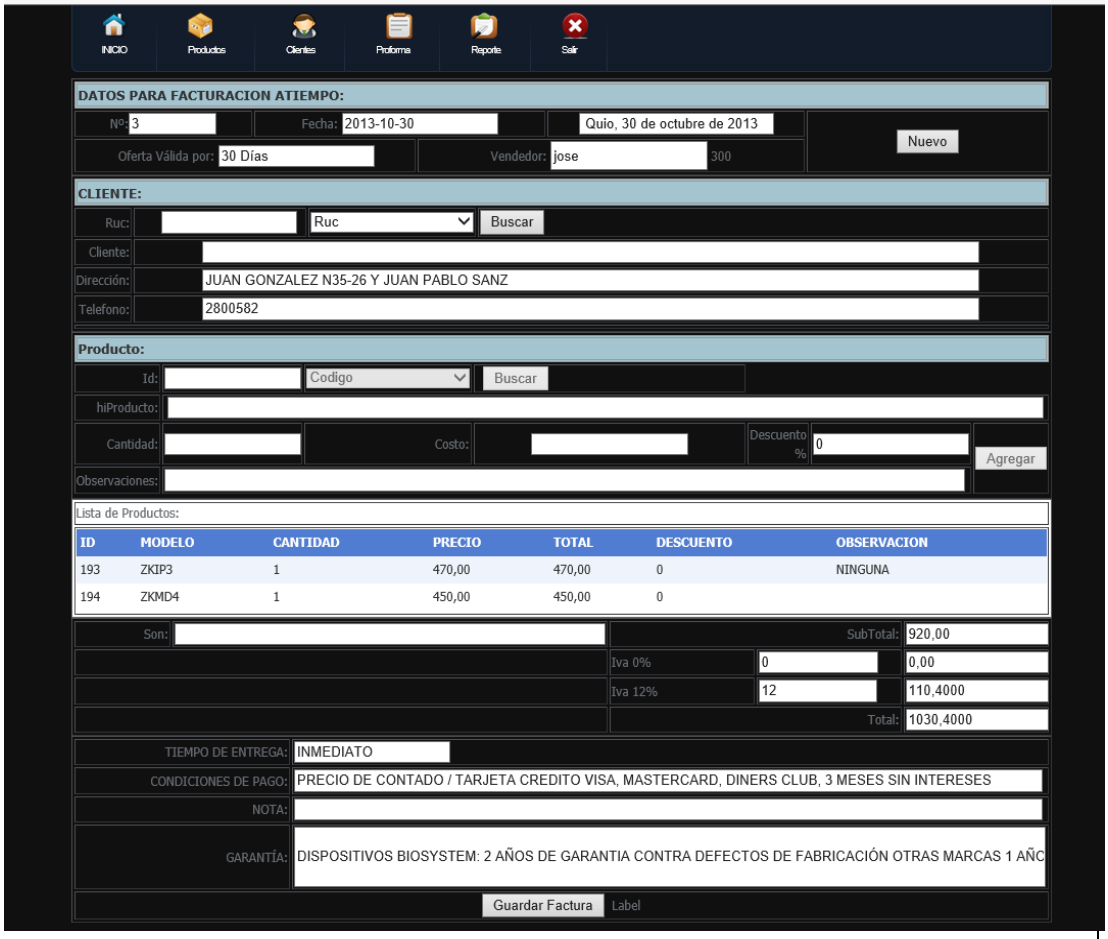


Figura 42. *Interfaz para la creación de facturas*

En esta interfaz los usuarios podrán facturar lo que permitirá emitir reportes de facturación de acuerdo a cada vendedor lo que ayudara a la toma de decisiones en el área administrativa.

5.07. Casos de Pruebas

5.07.01. Técnicas de prueba

El desarrollo de Sistemas implica la realización de una serie de actividades predispuestas a incorporar errores.

Debido a que estos errores se deben a nuestra habilidad innata de provocar los para esto tenemos que incorporar una actividad que garantice la calidad del software.

5.07.02. Fundamentos de la prueba del software

En la etapa de prueba del software se crean una serie de casos de prueba que intentan obtener todos los posibles errores que puedan generarse en el software desarrollado.

5.07.03. Objetivos de la prueba

La prueba es un proceso de ejecución de un programa con la intención de descubrir la mayor cantidad de errores que puedan ocurrir.

Un caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces

Una prueba tiene éxito si descubre un error no detectado hasta entonces.

El objetivo es diseñar casos de prueba que, sistemáticamente, saquen a la luz diferentes clases de errores, haciéndolo con la menor cantidad de tiempo y de esfuerzo.

La prueba no puede asegurar la ausencia de errores; sólo puede demostrar que existen defectos en el software, para ello se determinó los casos de prueba siguientes.

5.07.04. Prueba registro de nuevos clientes

Tabla 14.

Caso de prueba registro de nuevos clientes

Registro de Nuevos Clientes

Caso de Uso	CU_01 Registrar Clientes
Caso de Prueba	CP_01 Registro de Clientes
Actor	Gerente, Vendedor
Pre Condiciones	Gerente o vendedor deben estar logeados en el sistema Tener los privilegios necesarios
Propósito	Registrar nuevos clientes

Escenario 1 CP_01: Verifica Información Correcta

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar clientes	Evento	Muestra formulario clientes
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Datos Guardados"

Escenario 2 CP_02: Verifica Información Correcta

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar clientes	Evento	Muestra formulario clientes
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Datos Incorrectos"

5.07.05. Pruebas registro de nuevos clientes

Tabla 15.

Caso de prueba registro de nuevos productos

Registro de Nuevos Productos

Caso de Uso	CU_02 Registrar Productos
Caso de Prueba	CP_02 Registro de Productos
Actor	Gerente
Pre Condiciones	Gerente debe estar logeado en el sistema Tener los privilegios necesarios
Propósito	Registrar nuevos productos

Escenario 1 CP_02: Ingreso de información correcta

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar productos	Evento	Muestra formulario productos
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Datos Guardados"

Escenario 2 CP_02: Ingreso de Información Incorrecta

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar clientes	Evento	Muestra formulario clientes
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Datos Incorrectos"

5.07.06. Pruebas asignación de Clientes a Vendedor

Tabla 16.

Asignación de clientes a vendedores

Asignación de Clientes a Vendedor				
Caso de Uso	CU_03 Entrega información de clientes a vendedores			
Caso de Prueba	CP_03 Entrega información de clientes a vendedores			
Actor	Gerente, Vendedor			
Pre Condiciones	Gerente o vendedor deben estar logeados en el sistema Tener los privilegios necesarios			
Propósito	Aginar listado de clientes a vendedores			
Escenario 1	CP_03: Valida Asignación			
	Sec.	Actividad	Clase Equivalencia	Resultado Esperado
	1	Ingresar al sistema	Válida	Muestra pantalla principal
	2	Presionar Asignar clientes	Evento	Muestra formulario Asignación
	3	Seleccionar clientes	Lista Cliente	Información completa
	4	Seleccionar vendedor	Lista Vendedor	Información completa
	5	Presionar “Guardar”	Valida	Mensaje "Datos Guardados"
Escenario 2	CP_03: Valida Asignación			
	Sec.	Actividad	Clase Equivalencia	Resultado Esperado
	1	Ingresar al sistema	Válida	Muestra pantalla principal
	2	Presionar Asignar clientes	Evento	Muestra formulario Asignación
	3	Seleccionar clientes	Lista Cliente	Información completa

5.07.07. Pruebas gestión de telemercadeo

Tabla 17.

Caso de prueba gestión de telemercadeo

Gestión de Telemercadeo

Caso de Uso	CU_04 Gestión telemercadeo			
Caso de Prueba	CP_04 Gestión telemercadeo			
Actor	Vendedor			
Pre Condiciones	vendedor debe estar logeado en el sistema Tener los privilegios necesarios Tener listado de clientes Verificar información de clientes			
Propósito	Obtener citas para demostraciones			
Escenario 1	CP_04: Aceptación de demostración			
	Sec.	Actividad	Clase Equivalencia	Resultado Esperado
	1	Ingresar al sistema	Válida	Muestra pantalla principal
	2	Presionar clientes	Evento	Muestra formulario
	3	Seleccionar clientes	Lista clientes	Información completa
	4	Llamar al cliente	Evento	Comunicación con cliente
	5	Ofrecer demostración	Evento	Definir fecha y hora para la demo
Escenario 2	CP_04: Verifica Información Correcta			
	Sec.	Actividad	Clase Equivalencia	Resultado Esperado
	1	Ingresar al sistema	Válida	Muestra pantalla principal
	2	Presionar clientes	Evento	Muestra formulario
	3	Seleccionar clientes	Lista clientes	Información completa
	4	Llamar al cliente	Evento	Volver a llamar
	5	Ofrecer demostración	Evento	Registro en el sistema para posterior comunicación

5.07.08.Pruebas generar proforma

Tabla 18.

Caso de prueba generar proformas

Generar Proformas

Caso de Uso	CU_05 Generar proforma
Caso de Prueba	CP_05 Generar proforma
Actor	Vendedor
Pre Condiciones	Vendedor debe estar logeado en el sistema Tener los privilegios necesarios Tener conocimiento de productos
Propósito	Generar nueva proforma

Escenario 1 CP_05: Información ingresada correctamente

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar proformas	Evento	Muestra formulario proforma
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Datos Guardados Correctamente"

Escenario 2 CP_05: Información ingresada incorrectamente

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar clientes	Evento	Muestra formulario proforma
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Datos Incorrectos"

5.07.09. Pruebas generar facturas

Tabla 19.

Caso de prueba generar factura

Generar Facturas

Caso de Uso	CU_06 Generar facturas
Caso de Prueba	CP_06 Generar facturas
Actor	Vendedor
Pre Condiciones	vendedor debe estar logeado en el sistema Tener los privilegios necesarios Conocimiento de productos
Propósito	Generar nueva factura

Escenario 1 CP_01: Información ingresada correctamente

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar facturas	Evento	Muestra formulario facturas
3	Ingresar información	Válida	Información completa
4	Guardar información	Válida	Mensaje "Factura Guardada"

Escenario 2 CP_02: Información ingresada incorrectamente

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar clientes	Evento	Muestra formulario facturas
3	Ingresar información	Válida	Información incompleta
4	Guardar información	Válida	Mensaje "Datos Incorrectos"

5.07.10. Pruebas registro nuevos usuarios

Tabla 20.

Caso de prueba registrar nuevos usuarios

Registro de Nuevos Usuarios

Caso de Uso	CU_08 Registrar Usuarios			
Caso de Prueba	CP_08 Registro de Usuarios			
Actor	Gerente			
Pre Condiciones	Gerente debe estar logeado en el sistema Tener los privilegios necesarios			
Propósito	Registrar usuarios			
Escenario 1	CP_01: Ingreso de información correcta			
	Sec.	Actividad	Clase Equivalencia	Resultado Esperado
	1	Ingresar al sistema	Válida	Muestra pantalla principal
	2	Presionar Usuarios	Evento	Muestra formulario Usuarios
	3	Ingresar información	Válida	Información completa
	4	Guardar información	Válida	Mensaje "Usuario Guardado"
Escenario 2	CP_02: Ingreso de información incorrecta			
	Sec.	Actividad	Clase Equivalencia	Resultado Esperado
	1	Ingresar al sistema	Válida	Muestra pantalla principal
	2	Presionar usuarios	Evento	Muestra formulario usuarios
	3	Ingresar información	Válida	Información completa
	4	Guardar información	Válida	Mensaje "Datos Incorrectos"

5.07.11. Pruebas generar reportes

Tabla 21.

Caso de prueba generar reportes

Generar Reportes

Caso de Uso	CU_09 Reportes
Caso de Prueba	CP_09 Generar reportes
Actor	Gerente
Pre Condiciones	Gerente debe estar logeado en el sistema Tener los privilegios necesarios
Propósito	Obtener reportes para fines administrativos

Escenario 1 — CP_01: Reporte generado correctamente

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar Reportes	Evento	Muestra formulario reportes
3	Seleccionar tipo de reporte	Válida	Listado de reportes
4	Generar reporte	Válida	Mostrar reporte.

Escenario 2 — CP_02: Reporte incorrecto

Sec.	Actividad	Clase Equivalencia	Resultado Esperado
1	Ingresar al sistema	Válida	Muestra pantalla principal
2	Presionar Reportes	Evento	Muestra formulario reportes
3	Seleccionar tipo de reporte	Válida	Listado de reportes
4	Generar reporte	Válida	Mensaje "Error al generar el reporte"

CAPÍTULO VI

6.01. Recursos

Materiales

- Computador HP CORE i7
- Servidor HP CORE 2 DUO

Humano

- Profesores del ITSCO
- Personal involucrado de la empresa
- Clientes

6.02. Presupuesto

Para el desarrollo del proyecto se utilizaron recursos económicos los cuales se detallan en la siguiente tabla a continuación:

Tabla 22.

Descripción detalle presupuesto

DESCRIPCIÓN	CANT	V.UNITARIO	V.TOTAL
Tutoría	1	200	200.00
Seminario	1	520	520.00
Papel	5	6	30.00
Impresiones	180	0.05	9.00
Movilización	50	0,25	12.50
TOTAL			771.50

6.03. Cronograma de actividades

Ver Anexo 3.

CAPÍTULO VII

7.01. Conclusiones

Se realizó un diagnóstico de la problemática de la empresa dando como resultado en el presente trabajo se tuvo la oportunidad de estudiar varios casos, conceptos y técnicas relacionadas con el desarrollo del sistema po lo que se concluye lo siguiente

El sistema desarrollado es de gran ayuda para la empresa en especial en el departamento de ventas debido a los diferentes procesos que abarcan, la gran importancia que es para la empresa y sobre todo para poder controlar los procesos de telermercadeo y ventas, para realizar el desarrollo del sistema se utilizó la arquitectura 3 capas lo que nos permitió utilizar las características más importantes de dicha arquitectura es la utilización de Polimorfismo, Encapsulamiento, y herencia, haciendo mucho más factible la reutilización de código.

También se concluye que la realización de las pruebas es de suma importancia y sin duda de las principales fases por lo que debe pasar en cualquier proyecto a desarrollar, debido a que en dicha fase podemos cerciorarnos que el desarrollo del proyecto fue correcto y comprobar que las necesidades y los problemas se solventaron satisfactoriamente.

Es muy importante mencionar el tema de las seguridades y los permisos que deben tener los sistemas debido a al tipo de información que va a manejar y almacenar y poder darle al usuario final la confiabilidad de que la información es correcta y confiable para la parte administrativa y la toma de decisiones de acuerdo a los resultados.

7.02. Recomendaciones

Se recomienda a la empresa administrar el sistema de una forma muy responsable y cuidadosa, y que a quien se le otorgue el nivel de administrador debe ser un usuario confiable y además es recomendable que lea el manual de usuario y de administrador de modo que tenga idea clara de los beneficios y funcionabilidades de todo el sistema.

Realizar una capacitación de todo el sistema, ya que el sistema funcionaria de mejor manera si todas las personas involucradas tienen conocimiento del mismo.

El administrador del proyecto deberá crear un plan de respaldo de las bases de datos y se recomienda que los respaldos sean al final de cada día ya que la mayoría de las base de datos se modifican entre el día.

Instituto Tecnológico Superior Cordillera

Anexos

Anexo A 1. Manual Técnico

Anexo A 1.01. Instalación de SQL Server 2008

Iniciamos como en un instalador normal, doble clic en el ejecutable de instalación, si usamos Windows 7 nos saldrá un mensaje si queremos solucionar problemas simplemente seleccionamos ejecutar aplicación.

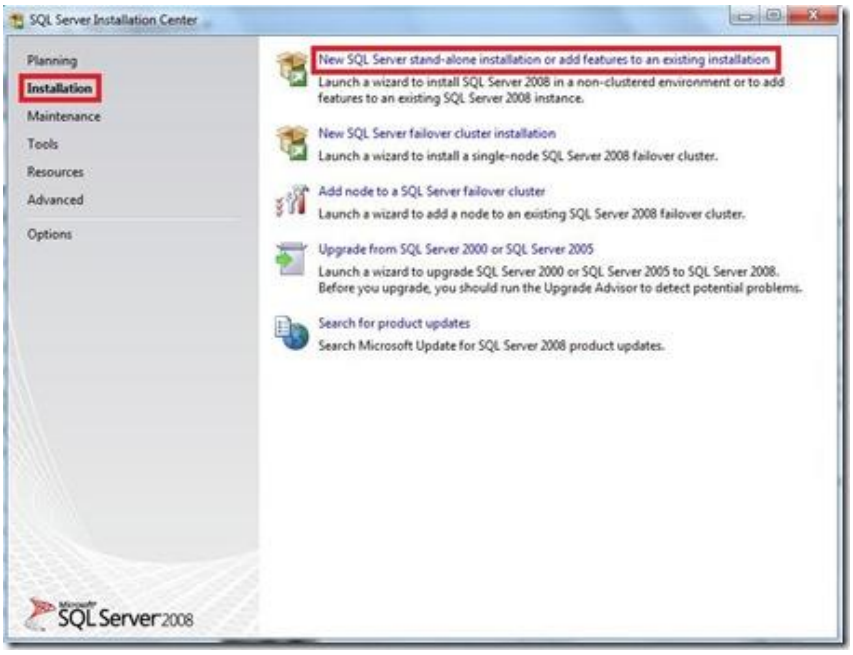


Figura 43 *Instalación de SQL Server*

se nos presenta una pantalla con varias opciones, la que nos interesa es la instalación, por lo que en el menú lateral escogemos “Instalación” y luego Instalar SQL Server ó agregar características en una instalación existente

- ahora bien se nos presenta un modal popup en donde recibimos un mensaje de espera que nos indica que se están realizando las operaciones solicitas

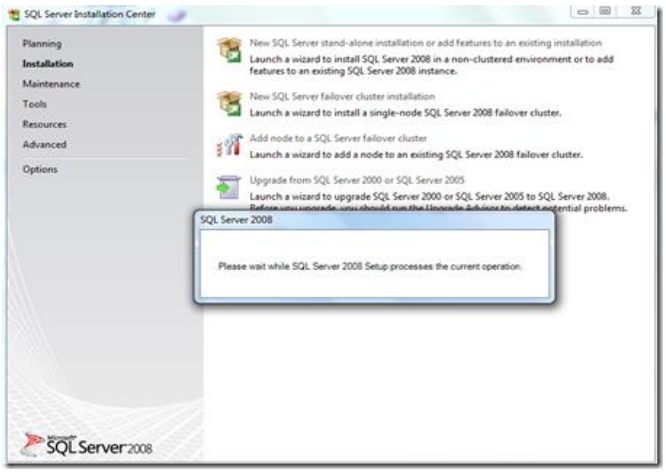


Figura 44 *Nueva Instalación*

Muestra la pantala principal del instalador del SQL server

- Verificar si existen aplicaciones o componentes que eviten la instalación de SQL

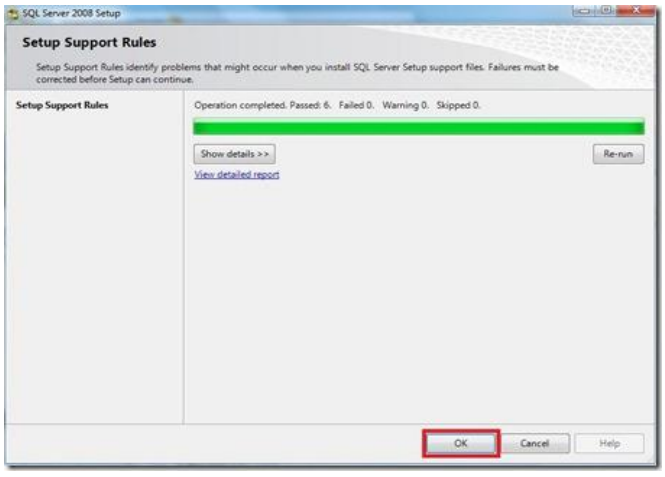


Figura 45 *Comprobando requerimientos.*

Muestra el proceso de recolección de características para la instalación

- En la siguiente pantalla debemos seleccionar “Edición Express”

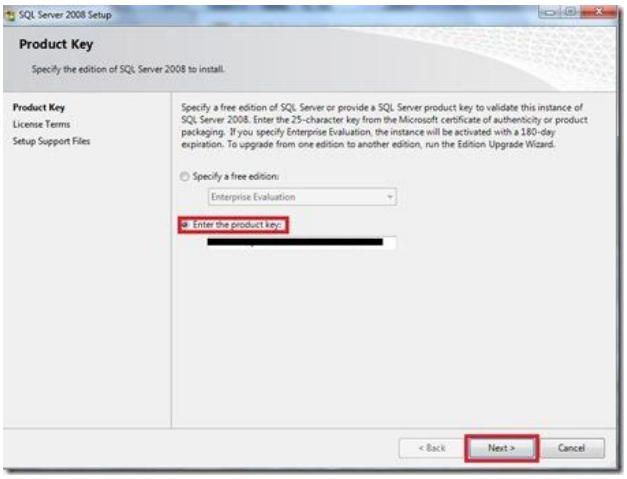


Figura 46 *Seleccionar Producto*

Muestra el tipo de licencia a instalar

- La próxima ventana nos trae los términos de uso de licencia de software, les recomiendo que lo lean y si están de acuerdo clic en "Siguiente"



Figura 47 *Términos de licencia del motor de base de datos.*

Muestra los términos de la licencia

- Instalamos ahora los archivos de soporte que serán nuestra guía durante la configuración de características de SQL Server 2008, simplemente seleccionamos “Instalar”

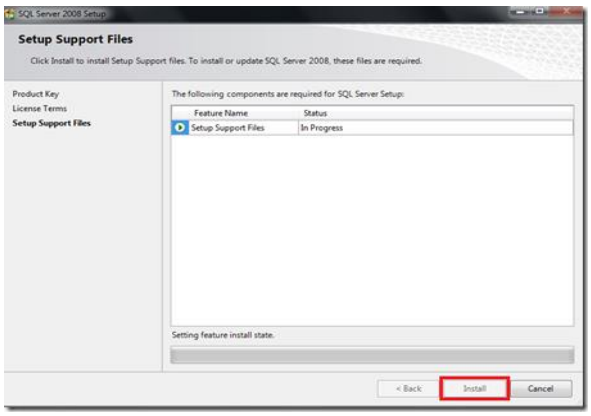


Figura 48 *Recopilando información*

Muestra los requerimientos depara la instalación

- Resumen de las características agregadas y seleccionamos “Siguiente”

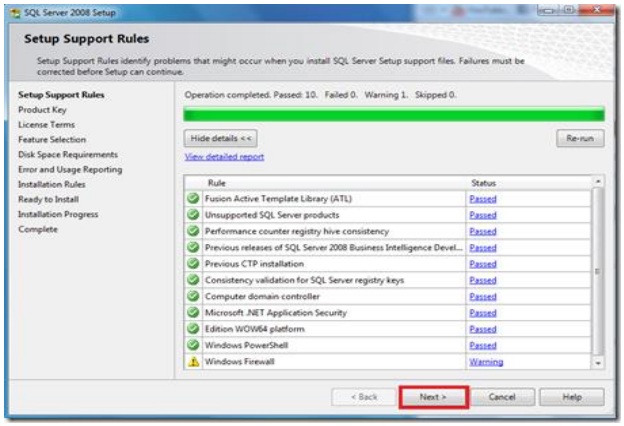


Figura 49 *Características de SQL Server*

Requerimientos para la instalación del SQL Server

- Esta es la parte más importante de la instalación en donde el usuario deberá seleccionar que características desea instalar para su equipo.

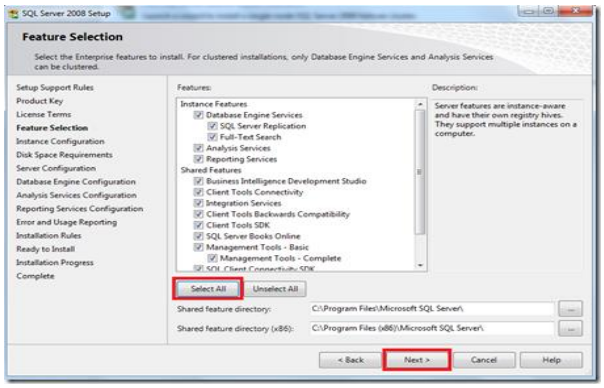


Figura 50 *Aplicaciones*

Muestra resumen para la instalación

- La pantalla a continuación nos muestra la instancia que queramos instalar yo uso la instancia "por defecto" y luego desde el Manager la cambio a mis necesidades, ahora clic en "siguiente"

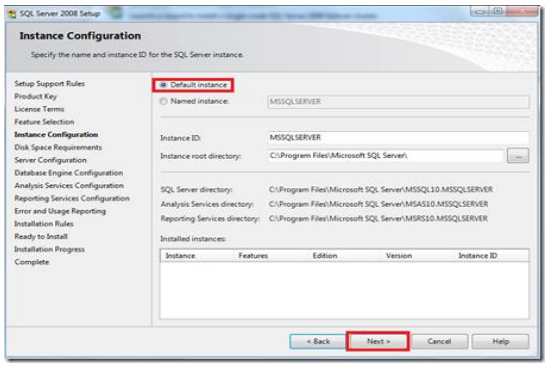


Figura 51 *Instancias de configuración*

Muestra los usuarios para el inicio de sesión en el SQL Server

- Vemos que se nos presenta una pantalla en donde debemos verificar el espacio del disco, que debe ser suficiente para el trabajo del motor de base de datos, si no tenemos problema alguno seleccionamos "siguiente"

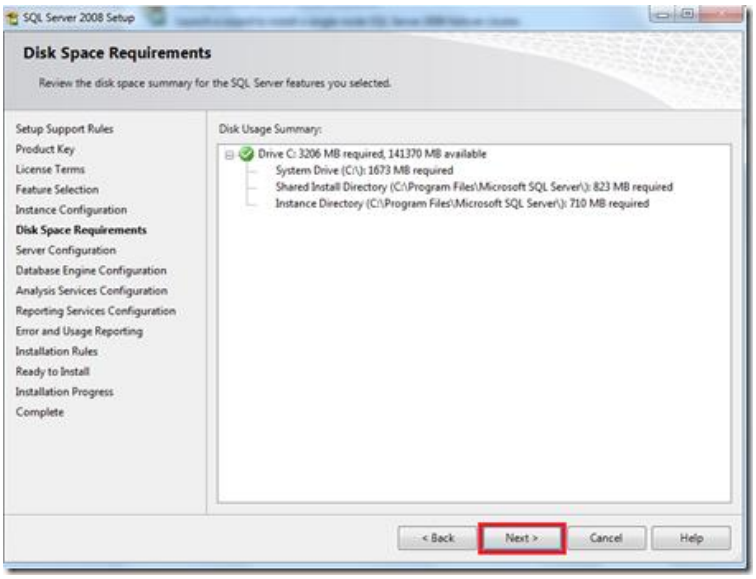


Figura 52 *Resumen de instalación*

Muestra las características para la instalación del SQL Server

- En esta parte muchos desisten de la instalación porque si tan solo seleccionamos siguiente se nos produce un error y se corta la instalación es por eso que según la necesidad de cada equipo se debe escoger la cuenta correcta, en mi caso quiero que toda la autorización sea parte del sistema ya que mi notebook no depende de una red, si fuera el caso escogería autorización para usuarios de red.
- Selecciono "Autorización\Sistema" en todas las área de ingreso y clic en "Siguiente"

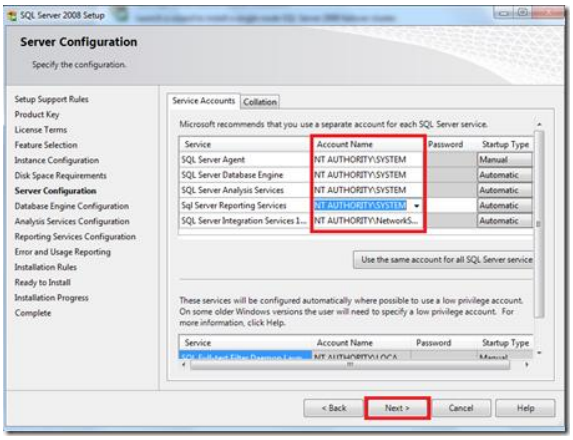


Figura 53 *Configuración de servidor*

Muestra los tipos de conexión a instalar

- La siguiente configuración tiene 2 opciones, yo elijo “Autenticación de Windows”, clic en “agregar un usuario existente” y “siguiente”

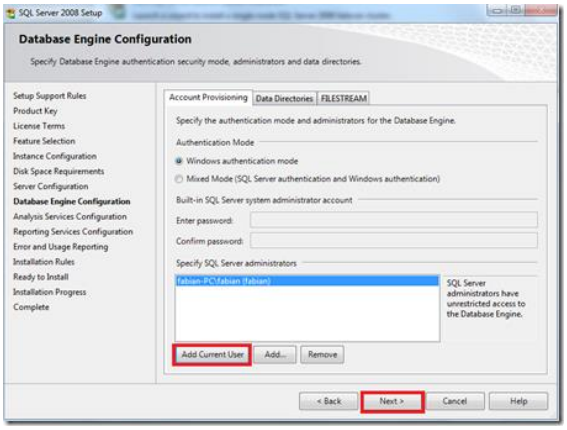


Figura 54 Agregar Uusuario

Muestra la interfaz para la creación del usuario sa de la base de datos

- Para el servicio de análisis debemos asignar un usuario, si queremos que el USER actual sea el mismo para el análisis de SQL seleccionamos “Agregar usuario existente” y “Siguiente”

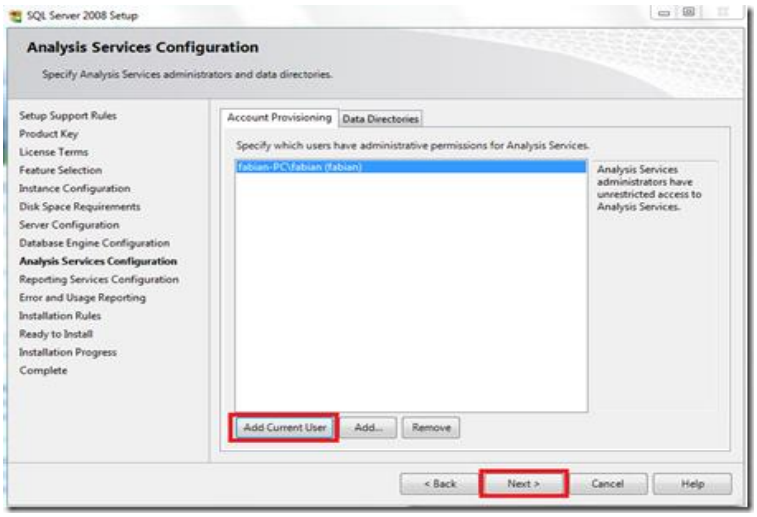


Figura 55 *Configuración de servicios.*

Muestra la configuración a instalar de los servicios que provee el SQL Server

- En la pantalla siguiente seleccionamos qué modo de instalación necesitamos, si queremos integrar con SharePoint o no configurar el servidor de reportes
- En este caso elegiremos una “configuración nativa“ y “Siguiente”

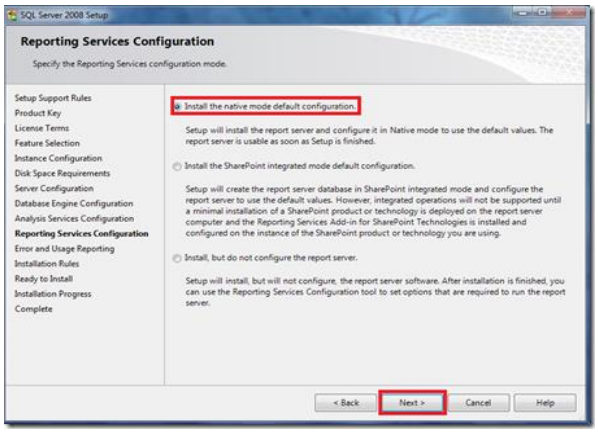


Figura 56 *Guardar configuración*

Muestra detalle de la instalación y envió de error al proveedor.

- En la siguiente ventana seleccionamos si deseamos enviar reportes de errores o características a Microsoft, si lo deseamos damos clic en cada checkbox o simplemente “siguiente” (Recomendable Siguiente).

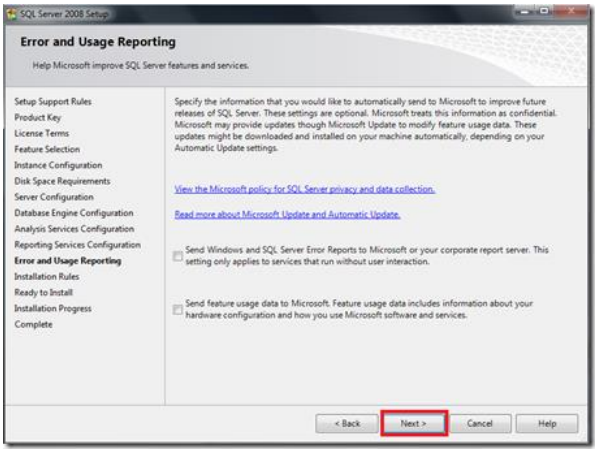


Figura 57 *Enviar Información de errores.*

Muestra la opción de enviar información al proveedor.

- Ahora instalemos todas las reglas de Installation, que son en un número de 11 y “Siguiente”

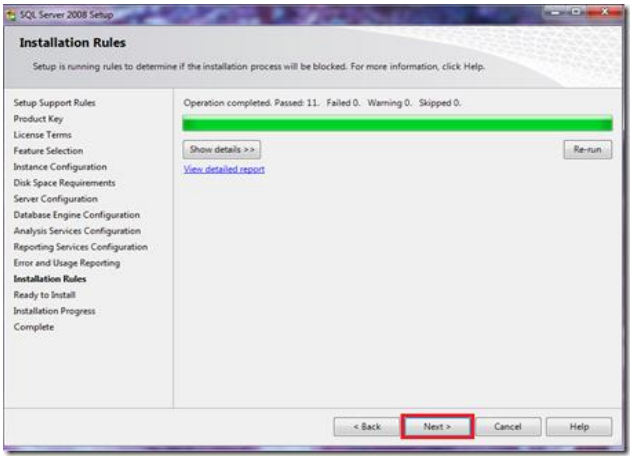


Figura 58 *Proceso de instalación*

- Ahora si está todo listo para instalarse, vemos el resumen de los componentes que se agregarán, y si está todo bien clic en “Instalar”

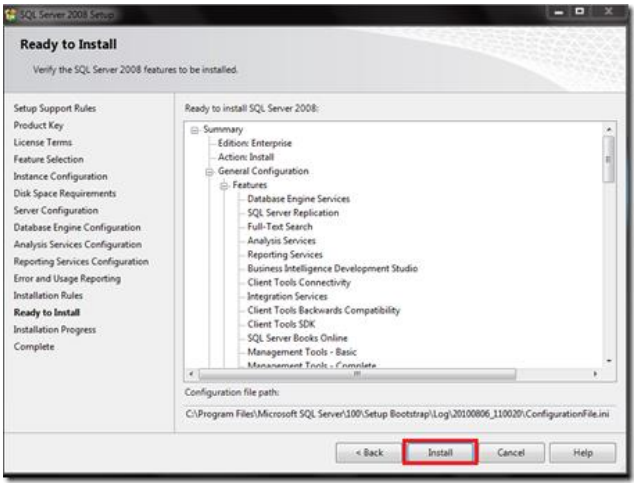


Figura 59 *Resumen de instalación*

- Inicia nuestra instalación, debemos ser pacientes toma alrededor de 45 a 60 minutos, dependiendo de las características del equipo en donde lo instalamos

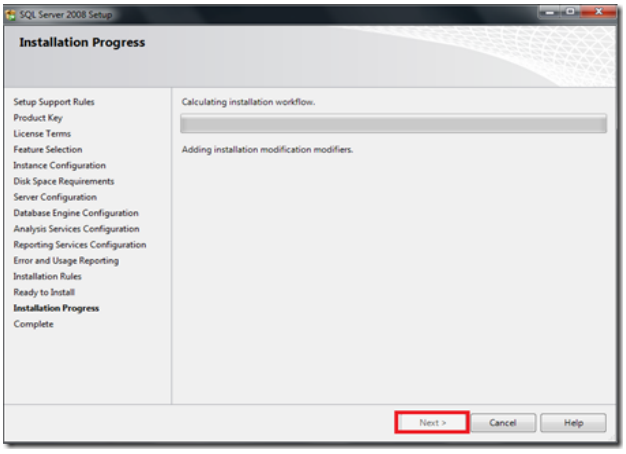


Figura 60 *Proceso de instalación*

- Después de la espera se completa la instalación mostrándonos si existieron errores, en este caso todo se instaló con normalidad y seleccionamos "Siguiente"

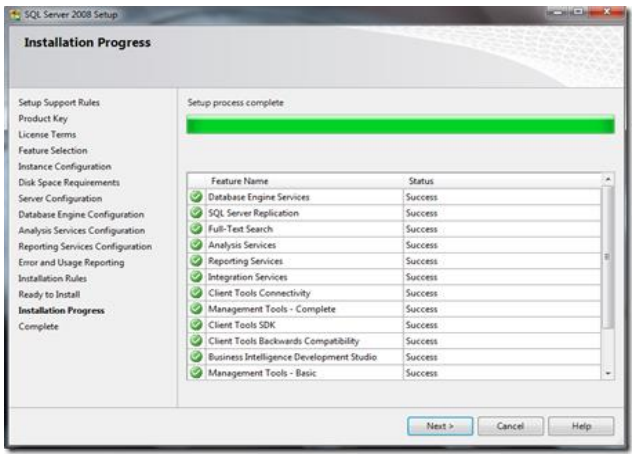


Figura 61 *Instancian*

- La pantalla final nos avisa que SQL Server 2008 está operando e instalado correctamente

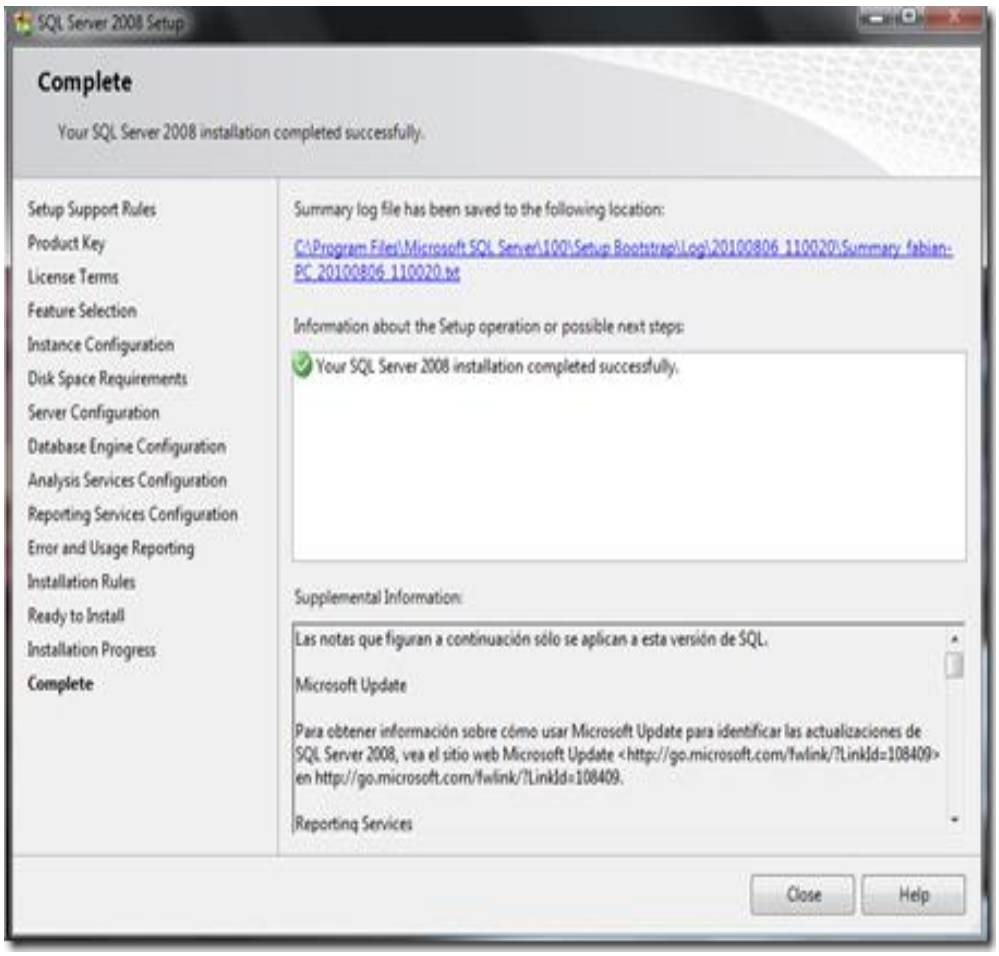


Figura 62 *Instalación completa*

Anexo 1.02. Instalación Visual Studio 2010

Comenzamos la instalación para ello primeramente descomprimimos el instalador si fuera el caso.

Seleccionaremos la carpeta de destino para la descompresión de los ficheros y pulsaremos "Instalar":

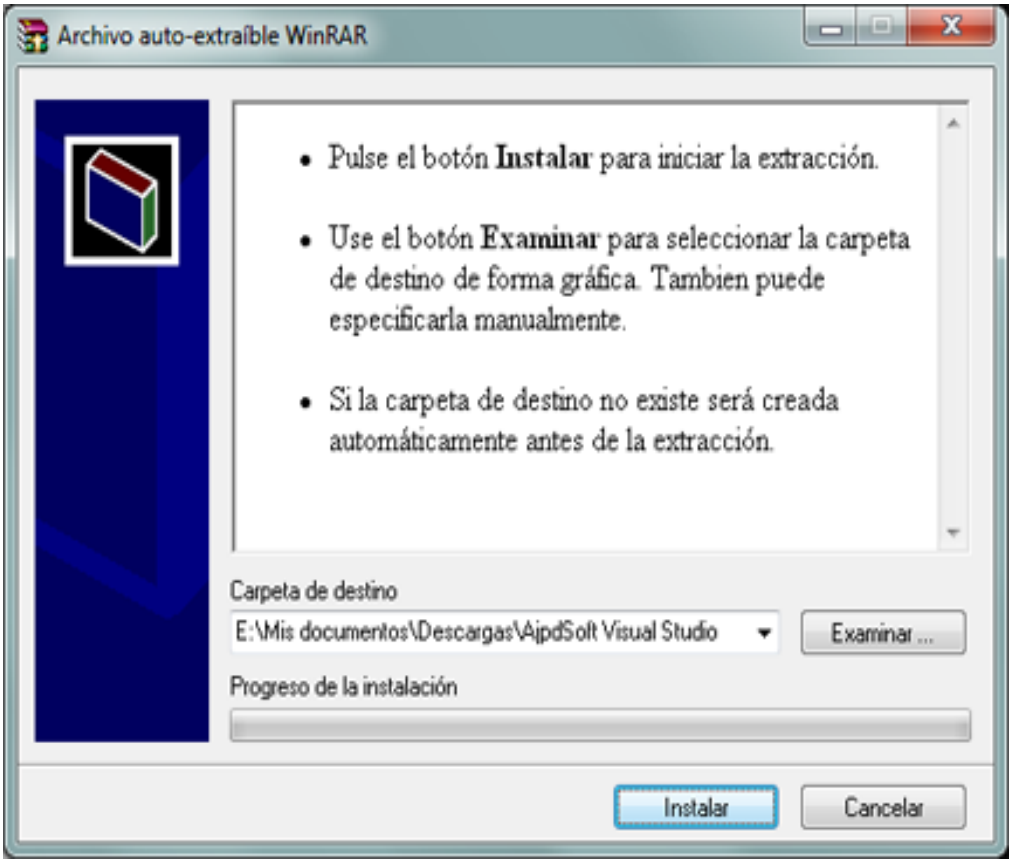


Figura 63 *Extraer instaladores*

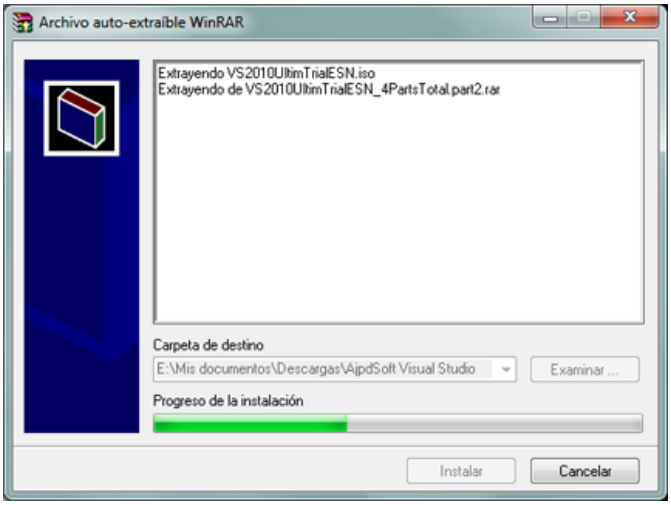


Figura 64 *Proceso de descompresión*

Se iniciará la descompresión de los ficheros descargados en el fichero .iso con la instalación de Microsoft Visual Studio 2010:

Una vez descomprimidos los ficheros, quedará el fichero VS2010Ultimate.iso de 2,5GB, este fichero podremos pasarlo a un DVD para realizar un ejecutable y poder instalar.



Figura 65 *Inicio de instalación*

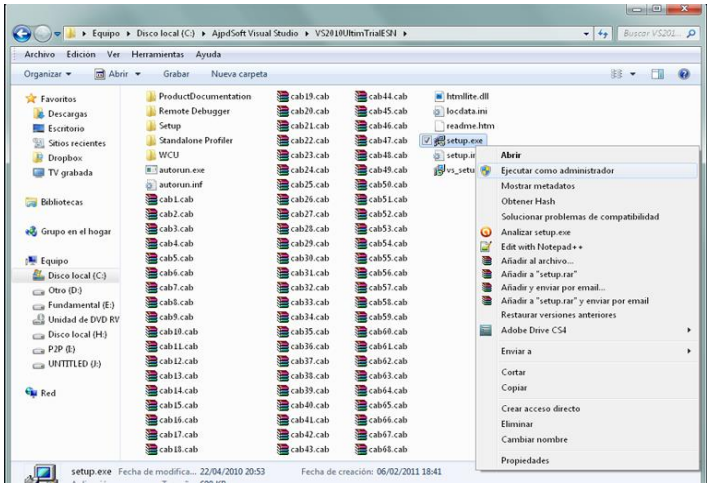


Figura 66 *Ubicación de archivos*

Ejecutamos el archivo setuo.exe.



Figura 67 *Menú de instalación*

En el primer paso de la instalación de Microsoft Visual Studio 2010 Ultimante, desmarcaremos "Sí, enviar a Microsoft Corporation información sobre la instalación" (si no queremos enviar esta información) y pulsaremos "Siguiente":

Leeremos los términos de licencia del software de Microsoft Visual Studio 2010 Ultimate. Si estamos de acuerdo marcaremos "He leído los términos de la licencia y los acepto". Pulsaremos "Siguiente" para continuar:

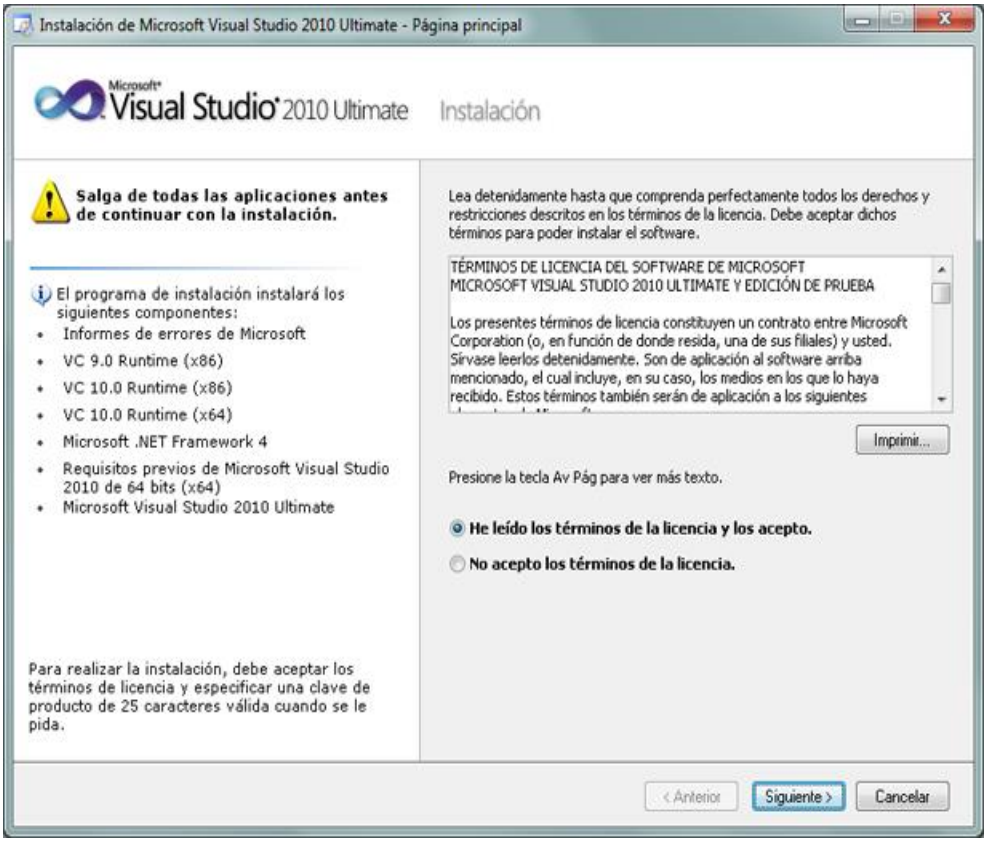


Figura 68 *Inicio de instalación*

A continuación marcaremos "Personalizar" para seleccionar los lenguajes de programación y las herramientas que se quieran instalar. En "Ruta de instalación del producto" indicaremos la unidad y carpeta de destino de la instalación:

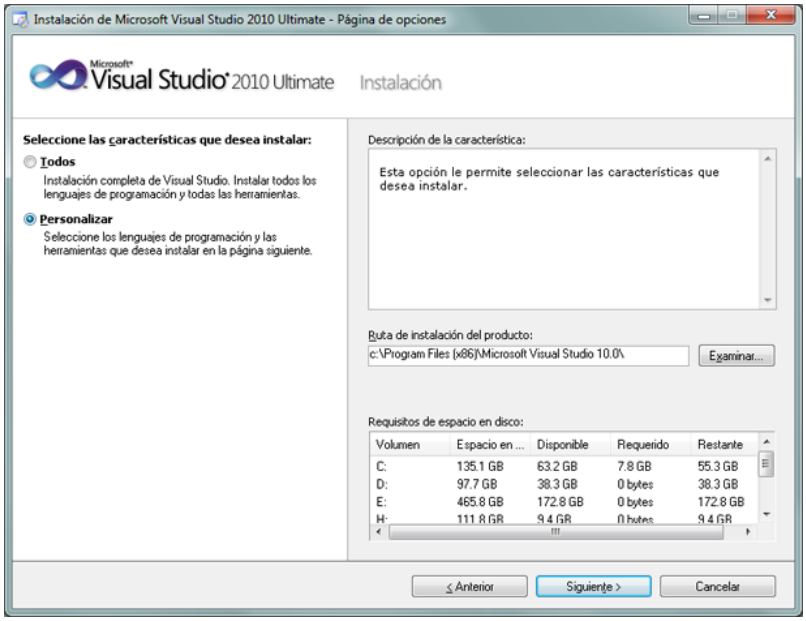


Figura 69 *Proceso de instalación*

Seleccionaremos los lenguajes a instalar:

- Visual Basic.
- Visual C++.
- Visual C#.
- Visual F#.
- Entre otros.

Seleccionaremos también las características a instalar:

Microsoft Office Developer Tools.

Dotfuscator Software Services.

Microsoft SQL Server 2008 Express.

Microsoft sharepoint Developer Tools.

Una vez seleccionadas las características a instalar pulsaremos en el botón "Instalar":

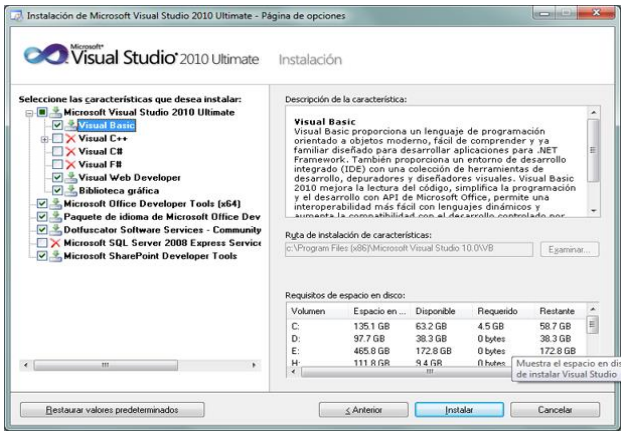


Figura 70 *Proceso de instalación*

Se iniciará la instalación de Microsoft Visual Studio 2010 Ultimate: informe de errores de Microsoft, VC 9.0 Runtime, VC 10.0 Runtime, Microsoft .Net Framework 4, Microsoft Visual Studio 2010 Ultimate, Microsoft ASP .Net, etc.:

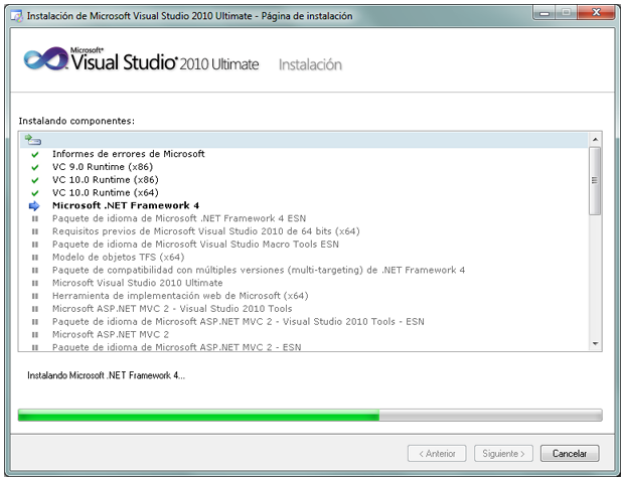


Figura 71 *Proceso de instalación*

Tras la instalación, el asistente para instalar Visual Studio mostrará la siguiente ventana, con el texto "Correcto. Se instaló Visual Studio 2010 y finalizó el programa de instalación". Pulsaremos "Finalizar":

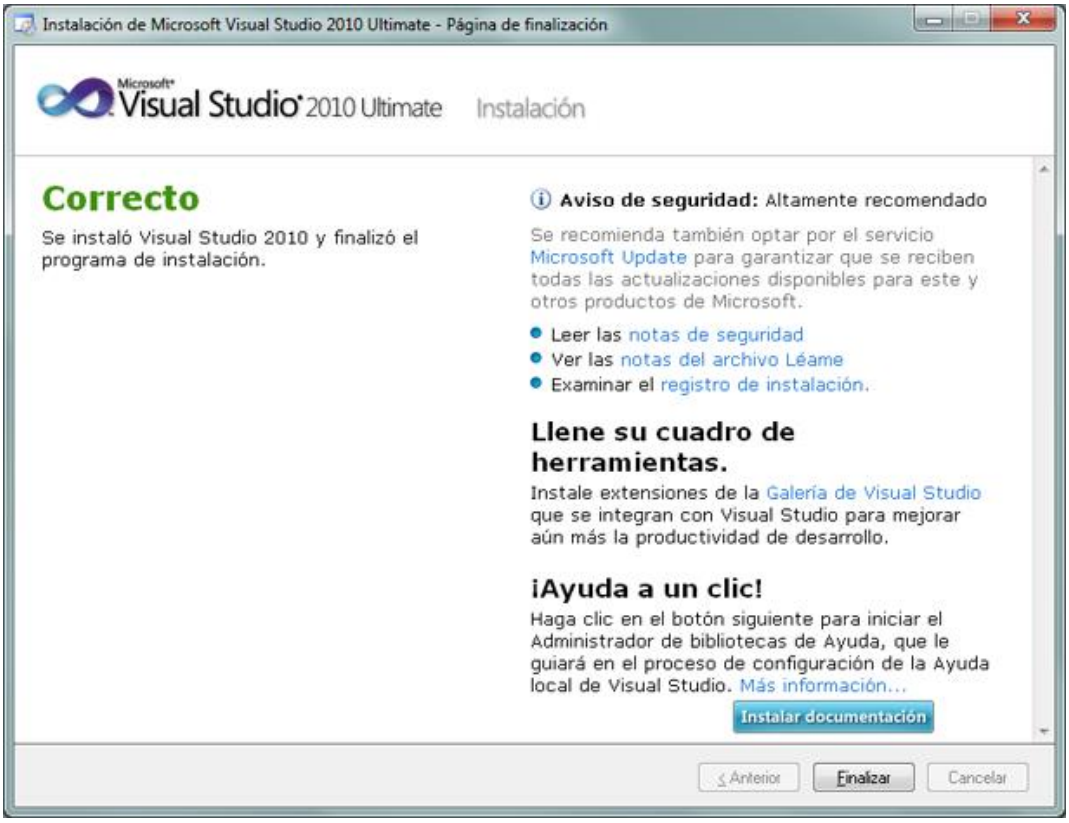


Figura 72 *Instalación correcta*

Tras el proceso de instalación puede que pida reiniciar el equipo:

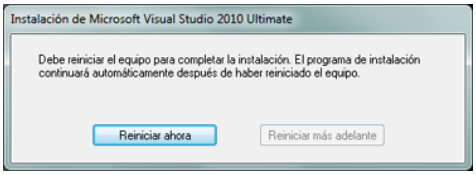


Figura 73 *Reiniciar maquina*

Con el texto: "Debe reiniciar el equipo para completar la instalación. El programa de instalación continuará automáticamente después de haber reiniciado el equipo".

Desde el botón "Iniciar" , "Todos los programas" podremos ver el grupo de programas "Microsoft Visual Studio 2010":

Tras iniciar por primera vez Microsoft Visual Studio 2010 Ultimate, seleccionaremos la configuración de entorno predeterminada, por ejemplo "Configuración de desarrollo de C #":

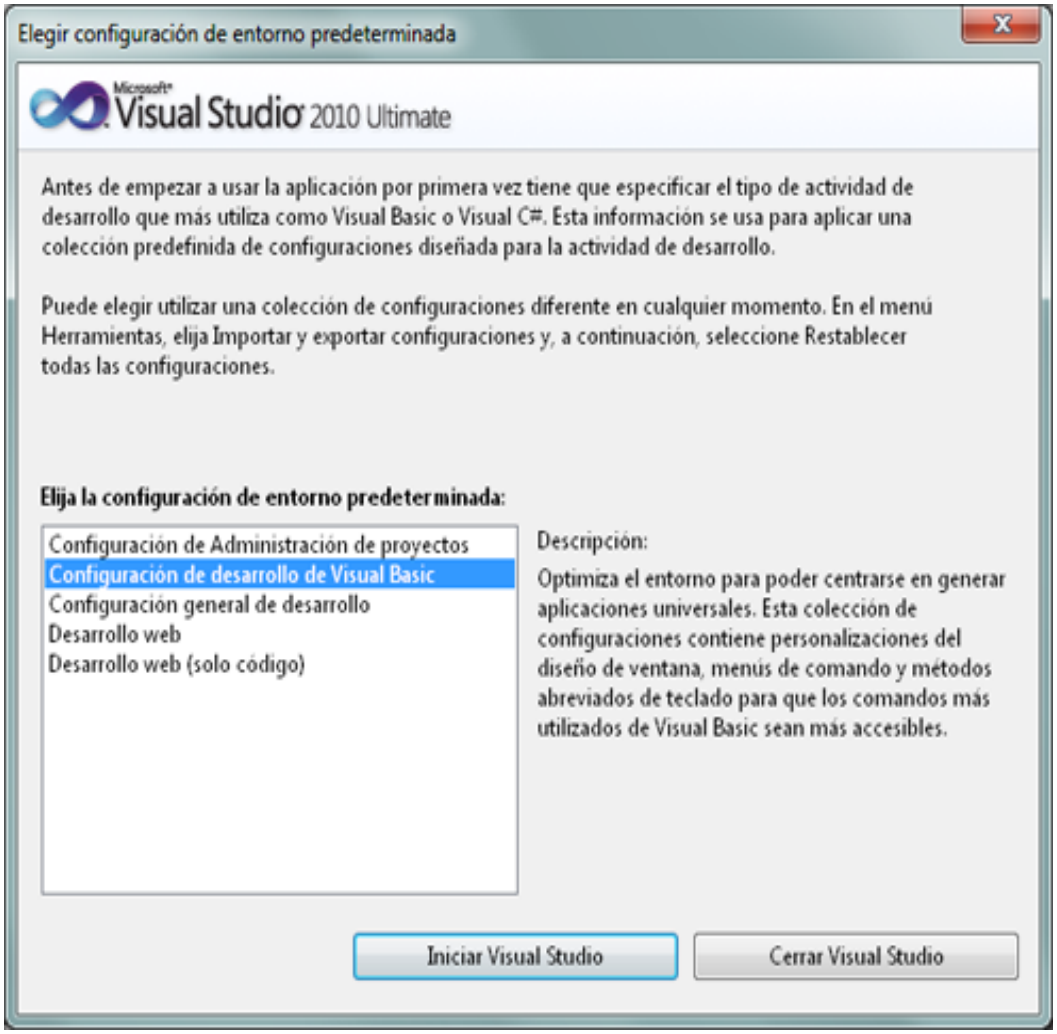


Figura 74 *Configuración Visual Studio 2010*

Muestra la configuración de visual studio 2010

Y ya podremos disponer de este entorno de programación:

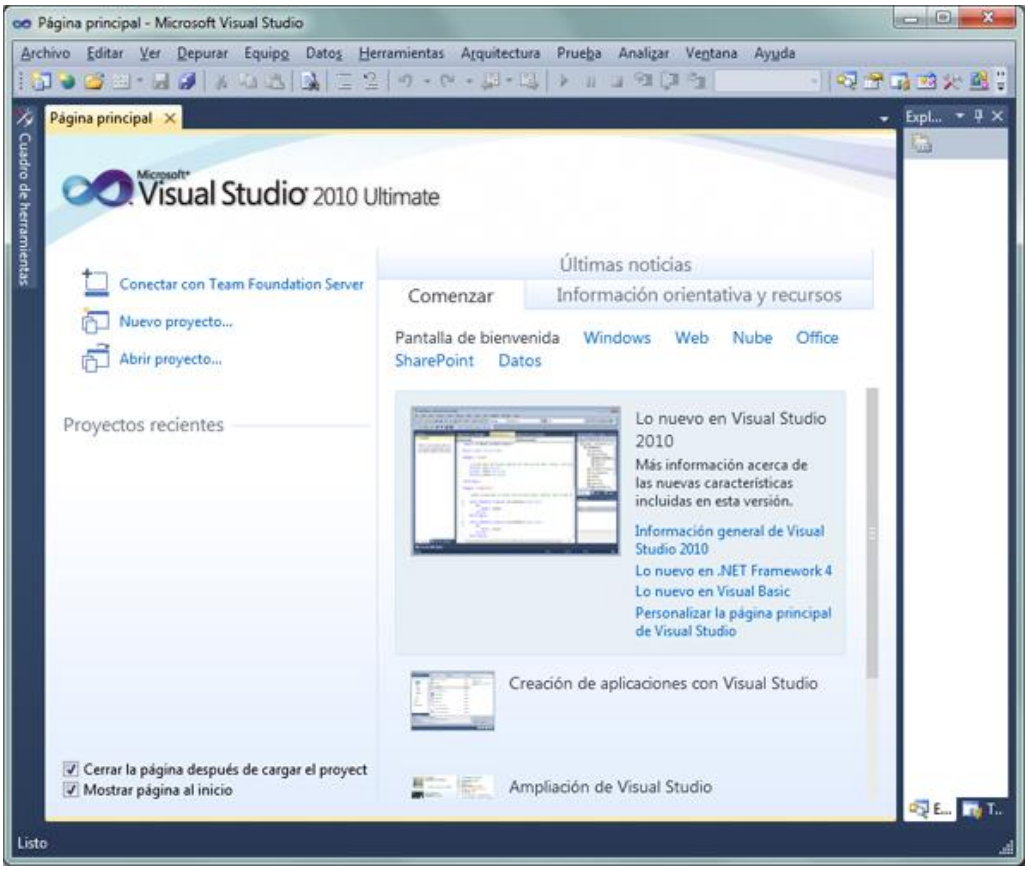


Figura 75 *Visual Studio 2010*

Muestra la página principal de visual studio 2010

Anexo A 2. Código fuente de la lógica principal del negocio.

Creación de facturas

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using AccesoDatos;
using LogicaNegocio;
using System.Data.Linq;
using System.IO;
using System.Data;

namespace VentasAtiempo.Administrador.Facturacion
{
   public partial class WfrmFacturacionNuevo : System.Web.UI.Page
   {
      DataTable t = new DataTable();
      tbl_cab_fact CabeceraFactura = new tbl_cab_fact();
      tbl_det_fact DetalleFactura = new tbl_det_fact();
      tbl_usuario UsuarioVendedor = new tbl_usuario();
      List<tbl_clienprof> ClienteProformaList = new List<tbl_clienprof>();
      tbl_clienprof ClienteProformaInfo = new tbl_clienprof();
      List<tbl_producto> ProductoList = new List<tbl_producto>();
      tbl_producto ProductoInfo = new tbl_producto();


      protected void Page_Load(object sender, EventArgs e)
      {
         CargaControles();

         if (!IsPostBack)
         {
            if (Session["docinfo"] != null)
            {
               UsuarioVendedor = new tbl_usuario();
               UsuarioVendedor = Session["docinfo"] as tbl_usuario;
               txtVendedor.Text = string.Format("{0}", UsuarioVendedor.user_usu);
               txtVendedorId.Text = string.Format("{0}", UsuarioVendedor.cod_usu);
               Label1.Text = string.Format("{0}", UsuarioVendedor.cod_usu);
            }


         }
      }

      //Cargar los controles de la pagina de facturacion.
```

```
    private void CargaControles()
    {
        txtNumero.ReadOnly = true;
        txtFecha.ReadOnly = true;
        txtFechaLarga.ReadOnly = true;
        txtVendedor.ReadOnly = true;
        txtVendedorId.ReadOnly = true;
        txtNombreCliente.ReadOnly = true;
        txtNombreProducto.ReadOnly = true;
        txtValor.ReadOnly = true;
        txtIdCliente.ReadOnly = true;
        ddlcriterio.Enabled = false;
        ddlcriterioProducto.Enabled = false;
        txtIdCodigoProducto.ReadOnly = true;
        txtCantidad.ReadOnly = true;
        txtDescuento.ReadOnly = true;
        txtObervaciones.ReadOnly = true;
        txtIngresoDescuentoManual.ReadOnly = true;
        txtIngresoIva.ReadOnly = true;
        txtSubtotal.ReadOnly = true;
        txtValorDesacuentoManual.ReadOnly = true;
        txtValorIva.ReadOnly = true;
        txtTotal.ReadOnly = true;
        btnBuscar.Enabled = false;
        btnBuscarProducto.Enabled = false;
        btnAgregar.Enabled = false;
        btnGuardarFactura.Enabled = false;
    }

    //Para crear nueva Factura
    protected void btnNuevo_Click(object sender, EventArgs e)
    {
        NuevaFactura(Convert.ToInt32(Request["idusuario"]));

        int cod = LogicaFactura.Id_Codigo();
        txtFecha.Text = System.DateTime.Now.ToString("yyyy-MM-dd");
        txtFechaLarga.Text = System.DateTime.Now.ToString(" Quito, dd' de
'MMMM' de 'yyyy");
        txtNumero.Text = cod.ToString();
        txtIdCliente.ReadOnly = false;
        ddlcriterio.Enabled = true;
        btnBuscar.Enabled = true;
        VaciaControles();
    }

    private void NuevaFactura(int id)
    {
        if (id == 0)
        {
            Nuevo();
```

```
        }
        else
        {
            //ProductoInfo = LogicaProducto.ObtenerProductoIdinfo(id);
            //Modificacion(ProductoInfo);
        }

    }
    private void Nuevo()
    {
        try
        {
            CabeceraFactura = new tbl_cab_fact();

            //LogicaProforma.(CabeceraProforma);

        }
        catch (Exception)
        {

        }
    }
    private void VaciaControles()
    {

        txtNombreCliente.Text = "";
        txtNombreProducto.Text = "";
        txtValor.Text = "";
        txtIdCliente.Text = "";
        txtIdCodigoProducto.Text = "";
        txtCantidad.Text = "";
        txtDescuento.Text = "0";
        txtObervaciones.Text = "";
        txtSubtotal.Text = "";
        txtValorDesacuentoManual.Text = "";
        txtValorIva.Text = "";
        txtTotal.Text = "";

    }

    protected void btnAgregar_Click1(object sender, EventArgs e)
    {


        if (txtValor.Text != "" && txtCantidad.Text != "" && txtIngresoIva.Text !=
"")
        {
            Decimal Total;
            Decimal TotalDesacuento;
            Decimal DescuentoSubtotal;
```

```
            Decimal subiva;
            Decimal TotalDesacuentoGrid;
            Total = Convert.ToInt32(txtCantidad.Text) *
Convert.ToDecimal(txtValor.Text);
            DataGridItem reglon = dgr_lista.SelectedItem;
            DataTable detalles = null;
            if (Session["dgv"] != null)
               detalles = Session["dgv"] as DataTable;
            else
            {
               detalles = new DataTable();
               detalles.Columns.Add("ID");
               detalles.Columns.Add("MODELO");
               detalles.Columns.Add("CANTIDAD");
               detalles.Columns.Add("PRECIO");
               detalles.Columns.Add("TOTAL");
               detalles.Columns.Add("DESCUENTO");
               detalles.Columns.Add("OBSERVACION");
            }
            Total = Convert.ToInt32(txtCantidad.Text) *
Convert.ToDecimal(txtValor.Text);
            if (txtSubtotal.Text == "")
            {
               txtSubtotal.Text = "0";
            }
            TotalDesacuento = (Total - (Total *
(Convert.ToDecimal(txtDescuento.Text) / 100))) +
Convert.ToDecimal(txtSubtotal.Text);
            TotalDesacuentoGrid = Total - (Total *
(Convert.ToDecimal(txtDescuento.Text) / 100));
            DataRow registro = detalles.NewRow();
            registro[0] = txtIdCodigoProducto.Text;
            registro[1] = txtNombreProducto.Text;
            registro[2] = txtCantidad.Text;
            registro[3] = txtValor.Text;
            registro[4] = Convert.ToString(TotalDesacuentoGrid);
            registro[5] = txtDescuento.Text;
            registro[6] = txtObervaciones.Text;
            detalles.Rows.Add(registro);
            dgr_lista.DataSource = detalles;
            dgr_lista.DataBind();
            Session["dgv"] = detalles;
            VaciaControles();
            txtIdCliente.ReadOnly = false;
            ddlcriterio.Enabled = true;
            btnBuscar.Enabled = true;
            DescuentoSubtotal = (TotalDesacuento *
(Convert.ToDecimal(txtIngresoDescuentoManual.Text) / 100));
            txtValorDesacuentoManual.Text = Convert.ToString(DescuentoSubtotal);
```

```
            subiva = (TotalDesacuento - DescuentoSubtotal) *
(Convert.ToDecimal(txtIngresoIva.Text) / 100);
            txtSubtotal.Text = Convert.ToString(TotalDesacuento);
            txtValorIva.Text = Convert.ToString(subiva);
            txtTotal.Text = Convert.ToString((TotalDesacuento + subiva) -
DescuentoSubtotal);
            btnGuardarFactura.Enabled = true;
         }
         else
         {
            if (txtValor.Text == "")
            {
               lblmsj.Text = "Campo Costo Necesario";
               lblmsj.Visible = true;
            }
            if (txtCantidad.Text == "")
            {
               lblmsj.Text = "Campo Cantidad Necesario";
               lblmsj.Visible = true;
            }
            if (txtIngresoIva.Text == "")
            {
               lblmsj.Text = "Campo Iva Necesario";
               lblmsj.Visible = true;
            }

         }
      }

      protected void Unnamed4_Click(object sender, EventArgs e)
      {
         Guardando(Convert.ToInt32(Request["idusuario"]));
         GuardandoDetalle(Convert.ToInt32(Request["idusuario"]));
      }
      private void GuardandoDetalle(int id)
      {
         if (id == 0)
         {
            GuardarDetalle();
         }
      }
      private void GuardarDetalle()
      {
         foreach (DataGridItem item in dgr_lista.Items)
         {
            try
            {
               DetalleFactura = new tbl_det_fact();
               DetalleFactura.cod_cfac = Convert.ToInt32(txtNumero.Text);
               DetalleFactura.cod_prod = Convert.ToInt32(item.Cells[0].Text);
```

```
            DetalleFactura.can_dfac = item.Cells[2].Text;
            DetalleFactura.cuni_dfac = item.Cells[3].Text;
            DetalleFactura.pdes_dfac = item.Cells[5].Text;
            DetalleFactura.desc_dfac = item.Cells[6].Text;
            DetalleFactura.cod_usu = txtVendedorId.Text;
            LogicaFactura.SaveDetalle(DetalleFactura);
            lblMensaje.Text = "Datos Guardados";
            lblMensaje.Visible = true;
        }
        catch (Exception ex)
        {
            lblMensaje.Text = ex.Message;
            lblMensaje.Visible = true;
        }
        string valorcol2 = item.Cells[1].Text;

    }

}
private void Guardando(int id)
{
    if (id == 0)
    {
        Guardar();
    }
}
private void Guardar()
{
    try
    {
        CabeceraFactura = new tbl_cab_fact();
        CabeceraFactura.cod_usu = txtVendedorId.Text;
        CabeceraFactura.fec_cfac = txtFechaLarga.Text;
        CabeceraFactura.fecc_cfac = txtFecha.Text;
        CabeceraFactura.id_clienprof = Convert.ToInt32(txtId_ClienProf.Text);
        CabeceraFactura.ven_cfac = txtVendedor.Text;
        CabeceraFactura.sub_cfac = txtSubtotal.Text;
        CabeceraFactura.des_cfac = txtValorDesacuentoManual.Text;
        CabeceraFactura.iva_cfac = txtValorIva.Text;
        CabeceraFactura.son_cfac = txtTotalLetras.Text;
        CabeceraFactura.nota_cfac = txtNota.Text;

        LogicaFactura.Save(CabeceraFactura);
        lblMensaje.Text = "Datos Guardados";
        lblMensaje.Visible = true;
    }
    catch (Exception ex)
    {
        lblMensaje.Text = ex.Message;
        lblMensaje.Visible = true;
```

```
    }
}

protected void txtFecha_TextChanged(object sender, EventArgs e)
{

}

protected void txtFechaLarga_TextChanged(object sender, EventArgs e)
{

}

protected void txtIdCliente_TextChanged(object sender, EventArgs e)
{

}

protected void ddlcriterio_SelectedIndexChanged(object sender, EventArgs e)
{

}

protected void txtNombreCliente_TextChanged(object sender, EventArgs e)
{

}

protected void txtNumero_TextChanged(object sender, EventArgs e)
{

}

protected void txtVendedor_TextChanged(object sender, EventArgs e)
{

}

protected void txtVendedorId_TextChanged(object sender, EventArgs e)
{

}

protected void txtOferta_TextChanged(object sender, EventArgs e)
{

}

protected void txtIdCodigoProducto_TextChanged(object sender, EventArgs e)
{
```

```
    }

    protected void btnBuscarProducto_Click1(object sender, EventArgs e)
    {
        string op = ddlcriterioProducto.SelectedValue;
        try
        {
            switch (op)
            {
                case "Id":
                    ProductoList =
LogicaProducto.ObtenerProductoId(Convert.ToInt32(txtIdCodigoProducto.Text));
                    CargaNombreProducto(ProductoList);
                    break;
                case "nom":
                    ProductoList =
LogicaProducto.ObtenerProductoNombre(txtIdCodigoProducto.Text);
                    CargaNombreProducto(ProductoList);
                    break;

                default:
                    break;

            }
        }
        catch (Exception)
        {
            throw;
        }
    }
    private void CargaNombreProducto(List<tbl_producto> Lista)
    {
        if (Lista != null)
        {
            txtNombreProducto.Text = Lista[0].mod_pro;
            txtValor.Text = Convert.ToString(Lista[0].cos_pro);

            if (txtIdCodigoProducto.Text != "" && txtNombreProducto.Text != "")
            {
                txtIdCodigoProducto.ReadOnly = false;
                btnAgregar.Enabled = true;
                txtCantidad.ReadOnly = false;
                txtDescuento.ReadOnly = false;
                txtObervaciones.ReadOnly = false;
                ddlcriterio.Enabled = true;
                btnBuscar.Enabled = true;
                btnBuscarProducto.Enabled = true;
                ddlcriterioProducto.Enabled = true;
                txtIdCliente.ReadOnly = false;
```

```
            }
        }

    }

    protected void txtNombreProducto_TextChanged(object sender, EventArgs e)
    {

    }

    protected void txtValorIva_TextChanged(object sender, EventArgs e)
    {

    }

    protected void btnBuscar_Click(object sender, EventArgs e)
    {
        string op = ddlcriterio.SelectedValue;
        try
        {
            switch (op)
            {
                case "Ruc":
                    ClienteProformaList =
LogicaClienteProforma.ObtenerClienteProformaXruc(txtIdCliente.Text);
                    CargarDatosCliente(ClienteProformaList);
                    break;
                case "nom":
                    ClienteProformaList =
LogicaClienteProforma.ObtenerClienteProformaNombre(txtIdCliente.Text);
                    CargarDatosCliente(ClienteProformaList);
                    break;
                default:
                    break;

            }
        }
        catch (Exception)
        {

            throw;
        }
    }
    private void CargarDatosCliente(List<tbl_clienprof> Lista)
    {
        if (Lista != null)
        {
            txtNombreCliente.Text = Lista[0].nom_clienprof;

            txtId_ClienProf.Text = Convert.ToString(Lista[0].id_clienprof);
```

```
            txtDireccionCliente.Text = Lista[0].dir_clienprof.ToString();
            txtTelefonoCliente.Text = Lista[0].tel1_clienprof.ToString();

            if (txtNombreCliente.Text != "" && txtId_ClienProf.Text != "")
            {
               txtCantidad.ReadOnly = true;
               txtDescuento.ReadOnly = true;
               txtObervaciones.ReadOnly = true;
               btnBuscarProducto.Enabled = true;
               ddlcriterioProducto.Enabled = true;
               txtIdCodigoProducto.ReadOnly = false;
               txtIdCliente.ReadOnly = false;
               ddlcriterio.Enabled = true;
               btnBuscar.Enabled = true;
            }
         }
      }

   }
}
```

Código fuente Proformas

```
namespace VentasAtiempo.Administrador
{
   public partial class WfrmClienteProformaNuevo : System.Web.UI.Page
   {
      List<tbl_clienprof> ClienteProformaList = new List<tbl_clienprof>();
      tbl_clienprof ClienteProformaInfo = new tbl_clienprof();
      protected void Page_Load(object sender, EventArgs e)
      {
         if (!IsPostBack)
         {
            CargarClienteProforma();
            int id = 0;
            if (Request["id_clienprof"] != null)
            {
               id = Convert.ToInt32((Request["id_clienprof"]));
               if(id > 0)
               {
                  cargaClienteProforma(id);
               }
            }
         }


      }
      private void CargarClienteProforma()
      {
         ClienteProformaList = LogicaClienteProforma.ObtenerClienteProforma();
         if (ClienteProformaList != null)
         {
            dgr_ClienteProforma.DataSource = ClienteProformaList;
            dgr_ClienteProforma.DataBind();
         }
      }

      private void cargaClienteProforma(int id)
      {
         ClienteProformaInfo =
LogicaClienteProforma.ObtenerCLienteProformaInfo(id);
         if (ClienteProformaInfo != null)
         {
            Txt_Cliente.Text = ClienteProformaInfo.nom_clienprof;
            Txt_Ruc.Text = ClienteProformaInfo.ruc_clienprof;
            Txt_Atencion.Text = ClienteProformaInfo.rep_clienprof;
            Txt_Direccion.Text = ClienteProformaInfo.dir_clienprof;
            Txt_Telefono.Text = ClienteProformaInfo.tel1_clienprof;
            Txt_Telefono2.Text = ClienteProformaInfo.tel2_clienprof;
            Txt_Ext.Text = ClienteProformaInfo.ext1;
```

```
            Txt_Ext2.Text = ClienteProformaInfo.ext2;
            Txt_mail.Text = ClienteProformaInfo.ema_clienprof;

        }
    }

    protected void lnkSave_Click(object sender, EventArgs e)
    {

    }
    private void LimpiaText()
    {
        Txt_Atencion.Text = "";
        Txt_Cliente.Text = "";
        Txt_Direccion.Text = "";
        Txt_Ext.Text = "";
        Txt_Ext2.Text = "";
        Txt_mail.Text = "";
        Txt_Ruc.Text = "";
        Txt_Telefono.Text = "";
        Txt_Telefono2.Text = "";
    }
    //Metodo para guardar nuevo cliente proforma
    private void Guardar()
    {
        try
        {
            ClienteProformaInfo = new tbl_clienprof();
            ClienteProformaInfo.ruc_clienprof = Txt_Ruc.Text;
            ClienteProformaInfo.nom_clienprof = Txt_Cliente.Text;
            ClienteProformaInfo.rep_clienprof = Txt_Atencion.Text;
            ClienteProformaInfo.dir_clienprof = Txt_Direccion.Text;
            ClienteProformaInfo.tel1_clienprof = Txt_Telefono.Text;
            ClienteProformaInfo.ext1 = Txt_Ext.Text;
            ClienteProformaInfo.tel2_clienprof = Txt_Telefono2.Text;
            ClienteProformaInfo.ext2 = Txt_Ext2.Text;
            ClienteProformaInfo.ema_clienprof = Txt_mail.Text;
            LogicaClienteProforma.Save(ClienteProformaInfo);

            lblGuardar.Text = "Datos Guardados";
            lblGuardar.Visible = true;
            LimpiaText();

        }
        catch (Exception ex)
        {
            lblGuardar.Text = ex.Message;
            lblGuardar.Visible = true;
        }
    }
```

```
    private void Guardando(int id)
    {
        if (id == 0)
        {
            Guardar();
        }
        else
        {
            ClienteProformaInfo =
LogicaClienteProforma.ObtenerCLienteProformaInfo(id);
     Modificacion(ClienteProformaInfo);
        }
    }

    private void Modificacion(tbl_clienprof ClienteProformaModify)
    {
        try
        {


            ClienteProformaInfo.ruc_clienprof = Txt_Ruc.Text;
            ClienteProformaInfo.nom_clienprof = Txt_Cliente.Text;
            ClienteProformaInfo.rep_clienprof = Txt_Atencion.Text;
            ClienteProformaInfo.dir_clienprof = Txt_Direccion.Text;
            ClienteProformaInfo.tel1_clienprof = Txt_Telefono.Text;
            ClienteProformaInfo.ext1 = Txt_Ext.Text;
            ClienteProformaInfo.tel2_clienprof = Txt_Telefono2.Text;
            ClienteProformaInfo.ext2 = Txt_Ext2.Text;
            ClienteProformaInfo.ema_clienprof = Txt_mail.Text;
            LogicaClienteProforma.Modify(ClienteProformaModify);
            lblGuardar.Text = "Datos Modificados";
            lblGuardar.Visible = true;
        }
        catch (Exception x)
        {

            lblGuardar.Text = x.Message;
            lblGuardar.Visible = true;
        }


    }

    protected void imgSave_Click(object sender, ImageClickEventArgs e)
    {
        Guardando(Convert.ToInt32(Request["id_clienprof"]));
    }

    protected void Btn_Save_Click(object sender, EventArgs e)
```

```
    {
        Guardando(Convert.ToInt32(Request["id_clienprof"]));
    }

    protected void imgBuscar_Click(object sender, ImageClickEventArgs e)
    {
        string op = ddlcriterio.SelectedValue;
        try
        {
            switch (op)
            {
                case "Ruc":
                    ClienteProformaList =
LogicaClienteProforma.ObtenerClienteProformaXruc(txt_busqueda.Text);
                    CargaDgv(ClienteProformaList);
                    break;
                case "nom":
                    ClienteProformaList =
LogicaClienteProforma.ObtenerClienteProformaNombre(txt_busqueda.Text);
                    CargaDgv(ClienteProformaList);
                    break;
                default:
                    break;

            }
        }
        catch (Exception)
        {

            throw;
        }
    }
    private void CargaDgv(List<tbl_clienprof> Lista)
    {
        if (Lista != null)
        {
            dgr_ClienteProforma.DataSource = Lista;
            dgr_ClienteProforma.DataBind();
        }
    }
    protected void lnkNuevo_Click(object sender, EventArgs e)
    {
        Nuevo();
    }

    protected void imgNuevo_Click(object sender, ImageClickEventArgs e)
    {
        Nuevo();
    }
```

```
        private void Nuevo()
        {
            Response.Redirect("~/Administrador/WfrmClienteProformaNuevo.aspx",
true);

        }

        protected void dgr_ClienteProforma_SelectedIndexChanged(object sender,
EventArgs e)
        {

        }

        protected void dgr_ClienteProforma_ItemCommand(object source,
DataGridCommandEventArgs e)
        {
            int codigo =
Convert.ToInt32((((Label)e.Item.Cells[2].FindControl("lbl_Id_Cliente")).Text));

            if (e.CommandName == "Editar")
            {

Response.Redirect("~/Administrador/WfrmClienteProformaNuevo.aspx?id_clienpro
f=" + codigo.ToString(), true);
            }
            if (e.CommandName == "Eliminar")
            {
                ClienteProformaInfo =
LogicaClienteProforma.ObtenerCLienteProformaInfo(codigo);
                LogicaClienteProforma.Delete(ClienteProformaInfo);
                CargarClienteProforma();
            }
        }

        protected void ddlcriterio_SelectedIndexChanged(object sender, EventArgs e)
        {

        }

        protected void txt_busqueda_TextChanged(object sender, EventArgs e)
        {

        }
    }
}
```

Anexo A 3. Cronograma de actividades

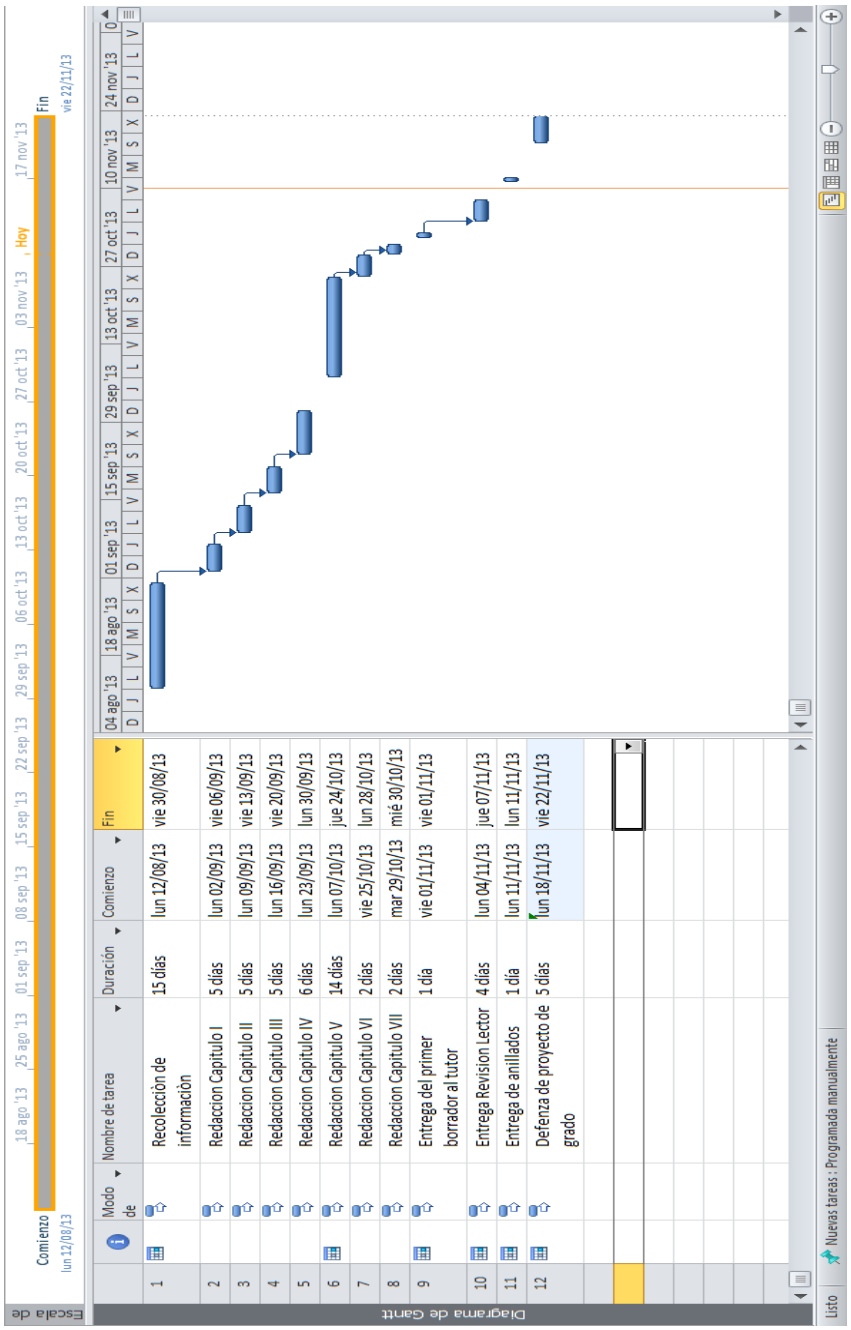


Figura 76 *Cronograma de Actividades*

Muestra gráficamente el cronograma de actividades utilizado para el desarrollo del sistema

Anexo A 4. Manual de Usuario

Ingreso al sistema

El presente manual se ha desarrollado con el propósito de explicar claramente la funcionabilidad del sistema para los usuarios que van a administrarlo haciéndolo de una manera gráfica.

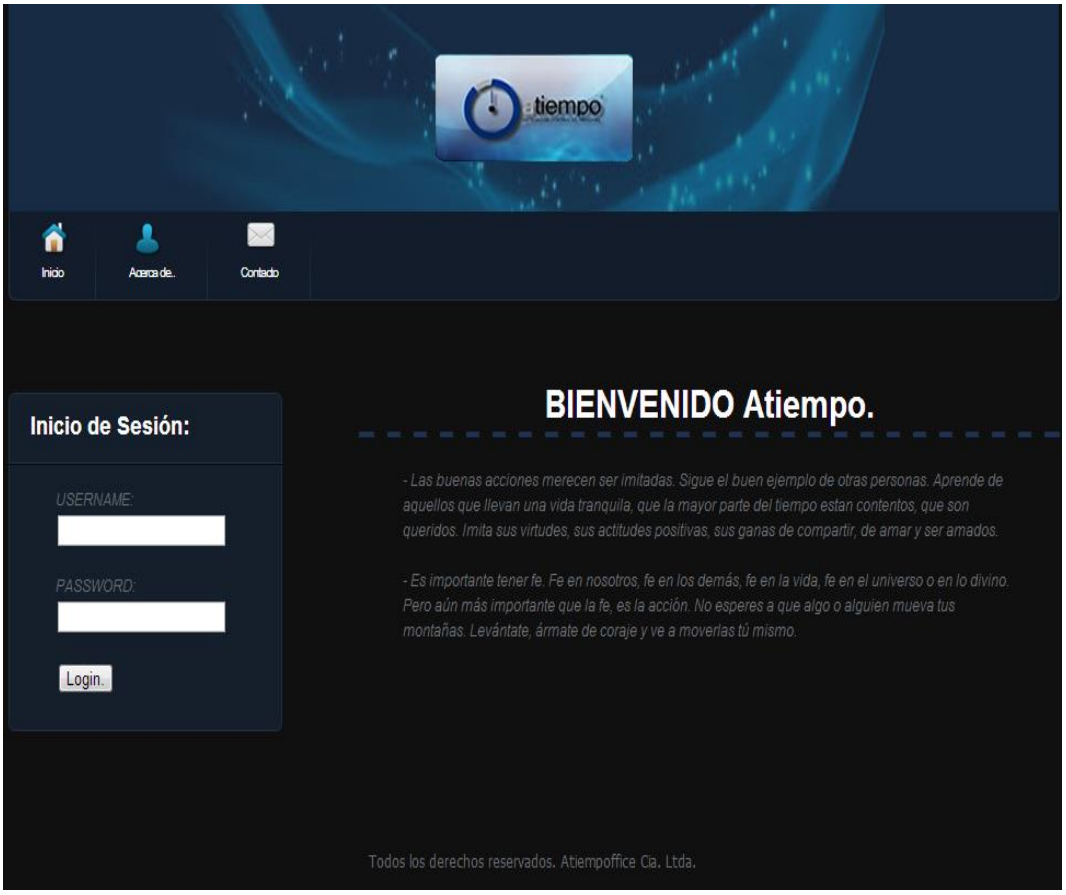


Figura 77 *Página principal*

Muestra gráficamente la interfaz principal del software

Muestra la página de error de usuario no registrado



Figura 78 *Interfaz de error*

Muestra gráficamente la interfaz de error al momento de ingresar al sistema.

Menú principal



Figura 79 *Interfaz Menú principal*

Muestra gráficamente la interfaz del menú principal del sistema

Ingresar nuevos productos

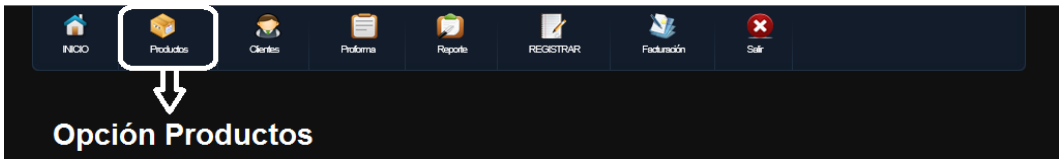


Figura 80 *Ingreso de productos*

Muestra gráficamente la interfaz para el ingresos, eliminación y actualización de productos.

Ventana principal productos

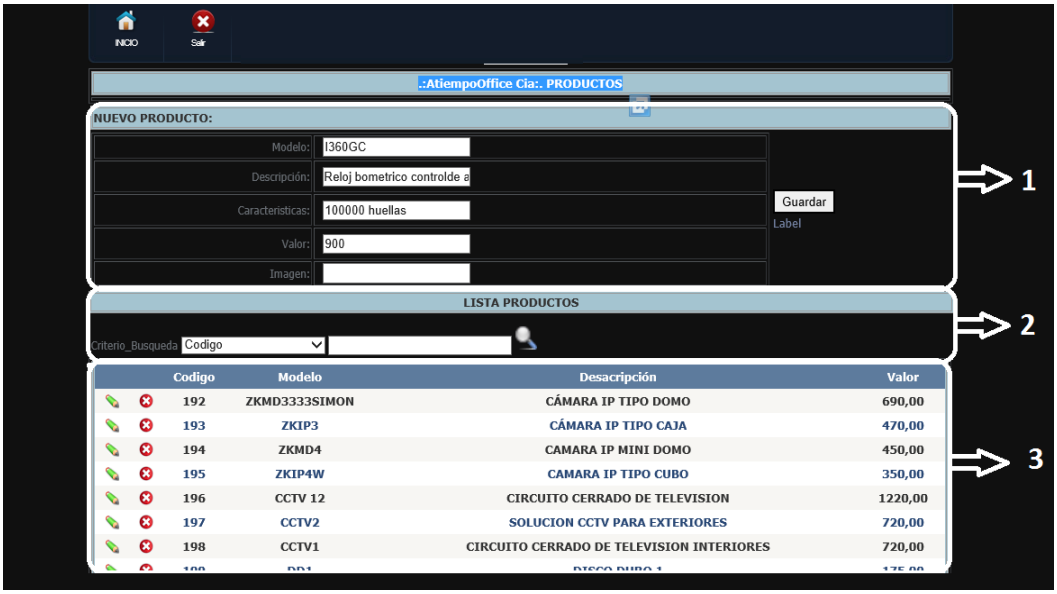


Figura 81 *Interfaz de productos*

Muestra gráficamente la interfaz de productos

1.- Ingrese nuevos datos del producto, Modelo, descripción, características, valor y una imagen.

2.- en criterios de búsqueda seleccione código o nombre, en la casilla a la derecha digite la información a buscar y presione .

Para la modificación o eliminación de un producto se realizara lo siguiente:

Búsqueda de producto para realizar la actualización de datos o eliminación, una vez encontrado seleccionar en la parte derecha de la lista la siguiente imagen,

 Esta opción cargara toda la información de la tabla del producto a las casillas de la sección 3 donde podremos modificar la información y guardarla.

 Esta opción permitirá la eliminación del producto seleccionado sin opción a poder recuperar la información.

3.- Muestra la lista de producto, y la búsqueda realizada en el punto número 2.

Nuevo cliente

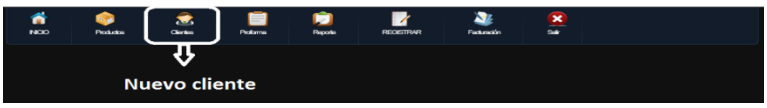


Figura 82 *Clientes*

Muestra gráficamente la interfaz para el ingreso de nuevos clientes.

Pantalla principal para ingresar nuevos clientes.

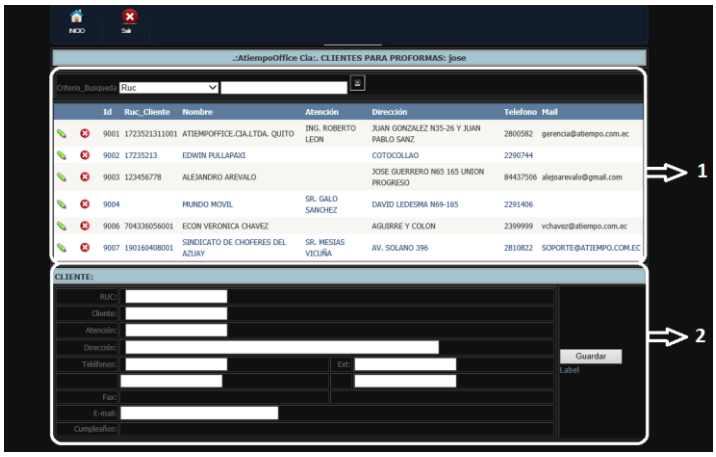


Figura 83 *Interfaz Cliente*

Muestra gráficamente la interfaz de cliente.

1.-En criterios de búsqueda seleccione ruc o nombre, en la casilla a la derecha digite la información a buscar y presione .

A continuación aparecerá un listado de clientes encontrados.

Para la modificación o eliminación de un cliente se realizara lo siguiente:

Búsqueda de cliente para realizar la actualización de datos o eliminación, una vez encontrado seleccionar en la parte derecha de la lista la siguiente imagen,

 Esta opción cargara toda la información de la tabla del cliente a las casillas de la sección 1 donde podremos modificar la información y guardarla.

 Esta opción permitirá la eliminación del cliente seleccionado sin opción a poder recuperar la información.

2.- Ingrese información de cliente a ingresar como, Ruc, cliente, atención, dirección, teléfonos, ext, email, una vez ingresada la información presionar el botón de guardar donde el software validara la información ingresada y guardara si todo es correcto de lo contrario mostrara un mensaje de error al guardar información.

Proforma

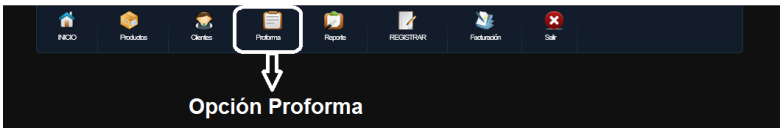


Figura 84 *Interfaz proforma*

Muestra gráficamente la interfaz para el ingreso a la ventana proforma

Pantalla principal de proformas

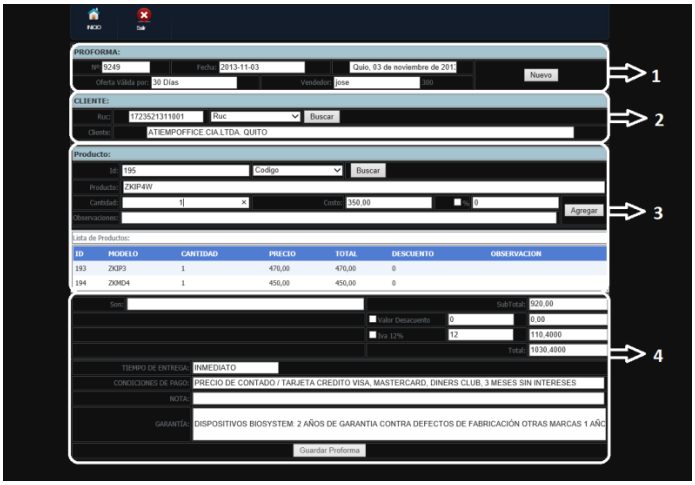


Figura 85 *Interfaz Proforma*

Muestra gráficamente la interfaz para generar una proforma.

1.- Presionar el botón nuevo para generar una nueva proforma y poder habilitar las casillas para poder ingresar información.

No se puede generar una proforma de un día pasado.

2.- En esta opción buscaremos el cliente para poder realizar la proforma, para esto seleccionaremos en criterio de búsqueda si desea buscar por nombre o por ruc, una vez seleccionado digitar en la casilla derecha el cliente a buscar y presionar .

Si la información ingresada del cliente es correcta se habilitara la opción para poder realizar la búsqueda de productos, de lo contrario nos mostrar un mensaje de error, en la parte derecha alado del botón de buscar.

3.- En esta opción buscaremos el producto para poder realizar la proforma, para esto seleccionaremos en criterio de búsqueda si desea buscar por nombre o por código, una vez seleccionado digitar en la casilla derecha la información del producto a buscar y presionar .

Si la información ingresada del producto es correcta se habilitara las demás casillas para ingresar el detalle del producto y el botón agregar con el cual podremos agragar varios productos para una proforma y un cliente de lo contrario mostrara un mensaje de error, en la parte derecha alado del botón de buscar.

4.- En esta sección mostrara la información de la proforma como: subtotal, descuento, iva, total, tiempo de entrega, características, nota y garantía, estos datos son solo informativos no se pueden modificar.

Registro de nuevo usuario



Figura 86 *Interfaz Usuario*

Muestra gráficamente la interfaz para el acceso a la ventana de usuario.

Pantalla principal de registro de usuarios

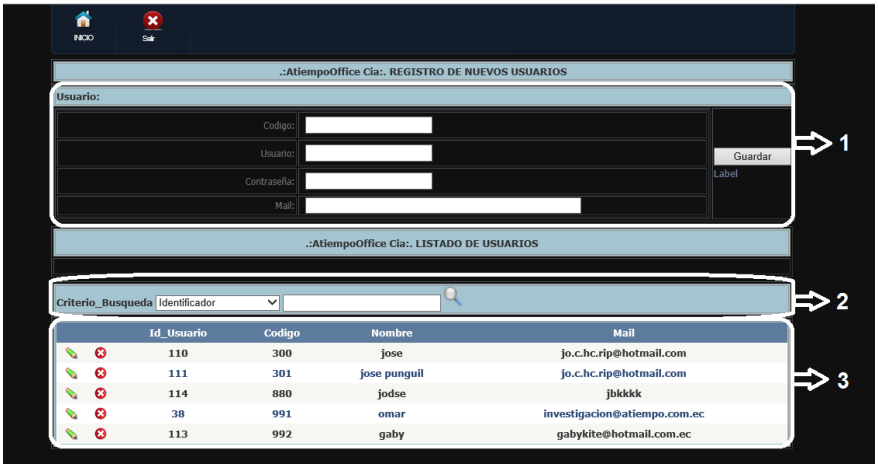


Figura 87 *Interfaz nuevo usuario*

Muestra gráficamente la interfaz ingreso, modificación y eliminación de usuarios

1.- En esta sección nos muestra la opción de ingresar la información del nuevo usuario como Codigo del usuario, nombre de usuario, contraseña y un mail, una vez ingresada la información presionar el botón de guardar donde el software validara la información ingresada y guardara si todo es correcto de lo contrario mostrara un mensaje de error en la parte inferior del botón al guardar información.

2.- En esta sección podemos elegir los criterios de búsqueda donde podremos seleccionar identificador o nombre, en la casilla a la derecha digite la información a buscar y presione .

3.- Muestra la lista delos usuarios, y la búsqueda realizada en el punto número 2.

Para la modificación o eliminación de un usuario registrado en el sistema se realizara lo siguiente:

Búsqueda de usuario por los diferentes criterios de búsqueda, para realizar la actualización de datos o eliminación, una vez encontrado el usuario seleccionar en la parte derecha de la lista la siguiente imagen,

 Esta opción cargara toda la información de la tabla del usuario a las casillas de la sección 1 donde podremos modificar la información y guardarla.

Esta opción permitirá la eliminación del usuario seleccionado sin opción a poder recuperar la información.

Facturación

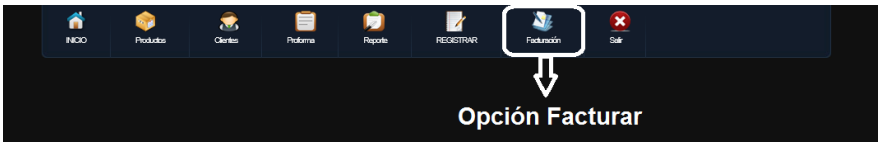


Figura 88 *Interfaz Facturación*

Muestra gráficamente la interfaz de acceso a la ventana de facturación

Página principal de facturación

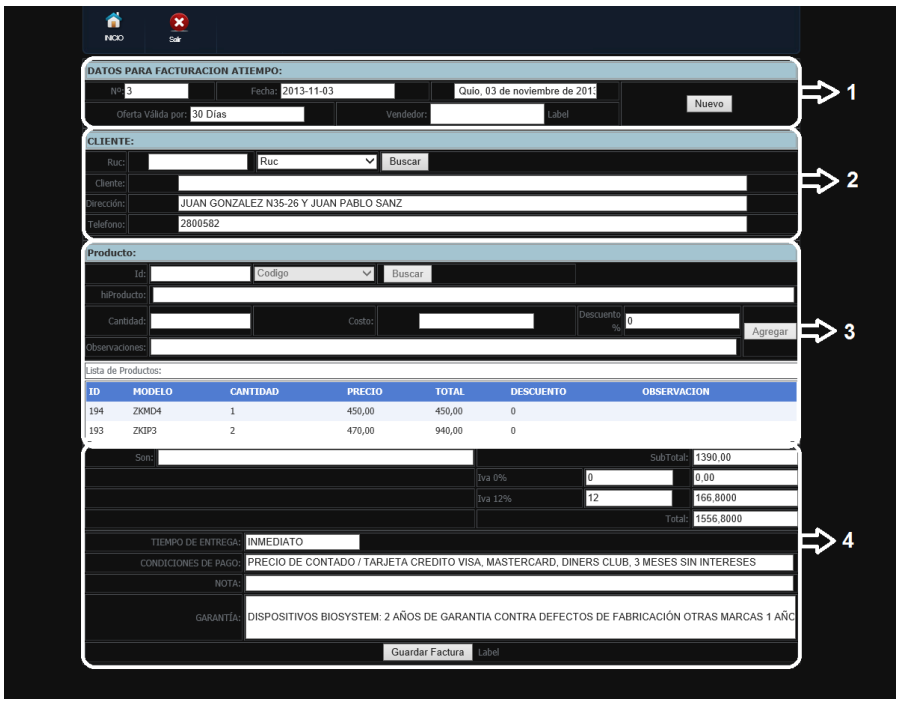


Figura 89 *Interfaz de facturación*

Muestra gráficamente la interfaz para generar una factura.

1.- Presionar el botón nuevo para generar una nueva factura y poder habilitar las casillas para poder ingresar información.

No se puede generar una factura de un día pasado.

2.- En esta opción buscaremos el cliente para poder realizar la factura, para esto seleccionaremos en criterio de búsqueda si desea buscar por nombre o por ruc, una vez seleccionado digitar en la casilla derecha la información del cliente a buscar y presionar .

Si la información ingresada del cliente es correcta se habilitara la opción para poder realizar la búsqueda de productos, de lo contrario nos mostrar un mensaje de error, en la parte derecha alado del botón de buscar.

3.- En esta opción buscaremos el producto para poder realizar la factura, para esto seleccionaremos en criterio de búsqueda si desea buscar por nombre o por código, una vez seleccionado digitar en la casilla derecha la información del producto a buscar y presionar .

Si la información ingresada del producto es correcta se habilitara las demás casillas para ingresar el detalle del producto y el botón agregar con el cual podremos agregar varios productos para una factura y un cliente de lo contrario mostrara un mensaje de error, en la parte derecha alado del botón de buscar.

4.- En esta sección mostrara la información de la factura como: subtotal, descuento, IVA, total y observación, estos datos son solo informativos no se pueden modificar a excepción del campo observación.

Nota: en todas las pantallas aparecen las siguientes imágenes:



.- Esto permitirá regresar al menú principal del sistema



.- Permite salir del sistema.

Anexo A 5 Bibliografía.

Configuración e instalación IIS

https://www.google.com.ec/search?q=modelo+fisico+uml&oq=modelo+fisico+uml&aqs=chrome..69i57.33534j0j7&sourceid=chrome&espv=210&es_sm=93&ie=UTF-8#es_sm=93&espv=210&q=ejemplo+de+configuracion+servidor+para+asp.net

http://msdn.microsoft.com/es-es/library/bb763174(v=vs.90).aspx

http://geeks.ms/blogs/rduarte/archive/2011/12/02/configurar-autenticaci-243-n-windows-en-iis-win-7-amp-server-2008.aspx

Controles Ajax

http://www.aspsnippets.com/Articles/ASPNet-AJAX-FilteredTextBoxExtender-Control-Example.aspx

Controles ASP

http://msdn.microsoft.com/es-es/library/ms228044%28es-es,VS.90%29.aspx

https://www.google.com.ec/search?q=calendar+asp.net&oq=calendar+asp.net&aqs=chrome..69i57j0l5.3388j0j7&sourceid=chrome&espv=210&es_sm=93&ie=UTF-8#es_sm=93&espv=210&q=drop+down+calendar+c%23+asp.net

Programación C#

https://www.google.com.ec/search?q=ejecutar+sstore+procedure+en+c%23&oq=ejecutar+sstore+procedure+en+c%23&aqs=chrome..69i57j0l5.12147j0j7&sourceid=chrome&espv=210&es_sm=93&ie=UTF-8#es_sm=93&espv=210&q=mes+actual+en+c%23

Store Procedure

http://joshuaolguin.wordpress.com/2012/02/28/como-ejecutar-un-procedimiento-almacenado-stored-procedure-con-visual-studio-2010-y-c/

Transact-SQL

http://technet.microsoft.com/es-es/library/ms186724.aspx