

# UNIVERSIDAD PÚBLICA DE EL ALTO

## CARRERA DE INGENIERÍA DE SISTEMAS



### PROYECTO DE GRADO

#### “SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO CONTABLE”

CASO: EMPRESA CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA  
“C.I.T.”

Para Optar al Título de Licenciatura en Ingeniería de Sistemas

**Mención: INFORMÁTICA Y COMUNICACIONES**

**Postulante:** David Condori Callata

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

**Tutor Especialista:** Lic. Wendy Sarmiento Martínez

**Tutor Revisor:** Lic. Beatriz Colque Condori

EL ALTO – BOLIVIA

2020

## **DEDICATORIA**

*A Dios, porque siempre ha estado a mi lado dándome fortaleza para seguir adelante día a día.*

*A mi madre, Inés Callata de Condori por su apoyo incondicional, comprensión y paciencia todo este tiempo.*

*A mi padre, Claudio Condori Laura por todo el apoyo y los ánimos brindados de seguir adelante y enseñanza.*

*A mis hermanas, Lidia Maribel, Miriam Yannet, Deysi Yezmin por el apoyo y ánimos que me brindaron.*

## AGRADECIMIENTOS

*Agradezco:*

*A dios, por la oportunidad de estar vivo, de cuidar de mi salud día a día y por cuidar de mi familia y a todas las personas que apreció mucho, también por permitirme llegar hasta este punto de mi vida permitiéndome poder culminar una de mis metas.*

*A mi familia por todo el apoyo brindado, su comprensión y sus palabras de aliento para poder seguir adelante y cumplir mis metas.*

*A mi tutora metodológico Ing. Maricel Yarari Mamani por su conocimiento brindado, apoyo, tiempo, persistencia, paciencia, confianza y todo el ánimo brindado a mi persona.*

*A mi tutora especialista Lic. Wendy Sarmiento Martínez por todo el apoyo brindado, compartiéndome su conocimiento, brindarme sus orientaciones, disponibilidad de tiempo, por sus observaciones y sugerencias brindadas con una acertada orientación durante el desarrollo del presente proyecto.*

*A mi tutora revisor Lic. Beatriz Colque Condori por los conocimientos brindados, disponibilidad de tiempo, su acertada orientación y observaciones brindadas en el presente proyecto.*

*A los docentes de la carrera de Ingeniería de Sistemas, por haberme impartido sus conocimientos y experiencias.*

*A la Universidad Pública de El Alto, por acogerme en sus aulas durante todo este transcurso de tiempo de estudio, así también a la carrera de Ingeniería de sistemas y a todos mis compañeros (as) y amigos (as) por su apoyo incondicional.*

*A la empresa Construcción innovación & tecnología por brindarme la oportunidad de aplicar mis conocimientos adquiridos, agradecer a la gerente Lic. Marycel J. Quisbert Torrez por el voto de confianza brindado.*

*Y finalmente gracias a todas las personas que me brindaron su ayuda directa o indirectamente en la elaboración del presente proyecto.*

*¡Muchas Gracias!*

## RESUMEN

El presente trabajo contiene el proceso de desarrollo del Sistema Web de Gestión Administrativa y Seguimiento Contable. El proyecto tiene como finalidad implementar un sistema de gestión administrativa y seguimiento contable permitiendo conocer la estabilidad de la empresa, en cuanto a ventas, costes y gastos en general y llevar un registro adecuado de la información contable, que contribuye a la toma de decisiones estratégicas en base a datos reales en cuanto a las inversiones o adquisiciones futuras son factibles para la empresa, también ayudando a calcular los balances mensuales y anual de la empresa.

Primeramente, en el marco preliminar se presenta la introducción con las descripciones de la empresa y referencias de trabajos similares que fueron desarrollados anteriormente, posteriormente se diagnostica la situación problemática, para luego extraer los problemas y objetivos propuestos para la solución de estos, así mismo se presenta la justificación, como los límites y alcances, metodología, herramientas de desarrollo y los aportes del proyecto.

Dentro del marco teórico, se describe la metodología Scrumban, lenguajes de modelado UML, herramientas, pruebas de software, métricas de calidad ISO/IEC 25010 y estimación de costos de software.

En el marco aplicativo, se aplica y desarrolla las fases de la metodología propuesta y los modelados de UML y las respectivas pruebas de software propuestas en el marco teórico.

Posteriormente, se aplica las métricas de calidad, las políticas de seguridad y estimaciones de costos del software.

Finalmente se presenta conclusiones y recomendaciones, las conclusiones se reflejan a los objetivos alcanzados en el presente proyecto y las recomendaciones son las ideas que se aportan para el mantenimiento del sistema y las futuras mejoras que se puede llevar a cabo.

## ÍNDICE

1. MARCO PRELIMINAR .....	1
1.1. Introducción.....	1
1.2. Antecedentes .....	2
1.2.1. Antecedentes de la empresa .....	2
1.2.2. Antecedentes de trabajos a fines .....	3
1.3. Planteamiento del problema .....	4
1.3.1. Problema principal .....	4
1.3.2. Problemas secundarios .....	4
1.4. Objetivos .....	5
1.4.1. Objetivo general.....	5
1.4.2. Objetivos específicos .....	5
1.5. Justificaciones .....	6
1.5.1. Justificación técnica .....	6
1.5.2. Justificación económica .....	7
1.5.3. Justificación social .....	7
1.6. Metodología .....	8
1.6.1. Metodología de desarrollo de software .....	8
1.7. Técnicas y Herramientas.....	8
1.7.1. Recolección de datos.....	8
1.7.2. Software de desarrollo .....	8
1.7.3. Hardware de desarrollo.....	9
1.7.4. Software de implementación.....	9
1.7.5. Hardware de implementación .....	9
1.8. Métrica de calidad .....	10
1.9. Estimación de costos de software .....	10

1.10.	Seguridad .....	10
1.10.1.	ISO/IEC 25010 .....	10
1.11.	Límites y alcances.....	11
1.11.1.	Limites .....	11
1.11.2.	Alcances .....	11
1.12.	Aportes .....	12
1.12.1.	Aporte Práctico .....	12
1.12.2.	Aparte Teórico .....	12
2.	MARCO TEÓRICO.....	
2.1.	Introducción .....	13
2.2.	Gestión administrativa.....	13
2.3.	Seguimiento contable.....	13
2.4.	Ingeniería de software.....	14
2.4.1.	Método de ciclo de vida .....	15
2.5.	Metodología de desarrollo Scrumban.....	15
2.5.1.	Scrumban .....	17
2.5.1.1.	Roles Scrumban .....	17
2.5.1.2.	Actividades en Scrumban .....	18
2.5.2.	Fases del modelo de desarrollo Scrumban .....	20
2.5.3.	Comparación de Scrumban.....	21
2.6.	Lenguaje de modelado unificado (UML) .....	22
2.6.1.	Vistas de UML.....	22
2.6.1.1.	Vista estática .....	24
2.6.1.2.	Vista de los casos de uso .....	24
2.6.1.3.	Vista de interacción.....	25

2.6.1.4. Vista de la máquina de estado .....	27
2.6.1.5. Vista de actividades .....	29
2.6.1.6. Vistas físicas .....	29
2.6.1.7. Vista de gestión del modelo .....	31
2.6.1.8. Construcciones de extensión .....	32
2.6.1.9. Conexión entre vistas .....	32
2.7. Herramientas .....	33
2.7.1. Framework Laravel .....	33
2.7.1.1. Cómo funciona Modelo-Vista-Controlador (MVC) .....	34
2.7.2. JetBrains PhpStorm .....	34
2.7.3. PHP 7 .....	35
2.7.4. Framework Bootstrap .....	35
2.7.5. Xampp .....	36
2.7.6. MySQL .....	36
2.7.7. Composer .....	36
2.8. Pruebas de software .....	36
2.8.1. Prueba de caja blanca .....	36
2.8.2. Prueba de caja negra .....	38
2.9. Métrica de calidad .....	38
2.9.1. ISO/IEC 25010 .....	39
2.9.1.1. Adecuación funcional .....	39
2.9.1.2. Eficiencia de desempeño .....	40
2.9.1.3. Compatibilidad .....	40
2.9.1.4. Usabilidad .....	41
2.9.1.5. Fiabilidad .....	41

2.9.1.6. Seguridad .....	42
2.9.1.7. Mantenibilidad.....	42
2.9.1.8. Portabilidad.....	43
2.10. Estimación de costos de software .....	44
2.10.1. Cocomo II .....	44
3. MARCO APLICATIVO .....	
3.1. Introducción .....	46
3.2. Arquitectura y requerimiento de software .....	46
3.2.1. Arquitectura .....	46
3.2.2 Requerimiento de software .....	47
3.3. Roles Scrumban.....	47
3.4. Fase 1 lista de requisitos.....	47
3.4.1. Requisitos de hardware y software de desarrollo .....	49
3.5. Fase 2 descripción de las tarjetas de tareas .....	50
3.5.1. Tarjetas de tareas .....	50
3.6. Fase 3 Análisis y diseño - Base de datos .....	55
3.6.1. Descripciones de los actores .....	55
3.6.2. Diagrama de caso de uso general .....	56
3.6.3. Diseño de base de datos e implementación .....	57
3.7. Fase 4 Programación – Base de datos .....	58
3.8. Fase 5 Implementación y pruebas – Base de datos .....	62
3.8.1. Resultados .....	63
3.9. Fase 3 Análisis y diseño - Módulo usuario .....	63
3.9.1. Caso de Uso .....	64
3.9.2. Diagrama de clases .....	65

3.9.3. Diagrama de actividades o de flujo .....	66
3.9.4. Diagrama de secuencia .....	67
3.10. Fase 4 Programación – Módulo usuario .....	67
3.11. Fase 5 Implementación y pruebas – Modulo usuario .....	68
3.11.1. Prueba de Caja Blanca .....	69
3.11.2. Prueba de Caja Negra .....	70
3.11.3. Resultados .....	71
3.12. Fase 3 Análisis y diseño - Modulo plan de cuentas .....	71
3.12.1. Caso de Uso .....	71
3.12.2. Diagrama de clases .....	72
3.12.3. Diagrama de actividades o de flujo .....	73
3.12.4. Diagrama de secuencia .....	74
3.13. Fase 4 Programación – Modulo plan de cuentas.....	74
3.14. Fase 5 Implementación y pruebas – Modulo plan de cuentas .....	75
3.14.1. Prueba de Caja Blanca .....	76
3.14.2. Prueba de Caja Negra .....	77
3.14.3. Resultados .....	77
3.15. Fase 3 Análisis y diseño - Módulo de libro diario .....	78
3.15.1. Caso de Uso .....	78
3.15.2. Diagrama de clases .....	79
3.15.3. Diagrama de actividades o de flujo .....	80
3.15.4 Diagrama de secuencia .....	81
3.16. Fase 4 Programación – Módulo de libro diario .....	81
3.17. Fase 5 Implementación y pruebas – Módulo de libro diario .....	82
3.17.1. Prueba de Caja Blanca .....	83

3.17.2. Prueba de Caja Negra .....	84
3.17.3. Resultados .....	84
3.18. Fase 3 Análisis y diseño – Modulo Libro Mayor.....	85
3.18.1. Caso de Uso .....	85
3.18.2. Diagrama de clases .....	86
3.18.3. Diagrama de actividades o de flujo .....	87
3.18.4 Diagrama de secuencia .....	88
3.19. Fase 4 Programación – Modulo Libro Mayor .....	88
3.20. Fase 5 Implementación y pruebas – Modulo Libro Mayor .....	89
3.20.1. Prueba de Caja Blanca .....	89
3.20.2. Prueba de Caja Negra .....	91
3.20.3. Resultados .....	91
3.21. Fase 3 Análisis y diseño - Módulo de Balance General.....	92
3.21.1. Caso de Uso .....	92
3.21.2. Diagrama de clases .....	93
3.21.3. Diagrama de actividades o de flujo .....	94
3.21.4 Diagrama de secuencia .....	95
3.22. Fase 4 Programación – Módulo de Balance General .....	95
3.23. Fase 5 Implementación y pruebas – Módulo de Balance General.....	96
3.23.1. Prueba de Caja Blanca .....	97
3.23.2. Prueba de Caja Negra .....	98
3.23.3. Resultados .....	98
3.24. Fase 3 Análisis y diseño - Módulo de Productos .....	99
3.24.1. Caso de Uso .....	99
3.24.2. Diagrama de clases .....	100

3.24.3. Diagrama de actividades o de flujo .....	101
3.24.4 Diagrama de secuencia .....	102
3.25. Fase 4 Programación – Módulo de Productos.....	102
3.26. Fase 5 Implementación y pruebas – Módulo de Productos .....	103
3.26.1. Prueba de Caja Blanca .....	104
3.26.2. Prueba de Caja Negra .....	105
3.26.3. Resultados .....	105
3.27. Fase 3 Análisis y diseño - Módulo de categorías.....	106
3.27.1. Caso de Uso .....	106
3.27.2. Diagrama de clases .....	107
3.27.3. Diagrama de actividades o de flujo .....	108
3.27.4 Diagrama de secuencia .....	109
3.28. Fase 4 Programación – Módulo de categorías .....	109
3.29. Fase 5 Implementación y pruebas – Módulo de categorías.....	110
3.29.1. Prueba de Caja Blanca .....	111
3.29.2. Prueba de Caja Negra .....	112
3.29.3. Resultados .....	112
3.30. Fase 3 Análisis y diseño - Módulo de Proveedor.....	113
3.30.1. Caso de Uso .....	113
3.30.2. Diagrama de clases .....	114
3.30.3. Diagrama de actividades o de flujo .....	115
3.30.4 Diagrama de secuencia .....	116
3.31. Fase 4 Programación – Módulo de Proveedor .....	116
3.32. Fase 5 Implementación y pruebas – Módulo de Proveedor .....	117
3.32.1. Prueba de Caja Blanca .....	118

3.32.2. Prueba de Caja Negra .....	119
3.32.3. Resultados .....	119
3.33. Fase 3 Análisis y diseño - Módulo de Clientes .....	120
3.33.1. Caso de Uso .....	120
3.33.2. Diagrama de clases .....	121
3.33.3. Diagrama de actividades o de flujo .....	122
3.33.4 Diagrama de secuencia .....	123
3.34. Fase 4 Programación – Módulo de Clientes .....	123
3.35. Fase 5 Implementación y pruebas – Módulo de Clientes .....	124
3.35.1. Prueba de Caja Blanca .....	125
3.35.2. Prueba de Caja Negra .....	126
3.35.3. Resultados .....	126
3.36. Fase 3 Análisis y diseño - Módulo de Compras.....	127
3.36.1. Caso de Uso .....	127
3.36.2. Diagrama de clases .....	128
3.36.3. Diagrama de actividades o de flujo .....	129
3.36.4 Diagrama de secuencia .....	130
3.37. Fase 4 Programación – Módulo de Compras .....	130
3.38. Fase 5 Implementación y pruebas – Módulo de Compras .....	131
3.38.1. Prueba de Caja Blanca .....	132
3.38.2. Prueba de Caja Negra .....	133
3.38.3. Resultados .....	133
3.39. Fase 3 Análisis y diseño - Módulo de Ventas .....	134
3.39.1. Caso de Uso .....	134
3.39.2. Diagrama de clases .....	135

3.39.3. Diagrama de actividades o de flujo .....	136
3.39.4 Diagrama de secuencia .....	137
3.40. Fase 4 Programación – Módulo de Ventas.....	137
3.41. Fase 5 Implementación y pruebas – Módulo de Ventas .....	138
3.41.1. Prueba de Caja Blanca .....	139
3.41.2. Prueba de Caja Negra .....	140
3.41.3. Resultados .....	140
3.42. Fase 3 Análisis y diseño - Generar Facturas o Recibos .....	141
3.42.1. Caso de Uso .....	141
3.42.2. Diagrama de clases .....	142
3.42.3. Diagrama de actividades o de flujo .....	143
3.42.4 Diagrama de secuencia .....	144
3.43. Fase 4 Programación – Generar Facturas o Recibos.....	144
3.44. Fase 5 Implementación y pruebas – Generar Facturas o Recibos .....	146
3.44.1. Prueba de Caja Blanca .....	146
3.44.2. Prueba de Caja Negra .....	147
3.44.3. Resultados .....	148
3.45. Fase 3 Análisis y diseño - Módulo de Archivos .....	148
3.45.1. Caso de Uso .....	149
3.45.2. Diagrama de clases .....	150
3.45.3. Diagrama de actividades o de flujo .....	151
3.45.4 Diagrama de secuencia .....	151
3.46. Fase 4 Programación – Módulo de Archivos .....	152
3.47. Fase 5 Implementación y pruebas – Módulo de Archivos.....	153
3.47.1. Prueba de Caja Blanca .....	153

3.47.2. Prueba de Caja Negra .....	154
3.47.3. Resultados .....	155
4. MÉTRICAS DE CALIDAD, SEGURIDAD Y ESTIMACIÓN DE COSTOS .....	
4.1. Norma ISO/IEC 25010 .....	156
4.1.1. Adecuación funcional .....	156
4.1.2. Usabilidad .....	160
4.1.3. Fiabilidad .....	162
4.1.4. Mantenibilidad.....	164
4.1.5. Portabilidad.....	165
4.1.6. Calidad General .....	166
4.2. Seguridad.....	167
4.2.1. Autenticación.....	167
4.2.2. Asignación de roles.....	168
4.2.3. Encriptación .....	169
4.2.4. Seguridad en la base de datos.....	170
4.3. Estimación de costos del software .....	171
4.3.1. Cocomo II .....	171
5. CONCLUSIONES Y RECOMENDACIONES.....	
5.1. Conclusiones .....	174
5.2. Recomendaciones .....	176
BIBLIOGRAFÍA.....	177
ANEXOS.....	

## ÍNDICE DE FIGURAS

Figura N.º 1. 1 Organigrama Construcción Innovación & Tecnología “C.I.T.” .....	2
Figura N.º 2. 1 Método de ciclo de vida iterativo .....	15
Figura N.º 2. 2 Descripción general de Scrumban.....	17
Figura N.º 2. 3 Flujo de Trabajo de Scrumban .....	18
Figura N.º 2. 4 Descripción de la Tarjeta de Scrumban .....	19
Figura N.º 2. 5 Modelo de proceso de desarrollo del proyecto .....	20
Figura N.º 2. 6 Diagrama de Clases .....	24
Figura N.º 2. 7 Diagrama de casos de uso .....	25
Figura N.º 2. 8 Diagrama de secuencias .....	26
Figura N.º 2. 9 Diagrama de Colaboración .....	27
Figura N.º 2. 10 Diagrama de Estado.....	28
Figura N.º 2. 11 Diagrama de Estado.....	28
Figura N.º 2. 12 Diagrama de componentes.....	30
Figura N.º 2. 13 Diagrama de despliegue (nivel de descriptor).....	31
Figura N.º 2. 14 Paquetes .....	32
Figura N.º 2. 15 Notación gráfico de flujos .....	37
Figura N.º 2. 16 Organigrama de la norma ISO 25010 .....	39
Figura N.º 3. 1 Arquitectura del software .....	46
Figura N.º 3. 2 Tarjetas de Tareas .....	51
Figura N.º 3. 3 Diagrama de Caso de Uso General.....	56
Figura N.º 3. 4 Modelo relacional de la Base de Datos .....	58
Figura N.º 3. 5 Prueba e implementación de Base de datos en el servidor .....	62
Figura N.º 3. 6 Caso de uso del Módulo Usuario en la seguridad.....	64
Figura N.º 3. 7 Diagrama de clases Modelo Usuario y acceso de seguridad.....	66
Figura N.º 3. 8 Diagrama de actividades y módulo de usuario y seguridad .....	66
Figura N.º 3. 9 Diagrama de secuencia del módulo de usuario y seguridad, registro	67
Figura N.º 3. 10 Resultados Modulo usuario y seguridad, acciones de registros.....	67
Figura N.º 3. 11 Grafo de flujo módulo de usuario y seguridad .....	69
Figura N.º 3. 12 Caso de uso del Módulo plan de cuentas .....	72

Figura N.º 3. 13 Diagrama de clases del Módulo de plan de cuentas .....	73
Figura N.º 3. 14 Diagrama de actividades o flujo modelo plan de cuentas .....	73
Figura N.º 3. 15 Diagrama de secuencia Modulo de registro plan de cuentas .....	74
Figura N.º 3. 16 Entornos gráficos del Módulo de plan de cuentas/Registros .....	75
Figura N.º 3. 17 Grafo de flujo módulo Plan de cuentas .....	76
Figura N.º 3. 18 Caso de uso del Módulo libro diario.....	79
Figura N.º 3. 19 Diagrama de clases del Módulo de libro diario .....	80
Figura N.º 3. 20 Diagrama de actividades o flujo modelo plan de cuentas .....	80
Figura N.º 3. 21 Diagrama de secuencia Modulo de registro libro diario .....	81
Figura N.º 3. 22 Entornos gráficos del Módulo de libro diario .....	82
Figura N.º 3. 23 Grafo de flujo módulo Libro diario.....	83
Figura N.º 3. 24 Caso de uso del Módulo Libro mayor .....	86
Figura N.º 3. 25 Diagrama de clases del Módulo de Libro mayor .....	87
Figura N.º 3. 26 Diagrama de actividades o flujo modelo libro mayor.....	87
Figura N.º 3. 27 Diagrama de secuencia Modulo de libro mayor .....	88
Figura N.º 3. 28 Entornos gráficos del Módulo de Libro Mayor.....	89
Figura N.º 3. 29 Grafo de flujo módulo libro mayor.....	90
Figura N.º 3. 30 Diagrama de clases del Módulo de balance general .....	94
Figura N.º 3. 31 Diagrama de actividades o flujo modelo balance general .....	94
Figura N.º 3. 32 Diagrama de secuencia Modulo de balance general .....	95
Figura N.º 3. 33 Entornos gráficos del Módulo de Balance general.....	96
Figura N.º 3. 34 Grafo de flujo módulo balance general .....	97
Figura N.º 3. 35 Caso de uso del Módulo productos .....	100
Figura N.º 3. 36 Diagrama de clases del Módulo de productos .....	101
Figura N.º 3. 37 Diagrama de actividades o flujo modulo productos.....	101
Figura N.º 3. 38 Diagrama de secuencia Modulo de productos.....	102
Figura N.º 3. 39 Entornos gráficos del Módulo de productos/Registros .....	103
Figura N.º 3. 40 Grafo de flujo módulo de productos.....	104
Figura N.º 3. 41 Caso de uso del Módulo de categorías.....	107
Figura N.º 3. 42 Diagrama de clases del Módulo de categorías .....	108
Figura N.º 3. 43 Diagrama de actividades o flujo módulo de categorías.....	108

Figura N.º 3. 44 Diagrama de secuencia Modulo de registro categorías .....	109
Figura N.º 3. 45 Entornos gráficos del Módulo de Categorías /Registros .....	110
Figura N.º 3. 46 Grafo de flujo módulo de categorías .....	111
Figura N.º 3. 47 Caso de uso del Módulo de proveedor .....	114
Figura N.º 3. 48 Diagrama de clases del Módulo de proveedor .....	115
Figura N.º 3. 49 Diagrama de actividades o flujo módulo de proveedores.....	115
Figura N.º 3. 50 Diagrama de secuencia Modulo de proveedor.....	116
Figura N.º 3. 51 Entornos gráficos del Módulo de proveedor/Registros.....	117
Figura N.º 3. 52 Grafo de flujo módulo de proveedor .....	118
Figura N.º 3. 53 Caso de uso del Módulo cliente.....	121
Figura N.º 3. 54 Diagrama de clases del Módulo de cliente .....	122
Figura N.º 3. 55 Diagrama de actividades o flujo modulo cliente .....	122
Figura N.º 3. 56 Diagrama de secuencia Modulo de registro cliente .....	123
Figura N.º 3. 57 Entornos gráficos del Módulo de clientes/Registros .....	124
Figura N.º 3. 58 Grafo de flujo módulo cliente .....	125
Figura N.º 3. 59 Caso de uso del Módulo de compras .....	128
Figura N.º 3. 60 Diagrama de clases del Módulo de compras .....	129
Figura N.º 3. 61 Diagrama de actividades o flujo módulo de compras .....	129
Figura N.º 3. 62 Diagrama de secuencia módulo de compras .....	130
Figura N.º 3. 63 Entornos gráficos del Módulo de compras.....	131
Figura N.º 3. 64 Grafo de flujo módulo de compras.....	132
Figura N.º 3. 65 Caso de uso del Módulo de ventas.....	135
Figura N.º 3. 66 Diagrama de clases del Módulo de ventas .....	136
Figura N.º 3. 67 Diagrama de actividades o flujo módulo de ventas .....	136
Figura N.º 3. 68 Diagrama de secuencia Modulo de registro ventas .....	137
Figura N.º 3. 69 Entornos gráficos del Módulo de ventas.....	138
Figura N.º 3. 70 Grafo de flujo módulo de ventas .....	139
Figura N.º 3. 71 Caso de uso Generar Facturas o Recibos.....	142
Figura N.º 3. 72 Diagrama de clases de facturas o recibos .....	143
Figura N.º 3. 73 Diagrama de actividades o flujo de facturas o recibos .....	143
Figura N.º 3. 74 Diagrama de secuencia para generar facturas o recibos .....	144

Figura N.º 3. 75 Entornos gráficos de facturas o recibos.....	145
Figura N.º 3. 76 Grafo de flujo de facturas o recibos .....	146
Figura N.º 3. 77 Caso de uso del Módulo de archivos .....	149
Figura N.º 3. 78 Diagrama de clases del Módulo archivos .....	150
Figura N.º 3. 79 Diagrama de actividades o flujo modelo Archivos.....	151
Figura N.º 3. 80 Diagrama de secuencia Modulo de archivos .....	152
Figura N.º 3. 81 Entornos gráficos del Módulo de archivos .....	152
Figura N.º 3. 82 Grafo de flujo módulo de archivos .....	153
Figura N.º 4. 1 Código de validación de autenticación.....	168
Figura N.º 4. 2 Roles de asignación y Acciones .....	169
Figura N.º 4. 3 Encriptación Laravel.....	170
Figura N.º 4. 4 Tipo de Consultas evitando las inyecciones SQL .....	171
Figura N.º 4. 5 Inserción de datos para COCOMO II.....	172

## ÍNDICE DE TABLAS

Tabla N.º 1. 1 Características de pc y laptop.....	6
Tabla N.º 1. 2 Características del equipo de desarrollo .....	9
Tabla N.º 1. 3 Características de dispositivos de acceso .....	9
Tabla N.º 2. 1 Comparación de Scrum, Kanban y Scrumban .....	22
Tabla N.º 2. 2 Muestra de las Vistas de UML .....	23
Tabla N.º 2. 3 Algunas Relaciones entre Elementos de Diferentes Vistas.....	33
Tabla N.º 2. 4 Constante de Coste Modelo Básico .....	45
Tabla N.º 3. 1 Roles del equipo de trabajo .....	47
Tabla N.º 3. 2 Lista de Requisitos del proyecto .....	48
Tabla N.º 3. 3 Requisitos de Software y Hardware mínimo para el Desarrollo .....	50
Tabla N.º 3. 4 Descripción de los actores del sistema .....	55
Tabla N.º 3. 5 Modelo relacional de la Base de Datos.....	57
Tabla N.º 3. 6 Diccionario de datos de la Base de Datos .....	59
Tabla N.º 3. 7 Resultados de la implementación de la Base de Datos .....	63
Tabla N.º 3. 8 Módulo de Usuario, seguridad y las acciones .....	63
Tabla N.º 3. 9 Descripción de caso de uso del Módulo Usuario en la seguridad .....	64
Tabla N.º 3. 10 Inicio de sesión ingresa al módulo de usuario .....	65
Tabla N.º 3. 11 Usuario asignar roles, registro de usuarios.....	65
Tabla N.º 3. 12 Cálculo de complejidad.....	69
Tabla N.º 3. 13 Caminos independientes .....	70
Tabla N.º 3. 14 Prueba de Caja negra del módulo usuario, seguridad y acciones.....	70
Tabla N.º 3. 15 Módulo de plan de cuentas.....	71
Tabla N.º 3. 16 Descripción de caso de uso del Módulo plan de cuentas.....	71
Tabla N.º 3. 17 Inicio de sesión, ingresar al módulo de plan de cuentas .....	72
Tabla N.º 3. 18 Calculo de complejidad.....	76
Tabla N.º 3. 19 Caminos independientes .....	77
Tabla N.º 3. 20 Resultados de la implementación del módulo plan de cuentas .....	77
Tabla N.º 3. 21 Módulo Libro Diario .....	78
Tabla N.º 3. 22 Descripción de caso de uso del Módulo libro diario .....	78
Tabla N.º 3. 23 Inicio de sesión, ingresar al módulo de libro diario.....	79

Tabla N.º 3. 24 Cálculo de complejidad.....	83
Tabla N.º 3. 25 Caminos independientes .....	84
Tabla N.º 3. 26 Resultados de la implementación del módulo libro diario.....	84
Tabla N.º 3. 27 Módulo Libro Mayor .....	85
Tabla N.º 3. 28 Descripción de caso de uso del Módulo Libro Mayor .....	85
Tabla N.º 3. 29 Inicio de sesión, ingresar al módulo de libro mayor .....	86
Tabla N.º 3. 30 Cálculo de complejidad.....	90
Tabla N.º 3. 31 Caminos independientes .....	91
Tabla N.º 3. 32 Resultados de la implementación del módulo libro mayor .....	91
Tabla N.º 3. 33 Módulo Balance general .....	92
Tabla N.º 3. 34 Descripción de caso de uso del Módulo balance general .....	92
Tabla N.º 3. 35 Caso de uso del Módulo balance general .....	93
Tabla N.º 3. 36 Inicio de sesión, ingresar al módulo balance general.....	93
Tabla N.º 3. 37 Cálculo de complejidad.....	97
Tabla N.º 3. 38 Caminos independientes .....	98
Tabla N.º 3. 39 Resultados de la implementación del módulo Balance general .....	98
Tabla N.º 3. 40 Módulo Productos.....	99
Tabla N.º 3. 41 Descripción de caso de uso del Módulo productos .....	99
Tabla N.º 3. 42 Inicio de sesión, ingresar al módulo de productos .....	100
Tabla N.º 3. 43 Cálculo de complejidad.....	104
Tabla N.º 3. 44 Caminos independientes .....	105
Tabla N.º 3. 45 Resultados de la implementación del módulo plan de cuentas .....	105
Tabla N.º 3. 46 Módulo categorías .....	106
Tabla N.º 3. 47 Descripción de caso de uso del Módulo de categorías .....	106
Tabla N.º 3. 48 Inicio de sesión, ingresar al módulo de categorías .....	107
Tabla N.º 3. 49 Cálculo de complejidad.....	111
Tabla N.º 3. 50 Caminos independientes .....	112
Tabla N.º 3. 51 Resultados de la implementación del módulo de categorías.....	112
Tabla N.º 3. 52 Módulo de Proveedor .....	113
Tabla N.º 3. 53 Descripción de caso de uso del Módulo proveedor.....	113
Tabla N.º 3. 54 Inicio de sesión, ingresar al módulo de proveedor .....	114

Tabla N.º 3. 55 Cálculo de complejidad.....	118
Tabla N.º 3. 56 Caminos independientes .....	119
Tabla N.º 3. 57 Resultados de la implementación del módulo proveedor .....	119
Tabla N.º 3. 58 Módulo Cliente.....	120
Tabla N.º 3. 59 Descripción de caso de uso del Módulo cliente .....	120
Tabla N.º 3. 60 Inicio de sesión, ingresar al módulo cliente.....	121
Tabla N.º 3. 61 Cálculo de complejidad.....	125
Tabla N.º 3. 62 Caminos independientes .....	126
Tabla N.º 3. 63 Resultados de la implementación del módulo cliente .....	126
Tabla N.º 3. 64 Módulo de Compras .....	127
Tabla N.º 3. 65 Descripción de caso de uso del Módulo de compras .....	127
Tabla N.º 3. 66 Inicio de sesión, ingresar al módulo de compras .....	128
Tabla N.º 3. 67 Cálculo de complejidad.....	132
Tabla N.º 3. 68 Caminos independientes .....	133
Tabla N.º 3. 69 Resultados de la implementación del módulo de compras .....	133
Tabla N.º 3. 70 Módulo de Ventas.....	134
Tabla N.º 3. 71 Descripción de caso de uso del Módulo de ventas .....	134
Tabla N.º 3. 72 Inicio de sesión, ingresar al módulo de ventas .....	135
Tabla N.º 3. 73 Cálculo de complejidad.....	139
Tabla N.º 3. 74 Caminos independientes .....	140
Tabla N.º 3. 75 Resultados de la implementación del módulo de ventas.....	140
Tabla N.º 3. 76 Generar facturas o Recibos .....	141
Tabla N.º 3. 77 Descripción de caso de uso Facturas o Recibos .....	141
Tabla N.º 3. 78 Usuario Genera Facturas y Recibos .....	142
Tabla N.º 3. 79 Cálculo de complejidad.....	147
Tabla N.º 3. 80 Caminos independientes .....	147
Tabla N.º 3. 81 Resultados de la implementación de facturas o recibos .....	148
Tabla N.º 3. 82 Módulo de archivos .....	148
Tabla N.º 3. 83 Descripción de caso de uso del Módulo archivos .....	149
Tabla N.º 3. 84 Inicio de sesión, Modulo de archivos .....	150
Tabla N.º 3. 85 Cálculo de complejidad.....	154

Tabla N.º 3. 86 Caminos independientes .....	154
Tabla N.º 3. 87 Resultados de la implementación del módulo plan de cuentas .....	155
Tabla N.º 4. 1 Parámetros de medida y su cantidad.....	157
Tabla N.º 4. 2 Parámetros de medida y su cantidad.....	158
Tabla N.º 4. 3 Valores de ajuste de complejidad .....	158
Tabla N.º 4. 4 Encuesta sobre Usabilidad del Sistema.....	161
Tabla N.º 4. 5 Valores de fiabilidad de cada modulo .....	162
Tabla N.º 4. 6 Información requerida por el IMS.....	165
Tabla N.º 4. 7 Información de Calidad Global.....	167
Tabla N.º 4. 8 Líneas de código total.....	171
Tabla N.º 4. 9 Resultados de COCOMO II .....	173

# **1. MARCO PRELIMINAR**

## **1.1. Introducción**

Las Tecnologías de la Información están transformando las actividades económicas y cotidianas de las empresas, actualmente es uno de los fenómenos sociológicos más importantes del siglo, han invadido en cada uno de los campos de la actividad humana como ser en la ciencia, arte, educación, recreación, administración, economía. Estas tecnologías permiten mejorar la productividad tanto en instituciones públicas como privadas, organizando de una manera más eficiente su rutina de trabajo, siendo un apoyo en sus procesos administrativos y en la toma de decisiones.

Con la aparición del internet crece la necesidad en las empresas de contar con páginas web, para poder dar a conocer los servicios que ofrecen, de esta forma llegar a un mercado más amplio de consumidores.

Los sistemas contables o software contable comprenden los programas que ayudan a los empresarios a llevar las cuentas de su negocio al día, a esto se añade la innovación tecnológica que hace posible utilizar aplicaciones en línea sin tener que instalarlas en las computadoras.

Así la empresa Construcción Innovación & Tecnología “C.I.T.” entiende los beneficios que proporcionaría la implementación de un sistema que apoye a las necesidades en la empresa, al mismo tiempo que brinde un adecuado seguimiento contable, resguardo de información y ofrecer respuesta en tiempo real.

De esta forma se propone la implementación del Sistema Web de Gestión Administrativa y Seguimiento Contable, para un manejo óptimo, claro y preciso, de esta manera el sistema permitirá tener un registro eficiente del manejo de los contratos de obras como también de la parte contable de forma rápida utilizando herramientas tecnológicas y permita integrar la información con otras áreas de la organización.

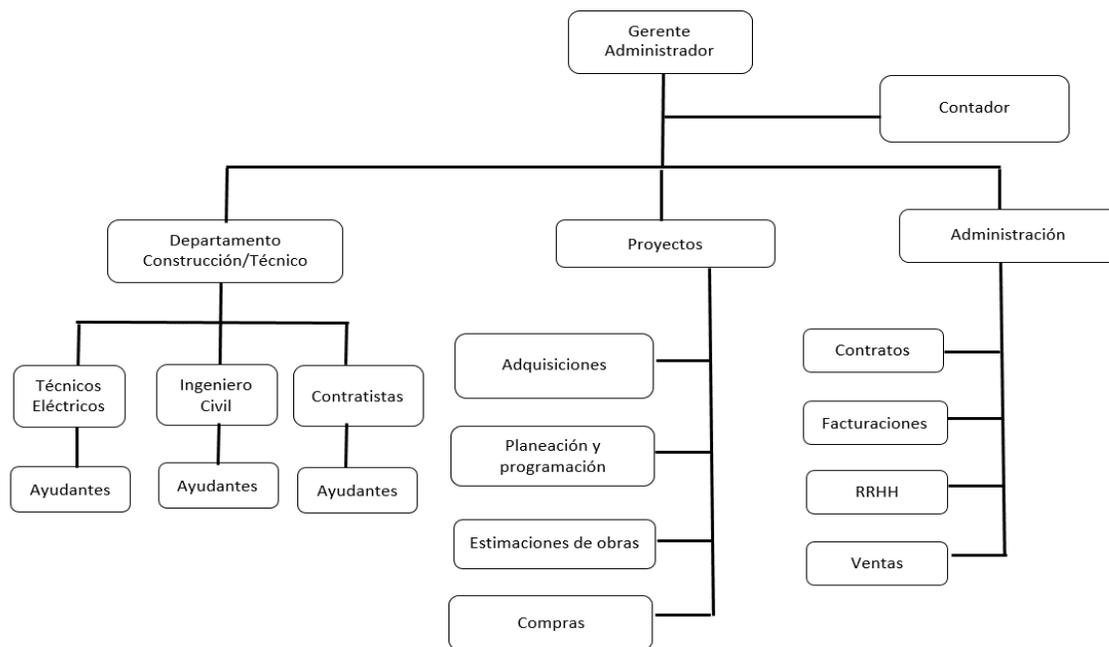
## 1.2. Antecedentes

### 1.2.1. Antecedentes de la empresa

La empresa Construcción Innovación & Tecnología “C.I.T.” inicia sus actividades como una empresa privada a un principio se destaca como una pequeña consultora online la cual luego empieza a buscar proyectos como licitaciones y obras en la parte de construcción y así fue como surge la empresa en diciembre de 2017 en la ciudad de La Paz – Av. San José Nro. 2217, esquina Av. Florida Zona Munaypata. El 30 de julio de 2018 es fundada la empresa. Es una entidad que tiene como objetivo social principal ofrecer servicios de construcciones civiles, instalaciones y reparaciones, así como instalaciones eléctricas, mantenimientos y también la venta de productos por mayor y menor para instituciones según sus clientes requieran, también se presenta a convocatorias de licitaciones las cuales son proyectos a los cuales que acuden.

- **Organigrama de la empresa C.I.T.**

Figura N.º 1. 1 Organigrama Construcción Innovación & Tecnología “C.I.T.”



Fuente:(Construcción Innovación & Tecnología “C.I.T.”)

- **Misión:** Nuestra misión es ofrecer servicios especiales de construcciones civiles, instalación, mantenimiento y ofrecer diversos productos a nuestros clientes.
- **Visión:** Ser una empresa de servicios con reconocimiento y reputación, con capacidad para acceder a la más novedosa tecnología disponible en el mercado para el cumplimiento de los compromisos alcanzados.

### 1.2.2. Antecedentes de trabajos a fines

- (Jhonathan Alfonso Fernández, Miguel Ángel Díaz Guzmán,2008) “**Sistema de control de inventarios y facturación de la licorería tropical club de Girardot**”, Para el desarrollo del sistema se utilizó la metodología del Proceso Unificado De Racional (RUP), tomando en cuenta todas las fases, artefactos e hitos definidos en la misma aplicación para el modelado del sistema el lenguaje de modelado unificado (UML), se utilizó el lenguaje de programación PHP, la base datos MySql y servidor Apache, que se adecuan a las necesidades e implementación de la facultad.
- (Imma Pujro Huito,2014) “**Sistema de información vía web para el control y seguimiento de proyecto** “. En la Universidad Pública y Autónoma de El Alto carrera de Ingeniería de Sistemas del año 2014.el sistema de información fue desarrollado con la metodología UWE (UML – Based Web Engineering). Las herramientas tecnológicas que se utilizan para el desarrollo del Sistema con HTML, CSS y java Script para la interfaz del usuario, lenguaje de programación PHP utilizando el framework Codeigniter y para la administración de base de datos MySql.
- (Ramos,2008) “**Sistema de Control y Seguimiento Administrativo Financiero Sub Alcaldía distrito 7**”, ciudad del Alto en la Universidad Mayor de San Andrés, carrera de Informática. Se propone un software de contabilidad para generar y proporcionar información contable para la toma

de decisiones usa el método científico como metodología de investigación y la metodología estructurada para el desarrollo.

- (Huayhua,2008) **“Sistema de información integrada de control administrativo”**, financiero y de proyectos caso Empresa Constructora “ELECTROPACHA”, en la Universidad Mayor de San Andrés, Carrera de Informática. Sistema que aborda los conceptos de inteligencia de negocios usa la metodología OOHDM

### **1.3. Planteamiento del problema**

#### **1.3.1. Problema principal**

Al hacer el análisis de la situación actual de la empresa Construcción Innovación & Tecnología “C.I.T.”, se pudo identificar que la misma no cuenta con ningún control automatizado de gestión administrativa y seguimiento contable, en los procesos de ventas, compras, facturación los registros son realizados manualmente las cuales ocasionan pérdida de tiempo y datos erróneos, la cual implica la pérdida de información de la parte económicos.

#### **1.3.2. Problemas secundarios**

- Los registros de facturación son realizados de forma manual ocasionando duplicidad o llenado de datos erróneos que esto genera pérdida de tiempo.
- Facturación manual, retrasando los procesos de presentación de facturas para el descargo en impuestos nacionales.
- Al no contar con un respaldo de contratos adecuados, ocasiona una pérdida de la información de suma importancia para la empresa.
- Las ventas de materiales de construcción entre otros productos en diferentes licitaciones los cálculos de precios totales son realizados de manera manual ocasionando pérdida de tiempo y afectando en la toma de decisiones inmediatas.

- Retrasos en las entregas de reportes e información detallada de las compras realizadas de los diversos productos.
- Demora en la búsqueda de información de clientes, ocasionando pérdida de tiempo.
- No tener un control efectivo de todos los cálculos contables de la empresa, causando un trabajo moroso.

De esta manera se formula el siguiente interrogante:

¿Cómo se podrá mejorar la gestión administrativa y el seguimiento contable en la empresa Construcción Innovación & Tecnología “C.I.T.”?

#### **1.4. Objetivos**

##### **1.4.1. Objetivo general**

Desarrollar e implementar un Sistema Web de Gestión Administrativa y Seguimiento Contable, para la empresa Construcción Innovación & Tecnología “C.I.T.” que coadyuvará con la toma de decisiones contables y así proporcionar datos oportunos y confiables en tiempo real.

##### **1.4.2. Objetivos específicos**

- Realizar el análisis de la situación actual de la empresa.
- Determinar los requisitos necesarios para el sistema de información.
- Diseñar un módulo para la parte contable y optimizar el tiempo de proceso y búsqueda de datos.
- Establecer el módulo de reportes de cuentas.
- Realizar módulo de compras tanto para ventas.
- Realizar módulo de registros de la información contable.

- Automatizar el registro de datos de una forma confiable y eficiente en un gestor de base de datos en la cual se almacenará toda la información indispensable de la empresa.
- Aplicar métricas de calidad el ISO/IEC 25010.

## 1.5. Justificaciones

### 1.5.1. Justificación técnica

La empresa “CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA” cuenta con los equipos necesarios, también con cableado estructurado de red la cual tiene salida a internet.

Cuenta con los siguientes equipos:

Descripciones:

Tabla N.º 1. 1 Características de pc y laptop

Nombre de equipos	Cantidad	Sistema operativo	Hardware	Marca
Laptop	1	Windows 10 pro	Intel Core i5-6200U 2.3-2.8 GHz- Sexta G Pantalla LED 15.6" HD RAM 8GB DDR4 Disco Duro 1 TB DVD+RW, Bluetooth 4.0 y WiFi Puerto USB 3	HP
Computadora de escritorio	1	Windows 10 pro	Intel core i3-4000M 3.30 GHz- Cuarta G RAM 8GB DDR3 Disco Duro 1TB DVD+RW Puerto USB 6	
Router	1	RouterOS	Router Mikrotik, 5 puertos 10/100/1000, 2 CPU 880 MHz. RAM 256 MB, 1 USB, micro SD.	MIKROTIK RB750GR3
Switch	1		24 puertos RJ45 a 10/100/1000 Mbps Soporta aprendizaje de direcciones MAC (Auto-Learning) y MDI/MDIX automático.	TP-LINK TL- SG1024

Fuente: Elaboración propia.

### **1.5.2. Justificación económica**

La implementación del Sistema Web de Gestión Administrativa y Seguimiento Contable no tendrá ningún costo monetario para la empresa porque todo se realiza con herramientas de licencia pública y también el desarrollo no tiene ningún costo.

Más al contrario los beneficios serán:

- La empresa agilizará sus procesos en la toma de decisiones así podrá tener más ingresos.
- Tendrá un mejor manejo en la información de proyectos la cual le beneficiará a la empresa.
- La empresa podrá realizar su publicidad sin ningún costo mediante la página web.
- A comparación a otros medios publicitarios el mantenimiento de una página es relativamente económico. Además, de que no será una publicación temporal y podrás llegar a todas las personas.
- Los resguardos de información contable no tendrán costo porque en el sistema resguardará gratuitamente en la base de datos.

### **1.5.3. Justificación social**

Con la implementación del sistema los beneficiados serán: la empresa, gerente, contador, los empleados en la parte administrativa la cual les ayudara en realizar registros, reportes, consultas en tiempo real y la emisión de facturas de forma rápida.

Permitirá que los clientes aumenten su nivel de confianza con la empresa del servicio que ofrece, con la implementación de la página web se obtendrá más clientes que contraten los servicios de la empresa

## **1.6. Metodología**

### **1.6.1. Metodología de desarrollo de software**

Para el desarrollo se utilizarán una metodología ágil conocido como Scrumban que es una derivada de los métodos de desarrollo Scrum y Kanban. La cual es una metodología adecuado para proyectos de mantenimiento o proyectos en que los requisitos del software cambien con frecuencia o en los cuales se pueden dar errores de programación inesperados durante el desarrollo del producto.

## **1.7. Técnicas y Herramientas**

### **1.7.1. Recolección de datos**

Para la obtención de la información de la empresa se utiliza la entrevista es una conversación dirigida, con un propósito específico y se utiliza preguntas y respuestas.

La documentación permitirá los análisis de la organización de la empresa de cómo se realiza todo, se puede revisar documentos cualitativos y cuantitativos, en este caso solo aplicaremos cuantitativos con la cual obtendremos los reportes, consultas financieras, registros y formularios de cotizaciones.

### **1.7.2. Software de desarrollo**

Los softwares que se utilizaran en el desarrollo del proyecto:

- Framework Laravel
- JetBrains PhpStorm
- PHP 7
- Framework Bootstrap
- Xampp
- MySQL

- Composer

### 1.7.3. Hardware de desarrollo

El equipo utilizado para el desarrollo será de una laptop tiene las siguientes características:

Tabla N.º 1. 2 Características del equipo de desarrollo

Equipo	Sistema operativo	Hardware
HP ENVY x360 Convertible	Windows 10 Pro 64 bits	Procesador: Intel(R) Core (TM) i7-6500U CPU @ 2.50GHz. Memoria RAM: 16GB Disco: TOSHIBA MQ02ABF100 - 931,51 GB

Fuente: Elaboración propia

### 1.7.4. Software de implementación

El software de implementación es la página web que será desarrollada en el framework Laravel, PHP 7, la base de datos que estará en MySQL y un dominio.

### 1.7.5. Hardware de implementación

Se necesitará un equipo con requerimientos mínimos y con acceso a internet por la cual se podrá acceder al sistema web de gestión administrativa y seguimiento contable.

Características del equipo desde donde se podrá acceder:

Tabla N.º 1. 3 Características de dispositivos de acceso

Equipo	Explorador	Tipo de conexión para acceder al sistema
PC Tablet celulares Smart	Mozilla Firefox Google Chrome Internet Explorer Apple Safari Opera	Conexión a internet

Fuente: Elaboración propia

## **1.8. Métrica de calidad**

Para evaluar este proyecto se aplicará las métricas de calidad según el estándar ISO/IEC 25010. Que se encuentra propuesta por las ocho características que son:

- Adecuación funcional
- Eficiencia de desempeño
- Compatibilidad
- Usabilidad
- Fiabilidad
- Seguridad
- Mantenibilidad
- Portabilidad

## **1.9. Estimación de costos de software**

Para estimar el costo se utilizará la herramienta COCOMO II el cual permitirá realizar estimaciones en función del tamaño del software, y en un conjunto de factores de costos y de escala. Los factores de costos describen aspectos relacionados con la naturaleza del producto, hardware utilizado, personal involucrado y características propias del proyecto.

## **1.10. Seguridad**

### **1.10.1. ISO/IEC 25010**

La información es un activo valioso que puede impulsar o destruir una empresa. Si se gestiona de forma adecuada, le permite trabajar con confianza. Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos.

Por lo tanto, el sistema para la seguridad de la información se basará en dicha norma.

## **1.11. Límites y alcances**

### **1.11.1. Límites**

El sistema propuesto no realizará, los siguientes puntos:

- Los usuarios serán definidos y previamente establecidos, no cualquiera podrá ingresar en el sistema sin previa autorización.
- Los clientes no podrán administrar el sistema.
- Los clientes no podrán realizar pagos online.
- No podrá realizar la revalorización de los activos fijos.
- El sistema no cuenta con una conexión directa con impuestos nacionales.

### **1.11.2. Alcances**

El sistema web facilita la gestión administrativa, se realizará los siguientes alcances:

- **Módulo de administración de usuarios:** Se otorgará privilegios al sistema y se aplicará técnicas de seguridad en cuanto al ingreso no autorizado.
- **Módulo de facturación:** Para la impresión de las facturas emitidas por la empresa.
- **Módulo de ventas y compras:** Se registrarán todos aquellos pedidos de compra y ventas de materiales solicitados por los clientes.
- **Módulo de reportes:** En el cual se pueden ver los reportes de existencia como de compras, ventas, las cuales coadyuvarán a la toma de decisiones.
- **Módulo de archivos:** Para obtener un respaldo de todos los archivos de cada proyecto o trabajos que se realizarán y entre otros.

- **Módulo de cliente:** Para tener un control de registro de las diversas ventas realizadas a los clientes.
- **Módulo de Proveedores:** Lista de proveedores de los cuales se realiza la compra de los diferentes productos.
- **Módulo contable:** En este módulo se tomará en cuenta los siguientes Submódulos:
  - Plan de cuentas.
  - Libro diario.
  - Libro mayor.
  - Balance general.
  - Generar reportes.

## **1.12. Aportes**

### **1.12.1. Aporte Práctico**

En este caso el sistema será el aportare, el mismo sistema de gestión administrativa y seguimiento contable aplicado en la empresa que no ha sido desarrollada hasta el momento.

### **1.12.2. Aparte Teórico**

El aporte teórico es el desarrollo del sistema utilizando la Metodología Scrumban para una mejor organización y para el modelado de diagramas se empleará UML para facilitar el diseño del sistema para luego programarla en el Leguaje PHP 7 utilizando un gestor de base de datos MySQL, framework Laravel y todo esto debe de estar en la web para a el dueño(a) de la empresa se conecte al internet inmediatamente pueda ver su información y tener respuestas en tiempo real y llevar un adecuado registro en la administración contable.

## **2. MARCO TEÓRICO**

## **2.1. Introducción**

En este capítulo se realizan las descripciones e introducciones a los principios y conceptos básicos para la elaboración del proyecto, sin embargo, no se podrá dar una teoría completa acerca de las metodologías, técnicas y herramientas que serán utilizados que se presenta para fácil comprensión de la misma.

## **2.2. Gestión administrativa**

Una Gestión Administrativa debe proveer información razonada, en base a registros técnicos, de las operaciones realizadas por la empresa con el fin de interpretar sus resultados. Estos datos permitirán conocer la estabilidad y solvencia de la compañía, la situación de cobros y pagos, las tendencias de las ventas, costes y gastos generales, entre otros. De este modo se podrá conocer la capacidad financiera de la empresa y tomar decisiones estratégicas en base a datos reales (Tibel Consulting, 2016).

## **2.3. Seguimiento contable**

La idea de seguimiento contable, por su parte, hace referencia a aquello vinculado a la contabilidad (el método que permite llevar las cuentas de una organización).

La noción de sistema contable, de este modo, puede entenderse de distintas maneras. En su sentido más amplio, se trata del conjunto de elementos que registran la información financiera y las interrelaciones de estos datos. Esta estructura, por sus características, contribuye a la toma de decisiones en el ámbito de la gerencia.

En concreto, se puede establecer que los pilares de todo sistema contable son tres, los que le permiten darle forma:

- Registro de la actividad financiera, que debe estar al día y que debe estar recogida en el documento establecido para ello.

- Clasificación de la información en grupos y categorías para que pueda ser estudiada, analizada y utilizada de la manera más adecuada posible.
- Resumen de toda la información de la contabilidad de la empresa en cuestión, para que pueda ser tomada en cuenta por las personas que se encuentran al frente de la entidad.

En la actualidad, el concepto de sistema contable suele asociarse al programa informático que permite registrar la información. El software contable cuenta con diferentes módulos para que una empresa pueda llevar sus libros y balances de manera digital y con herramientas que facilitan los cálculos (Julián Pérez, María Merino, 2016).

#### **2.4. Ingeniería de software**

La ingeniería de software es una disciplina que se interesa por todos los aspectos en la elaboración de un software desde las primeras etapas de la especificación del sistema hasta el mantenimiento después que se pone en operación (Somerville, 2011).

Es el proceso utilizado para crear, implantar y mantener aplicaciones y Sistemas Web de alta calidad. Esta breve definición nos lleva a abordar un aspecto clave de cualquier proyecto como es determinar qué tipo de proceso es más adecuado en función de las características del mismo. El desarrollo de aplicaciones Web posee determinadas características que lo hacen diferente del desarrollo de aplicaciones o software tradicional y Sistemas de información. Uno de los aspectos más tenidos en cuenta, en el desarrollo de sitios web es sin duda alguna el diseño gráfico y la organización estructural del contenido. En la actualidad la web está sufriendo grandes cambios, que han obligado a expertos en el tema a utilizar herramientas y técnicas basadas en la Ingeniería del Software, para poder garantizar el buen funcionamiento y administración de los sitios web. Para garantizar el buen funcionamiento y mantenimiento de los sitios web, este debe contar con ciertos atributos y características que en conjunto forman un concepto muy importante, para

alcanzar el éxito en cualquier organización, herramienta, y todo aquello que se pueda considerar como servicio. Dicho concepto es la calidad, que con atributos como, usabilidad, navegabilidad, seguridad, Mantenibilidad, entre otros, hace posible por un lado la eficiencia del artefacto web y por ende la satisfacción del usuario final (Jennyfer López, 2010)

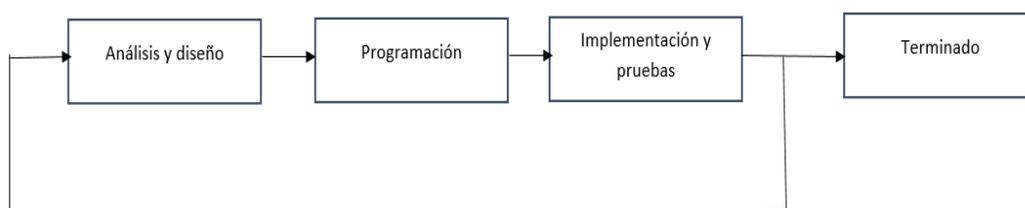
En este sentido, la ingeniería de la Web hace referencia a las metodologías, técnicas y herramientas que se utilizan en el desarrollo de aplicaciones Web complejas y de gran dimensión en las que se apoya la evaluación, diseño, desarrollo, implementación y evolución de dichas aplicaciones.

#### **2.4.1. Método de ciclo de vida**

Este método busca reducir los riesgos que surge entre las necesidades del usuario y el producto final por malos entendimientos durante la etapa de los requisitos, el cliente será quien evalúe después de cada iteración el producto, estas iteraciones se repetirán hasta obtener un producto satisfactorio.

Adaptando el método de ciclo de vida a Scrumban las iteraciones se repetirán como en la siguiente figura:

Figura N.º 2. 1 Método de ciclo de vida iterativo



Fuente: Elaboración propia

#### **2.5. Metodología de desarrollo Scrumban**

Para el desarrollo del sistema se utiliza una metodología ágil conocida como Scrumban, la cual es la combinación de los métodos ágiles más importantes en la actualidad: Scrum y Kanban.

**Scrum.** - Es una metodología ágil de desarrollo de software es adecuado para aquellos proyectos en que el trabajo de desarrollo y el mantenimiento de productos complejos (Guzmán, Castañeda, Pérez & pedroza,2014).

Características de la metodología:

- Divide la organización en pequeños equipos, interdisciplinarios y auto organizados.
- Divide el trabajo en una lista entregable, ordena la lista por prioridad y estima el esfuerzo relativo de cada elemento.
- Divide el tiempo en iteraciones cortas, generalmente de 1 a 4 semanas con el código entregable y demostrado después de cada iteración.
- Optimiza el plan de entregas y actualiza las prioridades en colaboración con el cliente, basada en los conocimientos adquiridos mediante las inspecciones de cada iteración.

**Kanban.** - Es una metodología de desarrollo ágil que es empleada para controlar el avance de un trabajo de una línea de producción mediante el uso de tarjetas.

En la actualidad Kanban es reconocida parte de las metodologías ágiles, cuyo objetivo es gestionar de manera general de cómo se realizan las tareas completas.

Entre las principales características de Kanban según (Guzmán, Castañeda, Pérez & pedroza,2014).

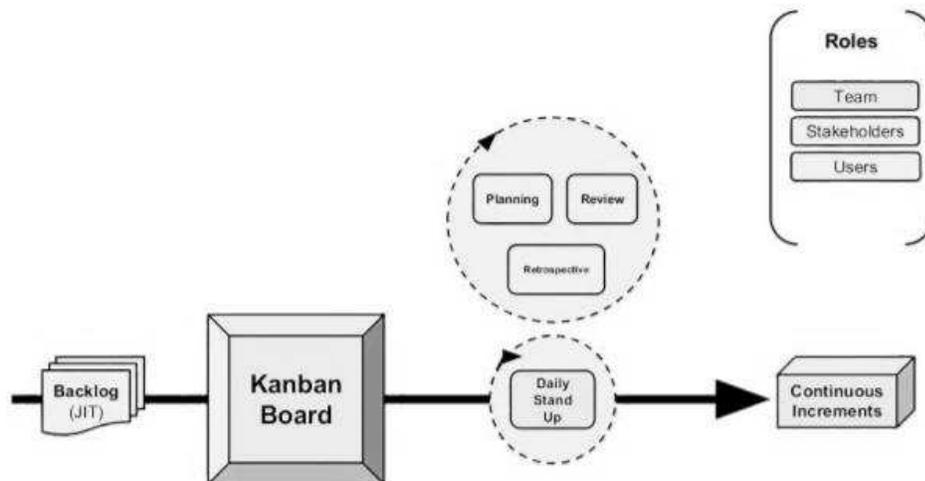
- Se puede visualizar cada flujo de trabajo en bloques, escribiendo cada elemento en una tarjeta y colocando en un muro, para luego posteriormente utilizar columnas con nombres para ubicar cada elemento en el flujo de trabajo.
- Limita el proceso de trabajo en curso, asigna límites concretos a cuantos elementos pueden estar en progreso en cada estado del flujo de trabajo.

### 2.5.1. Scrumban

Es la combinación de dos metodologías ágiles que son Scrum y Kanban utilizando de cada uno lo mejor de sus procesos. Scrumban aprovecha la visualización de los procesos por etapas y asignación del equipo en un tablero con tarjetas pegadas que presenten las tareas a realizar y que son seleccionadas por el personal apropiado, hasta llegar a la etapa de conclusión (Guzmán, Castañeda, Pérez & Pedroza, 2014).

Scrumban se está haciendo especialmente popular en industrias en las que el desarrollo del proyecto y el mantenimiento van juntos.

Figura N.º 2. 2 Descripción general de Scrumban



Fuente: (Guzmán, Castañeda, Pérez & Pedroza, 2014).

Las cuales proporcionan un límite de tareas en las que se esté trabajando, tanto en equipo y de manera personal, cada persona del equipo solo puede realizar dos tareas al mismo tiempo, este se debe por que en algún momento se tendrá que flexibilizar la aceptación de otra tarea

#### 2.5.1.1. Roles Scrumban

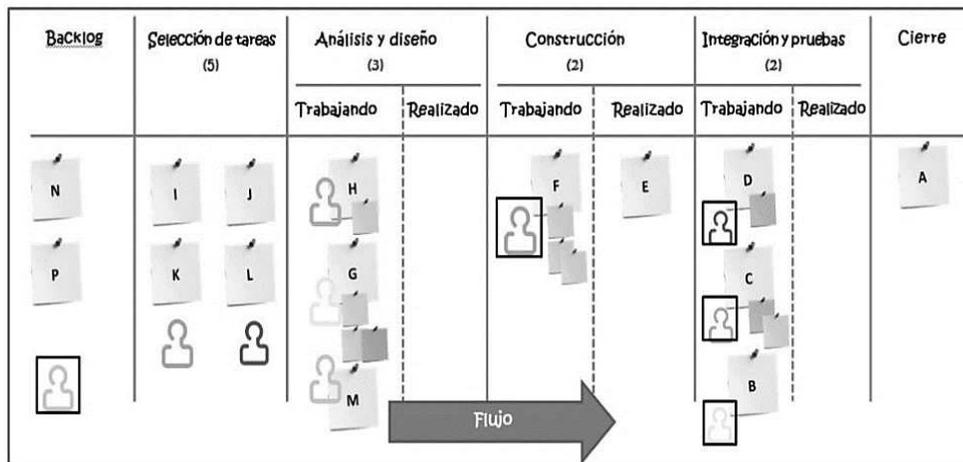
Scrumban es caracterizado en que los roles son el equipo de trabajo, las partes interesadas y el usuario final. los roles que tiene son:

- **RP:** responsable del proyecto
- **TEAM:** el equipo de desarrollo
- **USER:** el cliente para el cual se hace el proyecto

### 2.5.1.2. Actividades en Scrumban

- **Visualizar el flujo de trabajo:** es una de las herramientas tomadas de Kanban y se aplica en Scrumban. La cual permite visualizar el flujo de trabajo. En un Scrum normal el equipo empieza desde el sprint backlog y trabaja en los artículos, finalmente son movidos a la etapa de los hechos. Mientras en Scrumban la idea es visualizar el flujo de trabajo dentro y fuera del Sprint, la visualización permitirá al equipo de trabajo poder evitar los cuellos de botellas. las visualizaciones se realizan en una pizarra blanca o con una herramienta digital se realizan las divisiones en todo el tablero en las diferentes etapas (Guzmán, Castañeda, Pérez & pedroza,2014).

Figura N.º 2. 3 Flujo de Trabajo de Scrumban



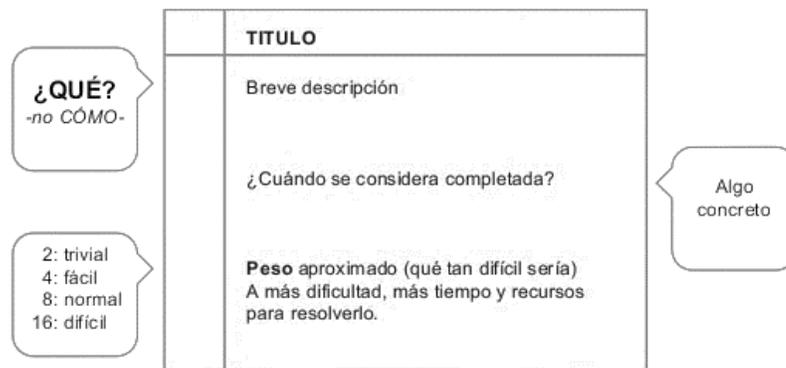
Fuente: (Guzmán, Castañeda, Pérez & pedroza,2014).

- **Cola de trabajo:** en Scrumban el trabajo ingresa en cola a diferencia de un Scrum tradicional, donde todo el trabajo tiene una fecha de inicio y fecha de entrega para el sprint backlog. Las tareas en cola son de alta prioridad y en

la cola no hay tareas vinculadas a una determinada persona el primero que se desocupe debe tomar una de las tareas en lugar de recoger de la cartera general (Guzmán, Castañeda, Pérez & pedroza,2014).

- **Límite de trabajo en proceso:** se aplica límite de trabajos en los puntos de progreso en todas las etapas, basado en la capacidad del equipo, esto es extraído de Kanban. En Scrum su significado es limitar los elementos de la pila de producto que está en curso en cualquier momento, incluye el sprint backlog. El objetivo es ayudar al resto del equipo cuando uno haya terminado antes del tiempo su actividad así lograr evitar los cuellos de botella (Guzmán, Castañeda, Pérez & pedroza,2014).
- **La tarjeta de una tarea:** En el tablero de Scrumban las tarjetas contienen el título, descripción de la tarea, descripción de la tarea completada y un valor que identifique la dificultad de la tarea.

Figura N.º 2. 4 Descripción de la Tarjeta de Scrumban



Fuente: (Guzmán, Castañeda, Pérez & pedroza,2014).

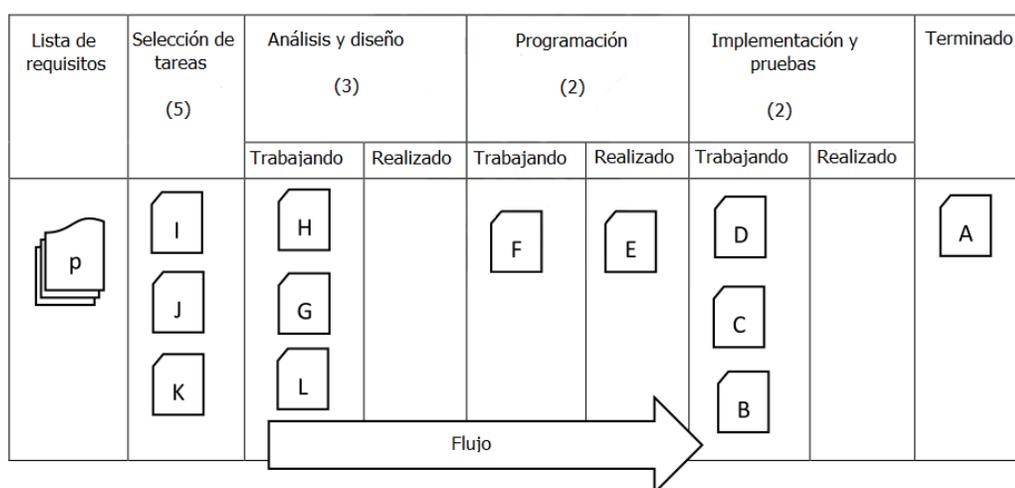
- **Reglas explícitas:** En Scrumban las reglas del equipo se hacen explícitas para que todos en el equipo estén para auto organizarse, con el fin de lograr un flujo de trabajo más satisfactorio (Guzmán, Castañeda, Pérez & pedroza,2014).

- **Reuniones de planificación:** Scrumban tiene las reuniones de planificación más cortas, con el fin de actualizar la cola de registros cuando sea necesario. El equipo siempre debe planificar para periodos muy cortos por delante.

### 2.5.2. Fases del modelo de desarrollo Scrumban

En el proceso del modelo de desarrollo que se utiliza para el proyecto la metodología Scrumban para obtener la representación física y conceptual del sistema hasta el momento de la implementación.

Figura N.º 2. 5 Modelo de proceso de desarrollo del proyecto



Fuente: (José Luis. M. Rodríguez, 2017).

**Arquitectura y Requerimiento de software.** – Se observará de forma gráfica la arquitectura y para la comprensión de los requerimientos de software se elaborará un diagrama de caso de uso general del funcionamiento del sistema.

Requerimientos para el desarrollo de software e implementación se deberán seguir las siguientes fases de acuerdo al modelo de desarrollo Scrumban.

**Fase 1: Lista de requisitos.** - En esta fase para poder seleccionar las tareas se realiza una lista de requisitos priorizadas.

**Fase 2: Descripción de las tarjetas de tareas.** - En esta fase se elaboran las tarjetas de la lista de requerimientos priorizados.

- **Diseño de base de datos e implementación.** - Primera tarea de mayor prioridad que se debe elaborar, la tarea de diseño del modelo relacional de la base de datos con todos los requerimientos del sistema.

**Fase 3: Análisis y diseño.** - Una vez que las tareas pasan a esta fase, se realizan los diseños de requisitos en diagramas de UML para luego pasar a ser programados. Las cuales se realizan en la siguiente orden:

- Caso de uso
- Diagrama de clases
- Diagrama de actividades o de flujo
- Diagrama de secuencia

**Fase 4: Programación.** - En esta fase se realiza el desarrollo del sistema de los módulos diseños en UML.

**Fase 6: Implementación y pruebas.** - En esta fase se realizan la implementación del sistema y todas las pruebas necesarias de integración unitariamente.

- **Resultados.** - Se realiza la tabla de resultados obtenidos de la implementación y pruebas de cada módulo programado unitariamente.

### **2.5.3. Comparación de Scrumban**

A continuación, se muestra las siguientes diferencias entre estos tres.

Tabla N.º 2. 1 Comparación de Scrum, Kanban y Scrumban

	<b>Scrum</b>	<b>Kanban</b>	<b>Scrumban</b>
Visualización del Trabajo	Parcial	Backlog con límites	Total
Backlog	Sprint backlog, ProductBacklog	Limitado	Backlog con límites
Límite de Trabajos en Proceso	No límites	Según sea necesario	Limitado
Cambios	Debe esperar hasta el siguiente Sprint	No	Según sea necesario
Roles	Si	No	Si
Estimaciones	Si	No	No
Iteraciones	Si	Tablero persistente	Si

Fuente: (Guzmán, Castañeda, Pérez & pedroza,2014).

## 2.6. Lenguaje de modelado unificado (UML)

Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar artefactos de un sistema de software. Se usa para entender, diseñar, hojear, configurar, mantener y controlar la información sobre los sistemas, está pensado para usarse con todos los métodos de desarrollo, etapas del ciclo de vida, dominios de aplicación y medios. El sistema de modelado la cual es una colección de objetos discretos que interactúan para realizar un trabajo que beneficie a un usuario externo. Las estructuras estáticas definen los tipos de objetos importantes para un sistema y para su implementación. También, así como las relaciones entre objetos (Rumbaugh, Jacobson & Booch,200, p 3).

### 2.6.1. Vistas de UML

Una vista de UML es simplemente un subconjunto que modela construcciones que representan un aspecto de un sistema. La división en diversas vistas es algo arbitraria, una o dos clases de diagramas proporciona una notación visual para los conceptos de cada vista. La vista de gestión de modelo cruza las otras vistas y las organiza para el trabajo de desarrollo y control de configuración (Rumbaugh, Jacobson & Booch,2000, p 21).

Tabla N.º 2. 2 Muestra de las Vistas de UML

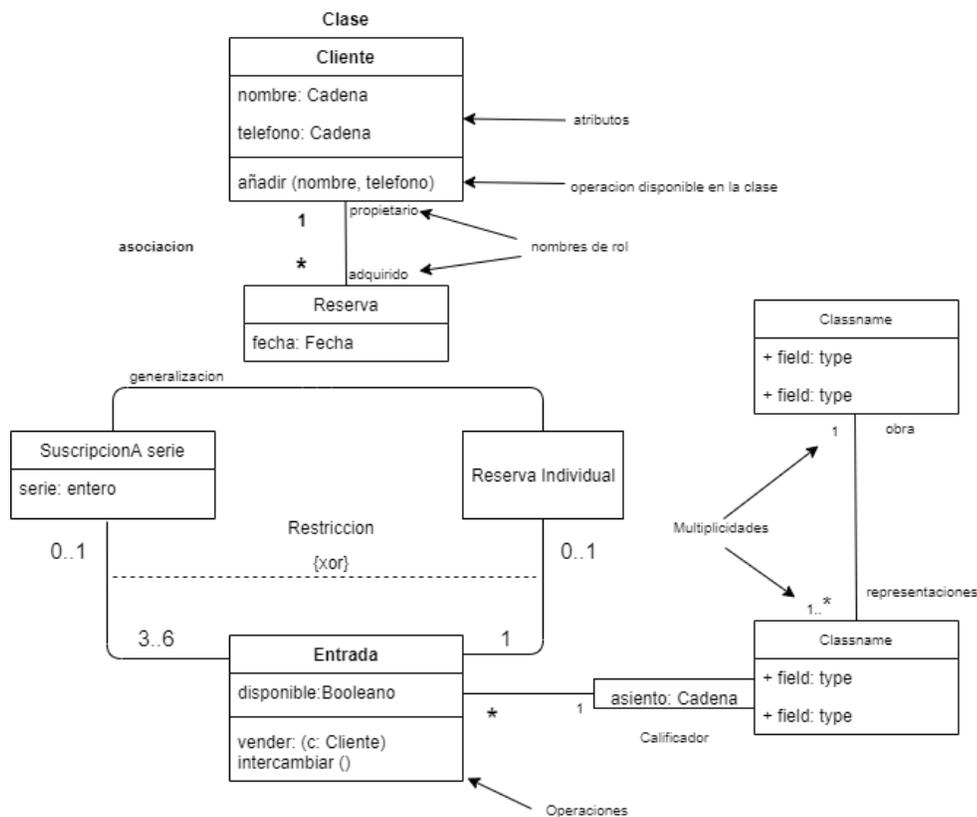
Área	Vista	Diagrama	Conceptos principales
Estructural	Vista estática	Diagrama de clases	Clase, asociación, generalización, dependencia, realización, interfaz
	Vista de casos de uso	Diagrama de casos de uso	Caso de uso, actor, asociación, extensión, inclusión, generalización de casos de uso
	Vista de implementación	Diagrama de componentes	Componentes, interfaz, dependencia, realización
Diagrama	Vista de despliegue	Diagrama de despliegues	Nodo, componentes, dependencia, localización
	Vista de máquina de estados	Diagrama de estados	Estado, evento, transición, acción
	Vista de actividad	Diagrama de actividad	Estado, actividad, transición de terminación, división, unión.
	Vista de interacción	Diagrama de secuencias	Interacción, objeto, mensaje, actividad
		Diagrama de colaboración	Colaboración, interacción, rol de colaboración, mensaje
Gestión de modelo	Vista de gestión del modelo	Diagrama de clases	Paquete, subsistema, modelo
Extensión de UML	Todas	Todos	Restricción. Estereotipo, valores etiquetados

Fuente: (Rumbaugh, Jacobson & Booch, 2000, p 22).

### 2.6.1.1. Vista estática

Modela los conceptos del dominio de la aplicación como parte de la implementación. Los componentes principales de la vista estática son las clases y sus relaciones. La visión estática se exhibe en los diagramas de clases, llamados así porque el objetivo es la descripción de las clases. (James Rumbaugh, Ivan Jacobson, Grady Booch,2000, p 23).

Figura N.º 2. 6 Diagrama de Clases

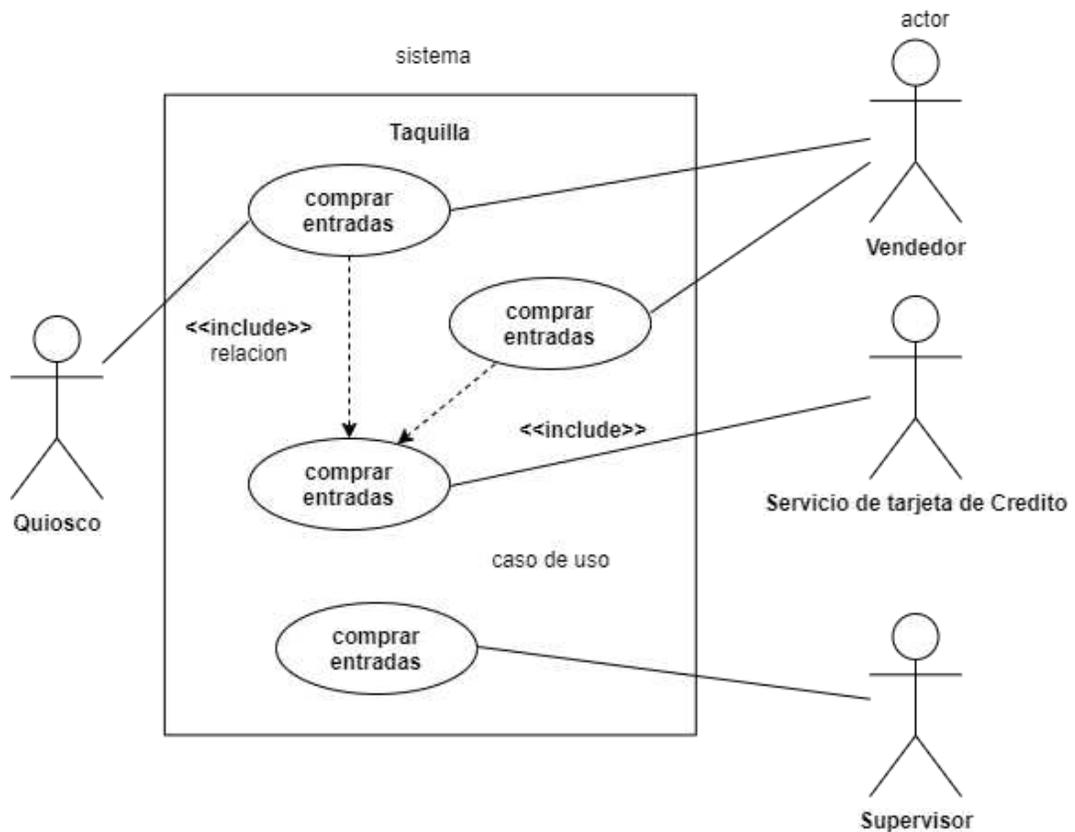


Fuente: (Rumbaugh, Jacobson & Booch,2000, p 23).

### 2.6.1.2. Vista de los casos de uso

Modela la funcionalidad del sistema según lo perciben los usuarios externos, llamado actores. El caso de uso es una unidad coherente de funcionalidad, expresa como transacciones entre actores y el sistema. (Rumbaugh, Jacobson & Booch,2000, p 24).

Figura N.º 2. 7 Diagrama de casos de uso



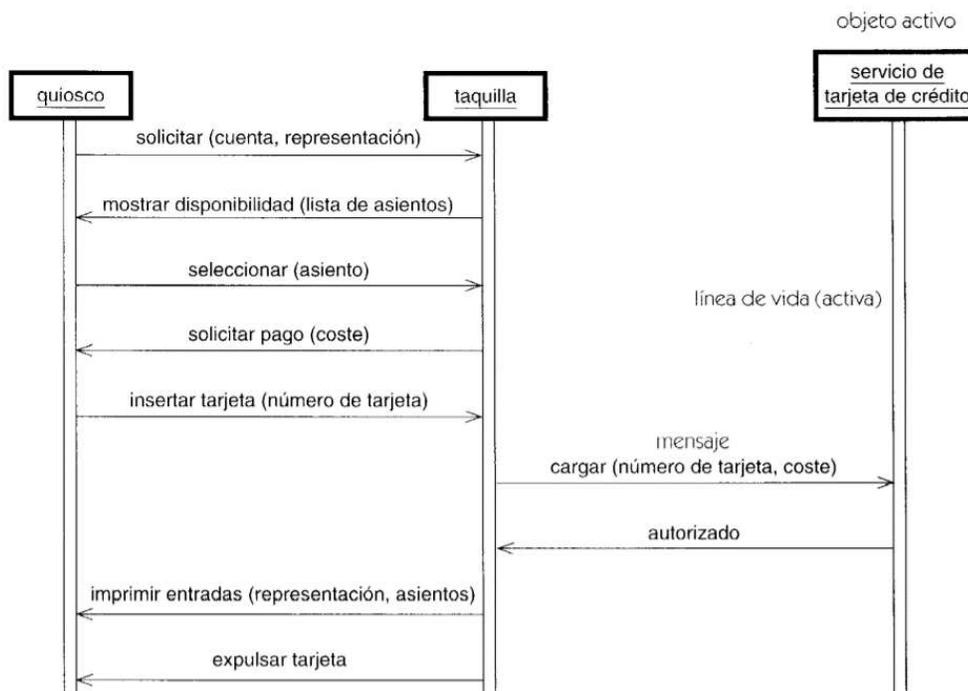
Fuente: (Rumbaugh, Jacobson & Booch,2000, p 24).

### 2.6.1.3. Vista de interacción

#### Diagrama de secuencia:

Describe las secuencias de intercambio de mensajes entre los roles. Cada rol en la secuencia se muestra como una línea de vida, en una línea vertical (Rumbaugh, Jacobson & Booch,2000, p 25).

Figura N.º 2. 8 Diagrama de secuencias

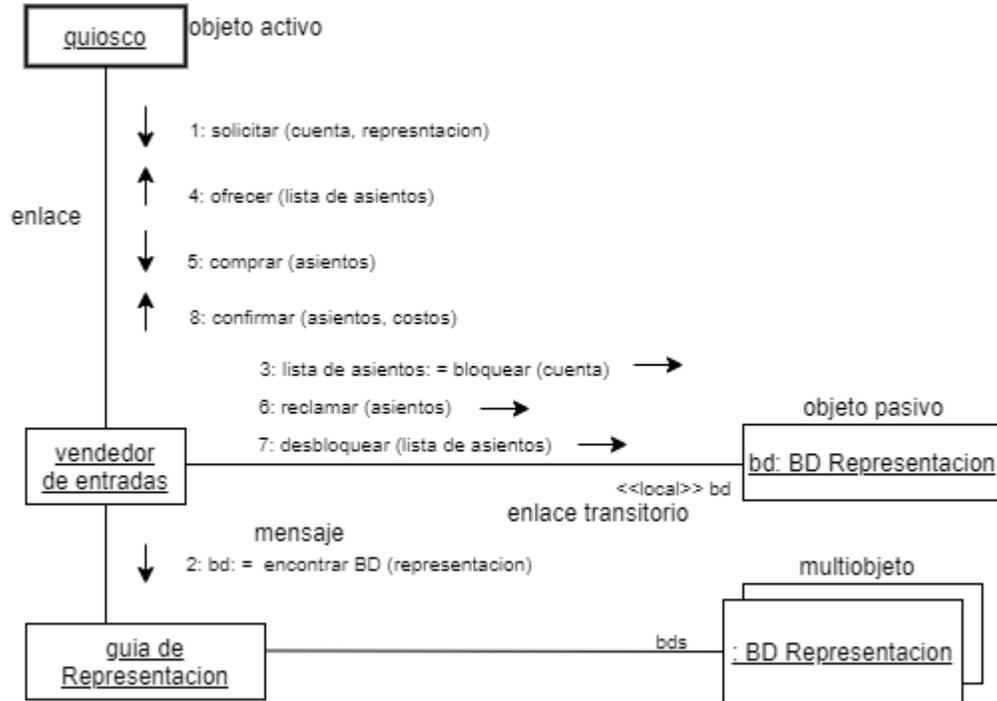


Fuente: (Rumbaugh, Jacobson & Booch,2000, p 25).

### Diagrama de colaboración:

Los diagramas de colaboración muestran las iteraciones, pero acentúan aspectos diferentes. Un diagrama de secuencias muestra el tiempo como dimensión geométrica pero las relaciones de roles son implícitas. El diagrama de colaboración muestra las relaciones entre roles geométricamente y relaciona los mensajes con las relaciones (Rumbaugh, Jacobson & Booch,2000, p 26).

Figura N.º 2. 9 Diagrama de Colaboración

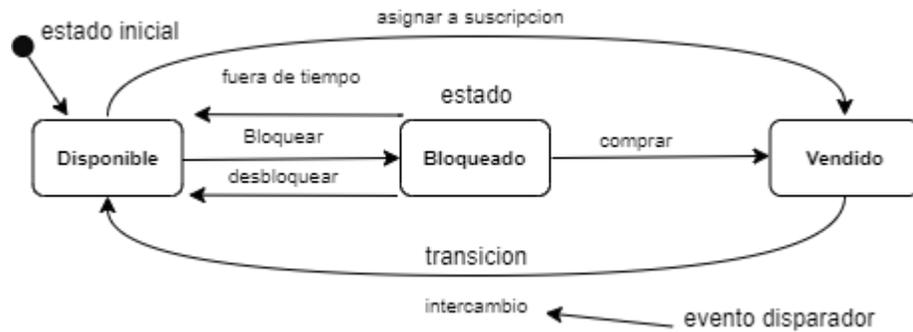


Fuente: (Rumbaugh, Jacobson & Booch, 2000, p 27).

#### 2.6.1.4. Vista de la máquina de estado

Es la posible historia de vida de un objeto de una clase, cuando ocurre un evento se desencadena una transacción que lleva al objeto a un nuevo estado. Cuando se dispara una transacción es unidad a una transmisión las máquinas de estado se muestran como diagramas de estados (Rumbaugh, Jacobson, Booch, 2000, p 27).

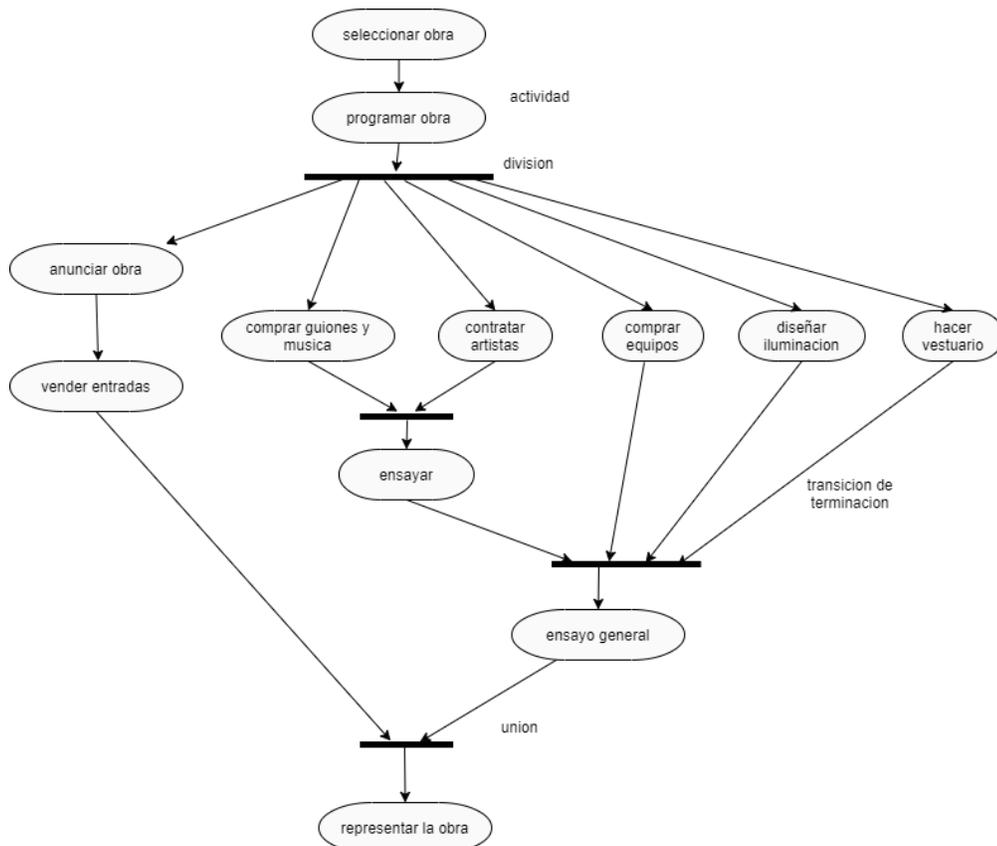
Figura N.º 2. 10 Diagrama de Estado



Fuente: (Rumbaugh, Jacobson & Booch,2000, p 27).

La máquina de estados se puede utilizar para describir las interfaces del usuario, controladores de dispositivos y otros subsistemas.

Figura N.º 2. 11 Diagrama de Estado



Fuente: (Rumbaugh, Jacobson & Booch,2000, p 27).

### **2.6.1.5. Vista de actividades**

El gráfico de actividades es una variante de la máquina de estados que muestra las actividades. Un estado de actividad de representa una actividad del flujo de trabajo o la ejecución de una operación. Un grafo de actividades describe grupos secuenciales y concurrentes de actividades, los grafos de actividades se muestran en diagramas de actividades. Cada parámetro de entrada y salida de una acción es demostrado usando las relaciones de flujo que conecten las acciones y un estado de flujo del objeto. (Rumbaugh, Jacobson & Booch,2000, p 29).

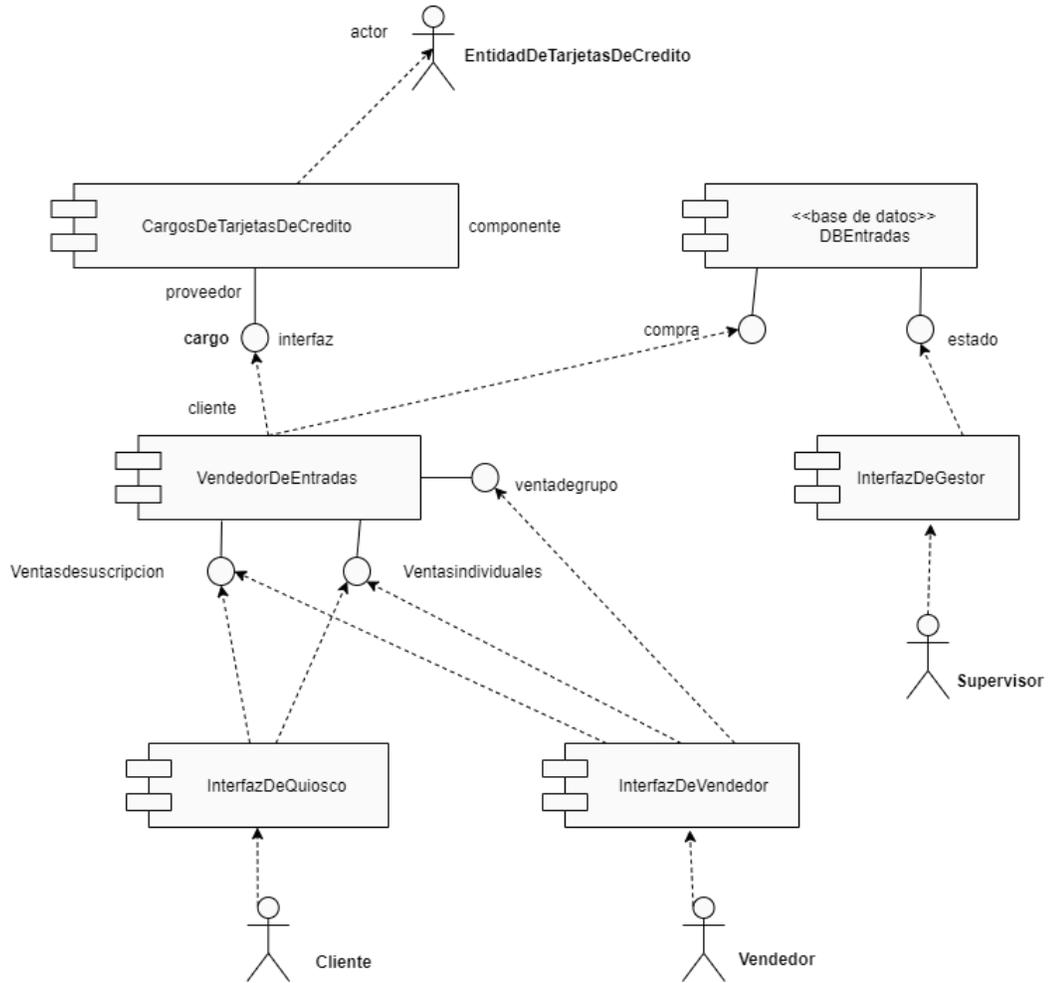
### **2.6.1.6. Vistas físicas**

Las vistas anteriores modelan los conceptos de la aplicación desde un punto de vista lógico. Las vistas físicas modelan la estructura de la implementación de la aplicación. Establece correspondencia entre las clases y los componentes de implementación y nodos (Rumbaugh, Jacobson & Booch,2000, p 29).

Hay dos vistas físicas: la vista de implementación y vista de despliegue.

- **Vista de implementación:** se presenta en diagramas de componentes

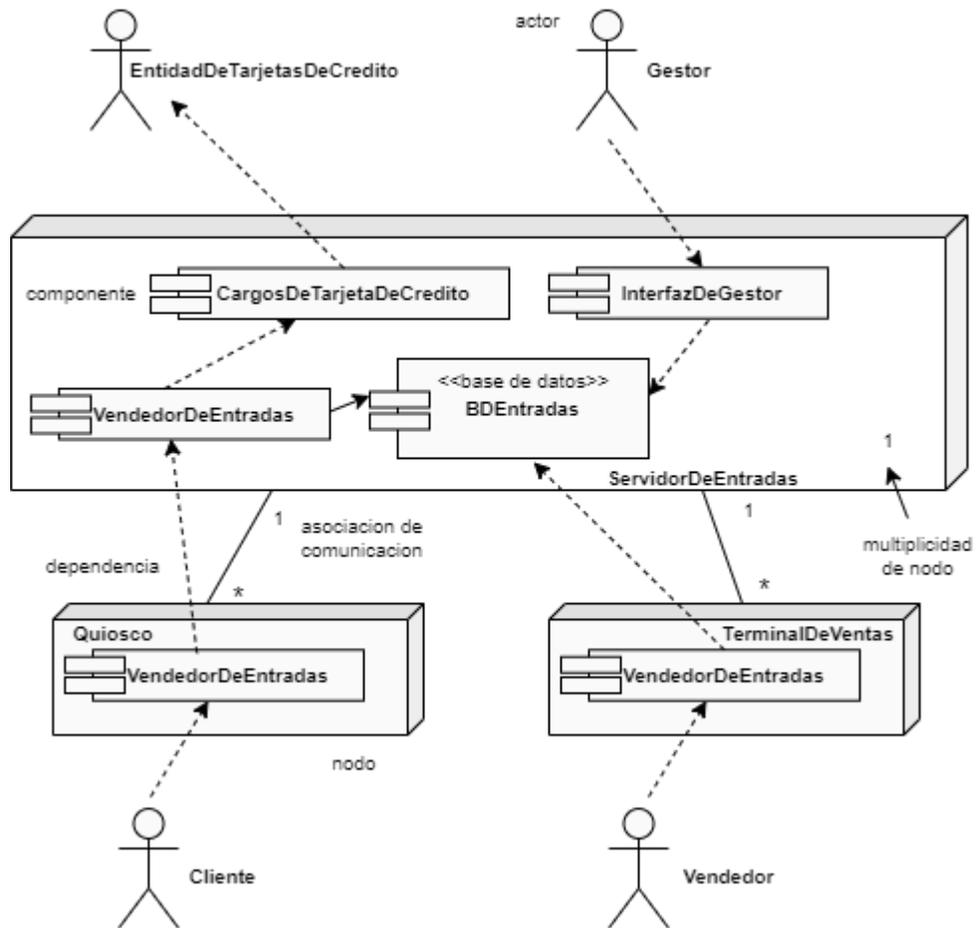
Figura N.º 2. 12 Diagrama de componentes



Fuente: (Rumbaugh, Jacobson & Booch,2000, p 30).

- **Vista de despliegue:** representa la disposición de las instancias de componentes de instancias de nodos.

Figura N.º 2. 13 Diagrama de despliegue (nivel de descriptor)

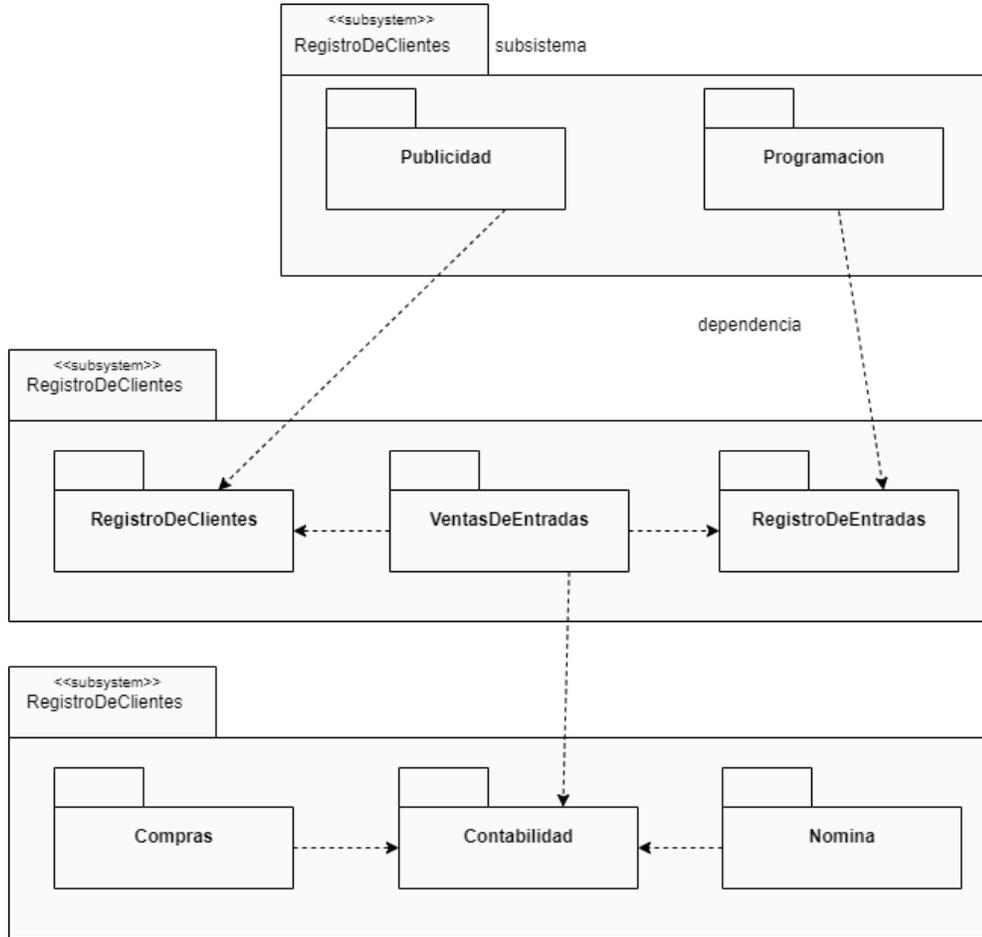


Fuente: (Rumbaugh, Jacobson & Booch,2000, p 31).

### 2.6.1.7. Vista de gestión del modelo

Es un modelo de descripción completa de un sistema con una determinada precisión desde un punto de vista, puede haber varios modelos de un sistema desde diferentes puntos de vista. Como el modelo análisis y un modelo de diseño se presenta como una clase especial de paquetes (Rumbaugh, Jacobson & Booch,2000, p 32).

Figura N.º 2. 14 Paquetes



Fuente: (Rumbaugh, Jacobson & Booch,2000, p 33).

### 2.6.1.8. Construcciones de extensión

Estas construcciones permiten que muchas clases de extensiones a UML, sin requerir cambios al metamodelo básico de UML. Pueden ser utilizadas para crear otras versiones adaptadas a un área de aplicación (Rumbaugh, Jacobson & Booch,2000, p 33).

### 2.6.1.9. Conexión entre vistas

Se muestra algunas de las conexiones de elementos, de diversas vistas.

Tabla N.º 2. 3 Algunas Relaciones entre Elementos de Diferentes Vistas

Elemento	Elemento	Relación
Clase	Máquina de estados	Propiedad
Operación	Interacción	Realización
Caso de uso	Colaboración	Realización
Caso de uso	Instancias de interacción	Escenas de ejemplo
Instancia de un componente	Instancia de un nodo	Localización
Acción	Operación	llamada
Acción	Señal	Envío
Actividad	Operación	llamada
Mensaje	Acción	Invocación
Paquete	Clase	Propiedad
Rol	Clase	Clasificación

Fuente: (Rumbaugh, Jacobson & Booch,2000, p 35).

## 2.7. Herramientas

### 2.7.1. Framework Laravel

Laravel es un framework PHP que trabaja con una arquitectura de carpetas avanzadas, de modo que promueve la separación de archivos con un orden correcto y definido, también dispone de una arquitectura de clases también muy adecuado que promueve la separación de código en una arquitectura de modelo-vista-controlador. Laravel pone en énfasis la calidad del código y la escalabilidad, mantenimiento lo cual permite realizar proyectos pequeños a grandes y muy grandes (desarrolloweb.com, 2020).

Contiene un amplio conjunto de características, que sirven para realizar la mayoría de las aplicaciones web:

- Un sistema de rutas, mediante la cual es fácil crear y mantener todo tipo de URLs.

- Un sistema de abstracción de base de datos, con un Mapeo Relacional de Objetos potente pero sencillo de manejar, mediante el podemos tratar los datos de la base de datos como si fueran simples objetos.
- Varias configuraciones para envíos de email, con diversos proveedores.
- Gestión de sesiones.
- Sistema de autenticación, con todo lo necesario.
- La posibilidad de acceder a los datos en realtime y recibir notificaciones cuando estos se alteren en la base de datos.

#### **2.7.1.1. Cómo funciona Modelo-Vista-Controlador (MVC)**

Estructuralmente el diseño del software está en tres partes:

- Modelo
- Vista
- Controlador

**Modelo.** – Es donde se procesa y obtienen los datos, la conexión con la base de datos.

**Vista.** – Presenta los datos en la pantalla, es donde va el código HTML.

**Controlador.** – Se encarga de controlar los datos de forma rápida que obtiene los datos en un modelo, luego los procesa y se los pasa a la vista.

#### **2.7.2. JetBrains PhpStorm**

Es un editor de texto con facilidad para adquirir licencia gratis para estudiantes y profesionales, es multiplataforma es una herramienta básica en el entorno de trabajo del desarrollo. PhpStorm es un editor muy completo y potente, con posibilidades

que integran una gran colección de extensiones que permite añadir las funcionalidades que el programador necesita en su día a día (jetbrains.com, 2019)

Alguna de las funcionalidades más destacados que se encuentran son:

- Resaltado de sintaxis para infinidad de lenguajes de programación (PHP, HTML, SQL, C#, Python, Java, CSS, JavaScript, Go, Ruby y entre otros.).
- Autocompletado con soporte para los lenguajes de programación más conocidos. La cual permite escribir un código más eficiente gracias a su sistema de autocompletado.
- Integración con Git, Github y Bitbucket que dispone de herramientas de control de versiones para poder gestionar.
- En momento de programar se puede depurar el código directamente y buscar errores en la sintaxis.

### **2.7.3. PHP 7**

PHP es una sigla a la cuales 'Hypertext Preprocessor', es un lenguaje de programación interpretado que se utiliza para la generación de páginas web de forma dinámica. Este código se ejecuta al lado del servidor, cabe destacar que es un lenguaje de código abierto, gratuito y multiplataforma. PHP es bastante simple para el principiante, pero también es completo y completo para profesionales (Hernán Beati. 2011. Pag.2).

### **2.7.4. Framework Bootstrap**

Es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web. Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS, así como extensiones de JavaScript adicionales (getbootstrap.com, 2019).

### **2.7.5. Xampp**

XAMPP es un servidor independiente de plataforma, software libre, que consiste principalmente en la base de datos MySQL, el servidor Web Apache y los intérpretes para lenguajes de script: PHP y Perl. El nombre proviene del acrónimo de X (para cualquiera de los diferentes sistemas operativos), Apache, MySQL, PHP, Perl. El programa está liberado bajo la licencia GNU y actúa como un servidor Web libre, fácil de usar y capaz de interpretar páginas dinámicas. Actualmente XAMPP está disponible para Microsoft Windows, GNU/Linux, Solaris, y MacOS) (apachefriends.org, 2019).

### **2.7.6. MySQL**

Es un sistema de gestión de bases de datos que es seguro, rentable aun nivel empresarial y relacional, desarrollado bajo licencia dual: Licencia pública general/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto (mysql.com, 2019).

### **2.7.7. Composer**

Composer es una herramienta para la gestión de dependencias en PHP. Le permite declarar las bibliotecas de las que depende su proyecto y las administrará (instalará / actualizará) por usted (getcomposer.org, 2019).

## **2.8. Pruebas de software**

### **2.8.1. Prueba de caja blanca**

La prueba de caja blanca se basa en el diseño de casos de prueba que usa la estructura de control del diseño procedimental para derivarlos. Mediante esta prueba de la caja blanca se puede obtener casos de prueba que:

- Garanticen que se ejerciten por lo menos una vez todos los caminos independientes de cada módulo, programa o método.
- Ejerciten todas las decisiones lógicas en las vertientes verdadera y falsa.

- Ejecuten todos los bucles en sus límites operacionales.
- Ejerciten las estructuras internas de datos para asegurar su validez.

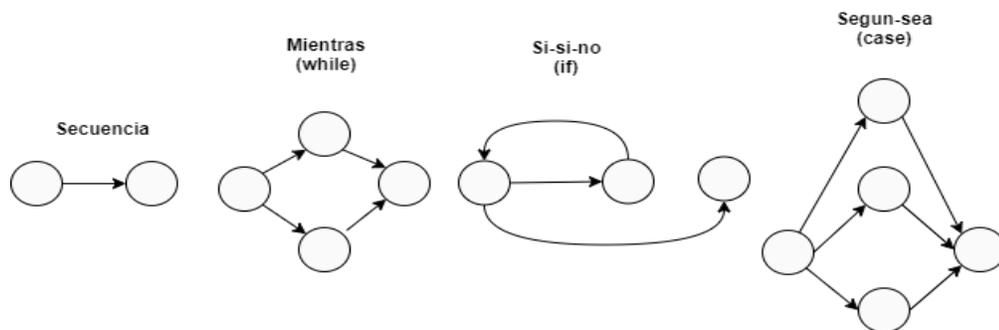
**Pruebas de ruta básica.** - Es una técnica que permite derivar una medida de complejidad lógica de un diseño de procedimiento y usar esta medida para definir un conjunto de rutas básicas de ejecución.

**Notación de grafico o grafo de flujo.** - En el gráfico de flujo de control lógico se muestra en cada circulo que es llamado nodo de grafico de flujo, representan una o más enunciados de procedimientos, las flechas que los unen son llamadas aristas o enlaces, representan el flujo de control. Una arista debe terminar en un nodo, las áreas acotadas por aristas y nodos se llaman regiones.

Se presentan los siguientes pasos para derivar la ruta básica:

- Dibujar el grafico de flujo a partir del diseño o código.
- Determinar la complejidad ciclomática del grafico de flujo resultante.
- Determinar un conjunto básico de rutas linealmente independiente.
- Preparar casos de prueba para forzar la ejecución de cada ruta en el conjunto básico.

Figura N.º 2. 15 Notación grafico de flujos



Fuente:(Pressman, 2010)

Con esta prueba se obtiene como resultados la disminución en un gran porcentaje el número de errores existentes en el software y por ende una mayor calidad y confiabilidad en la codificación.

### **2.8.2. Prueba de caja negra**

Las pruebas de caja negra, también denominadas pruebas de comportamientos, se centran en los requisitos funcionales del software. Analizando siempre los resultados que devuelve y probando todas las entradas en sus valores válidos e inválidos.

Las pruebas de caja negra intentan encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores de estructuras de datos o en acceso a base de datos externas.
- Errores de rendimiento.
- Errores de inicio y fin.

Al ejecutar las pruebas de Caja Negra se desarrollan casos de prueba reales para cada condición o combinación de condiciones y se analizan los resultados que arroja el sistema para cada uno de los casos. En esta estrategia se verifica el programa considerándolo una caja negra.

### **2.9. Métrica de calidad**

Se define que la calidad del producto software se puede tomar como el grado en que satisface los requisitos de sus usuarios, aportando de esta forma valor.

La ingeniería del software se diferencia de otras áreas, al no estar basada en leyes cuantitativas básicas, en su lugar se realiza un conjunto de medidas conocidas

como métricas, las cuales proporcionan una referencia de la calidad algún producto de software (Pressman, 2010).

Para el presente proyecto tomaremos en cuenta la como métrica de calidad la ISO/IEC 25010 hace parte de la familia ISO 25000. Es una norma que está centrada hacia la usabilidad, en el cual se determinan las características de calidad que se deben tener en cuenta en el momento de evaluar las propiedades de un producto software terminado.

### 2.9.1. ISO/IEC 25010

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las ocho características de calidad que se muestran en la siguiente figura (iso25000.com, 2019):

Figura N.º 2. 16 Organigrama de la norma ISO 25010



Fuente: (iso2500.com, 2019)

#### 2.9.1.1. Adecuación funcional

Representa la capacidad del producto software para proporcionar funciones que satisfacen las necesidades declaradas e implícitas, cuando el producto se usa en las condiciones especificadas. Esta característica también se subdivide en las siguientes subcaracterísticas:

- **Completitud funcional.** Grado en el cual el conjunto de funcionalidades cubre todas las tareas y los objetivos del usuario especificados.

- **Corrección funcional.** Capacidad del producto o sistema para proveer resultados correctos con el nivel de precisión requerido.
- **Pertinencia funcional.** Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de usuario especificados.

### 2.9.1.2. Eficiencia de desempeño

Representa el desempeño relativo a la cantidad de recursos utilizados bajo determinadas condiciones. Esta característica también se subdivide en las siguientes subcaracterísticas:

- **Comportamiento temporal.** Los tiempos de respuesta y procesamiento y las ratios de throughput de un sistema cuando lleva a cabo sus funciones bajo condiciones determinadas en relación con un banco de pruebas (benchmark) establecido.
- **Utilización de recursos.** Las cantidades y tipos de recursos utilizados cuando el software lleva a cabo su función bajo condiciones determinadas.
- **Capacidad.** Grado en que los límites máximos de un parámetro de un producto o sistema software cumplen con los requisitos.

### 2.9.1.3. Compatibilidad

Dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software. Esta característica también se subdivide en las siguientes subcaracterísticas:

- **Coexistencia.** Capacidad del producto para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes sin detrimento.

- **Interoperabilidad.** Capacidad de dos o más sistemas o componentes para intercambiar información y utilizar la información intercambiada.

#### 2.9.1.4. Usabilidad

Capacidad del producto software para ser entendido, aprendido, usado y resultar atractivo para el usuario, cuando se usa bajo determinadas condiciones. Esta característica también se subdivide en las siguientes subcaracterísticas:

- **Capacidad para reconocer su adecuación.** Capacidad del producto que permite al usuario entender si el software es adecuado para sus necesidades.
- **Capacidad de aprendizaje.** Capacidad del producto que permite al usuario aprender su aplicación.
- **Capacidad para ser usado.** Capacidad del producto que permite al usuario operarlo y controlarlo con facilidad.
- **Protección contra errores de usuario.** Capacidad del sistema para proteger a los usuarios de hacer errores.
- **Estética de la interfaz de usuario.** Capacidad de la interfaz de usuario de agrandar y satisfacer la interacción con el usuario.
- **Accesibilidad.** Capacidad del producto que permite que sea utilizado por usuarios con determinadas características y discapacidades.

#### 2.9.1.5. Fiabilidad

Capacidad de un sistema o componente para desempeñar las funciones especificadas, cuando se usa bajo unas condiciones y periodo de tiempo determinados. Esta característica también se subdivide en las siguientes subcaracterísticas:

- **Madurez.** Capacidad del sistema para satisfacer las necesidades de fiabilidad en condiciones normales.

- **Disponibilidad.** Capacidad del sistema o componente de estar operativo y accesible para su uso cuando se requiere.
- **Tolerancia a fallos.** Capacidad del sistema o componente para operar según lo previsto en presencia de fallos hardware o software.
- **Capacidad de recuperación.** Capacidad del producto software para recuperar los datos directamente afectados y reestablecer el estado deseado del sistema en caso de interrupción o fallo.

#### 2.9.1.6. Seguridad

Capacidad de protección de la información y los datos de manera que personas o sistemas no autorizados no puedan leerlos o modificarlos. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Confidencialidad.** Capacidad de protección contra el acceso de datos e información no autorizados, ya sea accidental o deliberadamente.
- **Integridad.** Capacidad del sistema o componente para prevenir accesos o modificaciones no autorizados a datos o programas de ordenador.
- **No repudio.** Capacidad de demostrar las acciones o eventos que han tenido lugar, de manera que dichas acciones o eventos no puedan ser repudiados posteriormente.
- **Responsabilidad.** Capacidad de rastrear de forma inequívoca las acciones de una entidad.
- **Autenticidad.** Capacidad de demostrar la identidad de un sujeto o un recurso.

#### 2.9.1.7. Mantenibilidad

Esta característica representa la capacidad del producto software para ser modificado efectiva y eficientemente, debido a necesidades evolutivas, correctivas

o perfectivas. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Modularidad.** Capacidad de un sistema o programa de ordenador (compuesto de componentes discretos) que permite que un cambio en un componente tenga un impacto mínimo en los demás.
- **Reusabilidad.** Capacidad de un activo que permite que sea utilizado en más de un sistema software o en la construcción de otros activos.
- **Analizabilidad.** Facilidad con la que se puede evaluar el impacto de un determinado cambio sobre el resto del software, diagnosticar las deficiencias o causas de fallos en el software, o identificar las partes a modificar.
- **Capacidad para ser modificado.** Capacidad del producto que permite que sea modificado de forma efectiva y eficiente sin introducir defectos o degradar el desempeño.
- **Capacidad para ser probado.** Facilidad con la que se pueden establecer criterios de prueba para un sistema o componente y con la que se pueden llevar a cabo las pruebas para determinar si se cumplen dichos criterios.

#### **2.9.1.8. Portabilidad**

Capacidad del producto o componente de ser transferido de forma efectiva y eficiente de un entorno hardware, software, operacional o de utilización a otro. Esta característica se subdivide a su vez en las siguientes subcaracterísticas:

- **Adaptabilidad.** Capacidad del producto que le permite ser adaptado de forma efectiva y eficiente a diferentes entornos determinados de hardware, software, operacionales o de uso.
- **Capacidad para ser instalado.** Facilidad con la que el producto se puede instalar y/o desinstalar de forma exitosa en un determinado entorno.

- **Capacidad para ser reemplazado.** Capacidad del producto para ser utilizado en lugar de otro producto software determinado con el mismo propósito y en el mismo entorno.

En este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado. (iso25000.com, 2019)

## 2.10. Estimación de costos de software

### 2.10.1. Cocomo II

El modelo COCOMO original, evoluciono hacia un modelo más exhaustivo, llamado COCOMO II, el cual permite realizar estimaciones en función del tamaño de software, y de conjunto de factores de costo y de escala. Los factores de costo incluyen aspectos relacionados con la naturaleza del sistema, equipo, y características propias del proyecto, los factores de escala incluyen la parte de escala producida a medida que un proyecto de software incrementa su tamaño.

El modelo de Cocomo II tiene 3 ecuaciones que son las siguientes:

$$E = a (KLDC)^b; \text{ Persona – Mes}$$

Ecuación 1

$$D = c (E)^d; \text{ Meses}$$

Ecuación 2

$$P = E / D; \text{ Personas}$$

Ecuación 3

Dónde:

E: Esfuerzo requerido por el proyecto expresado en persona-mes.

D: Tiempo requerido por el proyecto expresado en meses.

P: Número de personas requeridas para el proyecto.

A, b, c y d: Constantes con valores definidos según cada sub-modelo.

KLDC: Cantidad de líneas de código distribuidas en miles.

A la vez cada modelo se subdivide en modos, los mismos son (Calero, 2010).

- **Modo orgánico:** es un pequeño grupo de programadores experimentados desarrollando proyectos de software en un entorno familiar. El tamaño del software varía desde unos pocos miles de líneas (tamaño pequeño) a unas docenas de miles (tamaño medio).
- **Modo semi - libre o semi acoplado:** Corresponde a un esquema intermedio entre modo orgánico y el rígido, el grupo de desarrollo puede incluir una mezcla de personas experimentadas y no experimentadas.
- **Modo rígido o empotrado:** El proyecto tiene fuertes restricciones, que pueden estar relacionadas con la funcionalidad y/o pueden ser técnicas. El problema a resolver es único, siendo difícil basarse en la experiencia puesto que puede no haberla.

Tabla N.º 2. 4 Constante de Coste Modelo Básico

Proyecto de software	A	B	C	D
Orgánico	2.4	1.05	1.05	0.38
Semi-Acoplado	3.0	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

Fuente: (Calero, 2010)

## **3.MARCO APLICATIVO**

### 3.1. Introducción

En este capítulo se desarrolla el “Sistema Web de Gestión Administrativa y Seguimiento Contable” siguiendo la metodología de desarrollo Scrumban, para obtener la representación conceptual y física del sistema en el proceso de la elaboración.

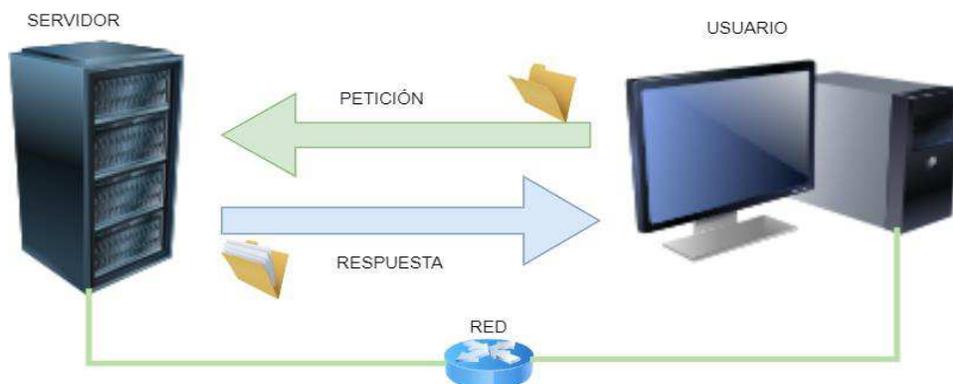
Se puede apreciar en la siguiente figura el modelo de procesos de desarrollo que se utilizó en el presente proyecto, la combinación de la metodología Scrumban la cual es importante para la implementación del producto final como se puede observar en la Figura N° 2.5 de la página 20.

### 3.2. Arquitectura y requerimiento de software

#### 3.2.1. Arquitectura

La arquitectura del software funciona de la siguiente forma en la interacción entre el servidor y el usuario encargado para poder realizar los debidos trabajos en el sistema.

Figura N.º 3. 1 Arquitectura del software



Fuente: Elaboración propia

### 3.2.2 Requerimiento de software

Para el mejor entendimiento del software para los requerimientos del cliente se utilizaron los diagramas de casos de uso que se muestran en la Figura N° 3.3 pagina 56.

### 3.3. Roles Scrumban

Se muestra en la siguiente tabla los roles del equipo de trabajo.

Tabla N.º 3. 1 Roles del equipo de trabajo

Roles	Descripción	Trabajos	Responsable
RP	Responsable del proyecto	Encargado	David Condori Callata
TEAM	Equipo de desarrollo	Análisis y diseño	David Condori Callata
		Programación	David Condori Callata
		Implementación y pruebas	David Condori Callata
USER	Cliente	Gerente General de C. T. I.	Lic. Marycel J. Quisbert Torrez

Fuente: Elaboración propia

### 3.4. Fase 1 lista de requisitos

Para poder seleccionar las tareas los primeros pasos que se debe realizar es definir las listas de requisitos del sistema a priorizar, para esto fue necesario realizar reuniones con el cliente, que en este caso la función del cliente la cumplió la Dueña y Gerente general de la empresa Construcción Innovación & Tecnología.

Para para obtención de la lista de requisitos se realizó diversas entrevistas personales con la Gerente general, preguntando el proceso y las etapas que se deben seguir para llevar los diferentes procesos de registro en la parte de contabilidad, tipos de cuentas que existen y repostes o búsquedas a realizar, ventas, compras, respaldo de documentos, cuantos tipos de usuarios existen la cual poder administrar usuarios y roles.

Viendo todos los detalles que el cliente proporciono se hizo un análisis de toda la información recolectada se elabora una tabla con los requisitos que son descrita en la siguiente tabla:

Tabla N.º 3. 2 Lista de Requisitos del proyecto

<b>LISTA DE REQUISITOS PRIORIZADA</b>					
Nº	Nombre	Modulo	Prioridad	Tiempo de desarrollo	Como probar
1	Diseño de la base de datos		Demasiada Alta	7 Días	Implementar en el servidor.
2	Autenticación de Usuario	Modulo Usuario	Muy Alta	5 Días	Probar la autenticación de usuario con los datos correctos Login al menú principal.
3	Acciones de Usuario		Muy Alta	3 Días	Verificar las diferentes URLs de acceso y el CRUD de usuarios.
4	Roles de Usuarios		Muy Alta	3 Días	Probar los diferentes roles de asignación a cada usuario.
5	Crear registro de plan de cuentas		Modulo Plan de Cuentas	Alta	4 Días
6	Crear registros de Libro Diario	Modulo Libro Diario	Alto	5 Días	Probar las vistas y CRUD necesarios y recuperación de datos desde el módulo de plan de cuentas.
7	Crear acciones de libro diario		Alta	4 Días	Generar los reportes con los datos necesario.
8	Generar libro mayor	Modulo Libro Mayor	Alta	5 Días	Generar el libro mayor según solicitado.
9	Generar Balance	Modulo Balance General	Alta	5 Días	Verificar el reporte del balance general con los datos necesarios que son solicitados desde libro diario.
10	Crear registro de productos	Módulo de Productos	Alta	2 Días	Probar los registros necesarios.
11	Crear acciones para productos		Alta	4 Días	Crear las rutas y CRUD necesarios y todas las acciones.
12	Crear registro de categorías	Módulo de Categorías	Alta	2 Días	Verificar tipos de registros que se realicen y sus acciones
13	Acciones a realizar en categorías		Alta	2 Días	Probar las diferentes acciones que tendrá el CRUD.

14	Crear registro de Proveedores	Módulo de Proveedores	Alta	2 Días	Verificar tipo de registros del proveedor
15	Acciones a realizar en proveedores		Alta	3 Días	Probar las diferentes acciones del CRUD
16	Crear registro de clientes	Módulo de Clientes	Alta	2 Días	Probar el formato Según lo requerido con los datos específicos.
17	Acciones a realizar en clientes		Alto	3 Días	Probar las diferentes funciones del CRUD.
18	Crear registrar de datos de las compras	Módulo de Compras	Alta	3 Días	Probar los tipos de datos a registrar
19	Acciones a realizar en compras		Alta	5 Días	Probar las acciones del CRUD y reportes a generar y comprobantes de compra
20	Crear registro de ventas	Módulo de Ventas	Alto	4 Días	Probar todo lo que se registrar en las ventas
21	Crear acciones a realizar		Alto	4 Días	Probar las diferentes acciones CRUD y reportes a genera tanto la facturación
22	Crear la generación de facturas o recibos	Generar Facturas o Recibos	Alta	4 días	Probar la correcta generación de las facturas de ventas
23	Crear registro de archivos de respaldo	Módulo de Archivos	Media	2 Días	Probar el almacenamiento de diferentes archivos como ser Word, Excel, PDF e imágenes.
24	Crear categoría de archivos y descargas		Baja	1 Días	Probar el almacenamiento de archivos en diferentes fólderes y descargar los diversos archivos.

Fuente: Elaboración propia

### 3.4.1. Requisitos de hardware y software de desarrollo

Para el desarrollo del proyecto se necesita tanto hardware como software, en la cual se irán realizando la implementación y debidas pruebas también diferentes herramientas que nos ayudarán en el desarrollo.

Tabla N.º 3. 3 Requisitos de Software y Hardware mínimo para el Desarrollo

Hardware	Software
PC de escritorio o laptop con las siguientes características: <ul style="list-style-type: none"> <li>• RAM 2 GB o superior</li> <li>• Disco duro Mínimo de 50 GB</li> <li>• Procesador AMD, Core i3 o Superior</li> <li>• Monito</li> <li>• Teclado</li> <li>• Mouse</li> <li>• Impresora</li> </ul>	<ul style="list-style-type: none"> <li>• Sistema Operativo Windows o Linux</li> </ul> Programas necesarios: <ul style="list-style-type: none"> <li>• MySQL (Para el servidor donde se realiza las pruebas de la base de datos)</li> <li>• Draw.io (Para la creación de diagramas)</li> <li>• Sublime o Atom (Editor de código de múltiples Lenguajes de Programación)</li> </ul>

Fuente: Elaboración propia

### 3.5. Fase 2 descripción de las tarjetas de tareas

Según el modelo que se propuso que es la metodología Scrumban para la elaboración de este proyecto, en esta fase se desarrollan las tarjetas las cuales son seleccionados de la lista de requisitos según su prioridad.

Las prioridades se definirían de forma numérica de la siguiente manera:

- Prioridad Demasiada Alta = 1
- Prioridad Muy Alta = 2
- Prioridad Alta = 3
- Prioridad Media = 4
- Prioridad Baja = 5

#### 3.5.1. Tarjetas de tareas

Se muestran las siguientes tareas según su prioridad y cuando cada tarjeta se considera completada.

Figura N.º 3. 2 Tarjetas de Tareas



Tarjeta # 7

Crear acciones de libro diario

Programar las diferentes acciones que tendra el registro de libros diarios.

Completo:

Se completa cuando las acciones necesarias son verificadas y probadas con éxito

PRIORIDAD 3

Tarjeta # 8

Generar libro mayor

Programar la generacion de adecuada del balance general para el reporte.

Completo:

Se completa cuando todos los reportes generados del libro mayor se cumplan.

PRIORIDAD 3

Tarjeta # 9

Generar balance

Programar las diferentes acciones que se debe realizar para el generar el balance general.

Completo:

Se completa cuando la generacion de los balances generales sean según lo deseado

PRIORIDAD 3

Tarjeta # 10

Crear registros de productos

Programar todos los registros necesarios

Completo:

Se completa cuando todos los datos necesarios se puedan registrar.

PRIORIDAD 3

Tarjeta # 11

Crear acciones para el producto

Programar las diferentes acciones necesarias

Completo:

Se completa cuando el todas las acciones necesarias son probados y son según la necesidad del modulo

PRIORIDAD 3

Tarjeta # 12

Crear registro de categorías

Programar la forma de registro de las diversas categorías que se podrá registrar

Completo:

Se completa cuando las diferentes categorías se puedan registrar correctamente

PRIORIDAD 3

Tarjeta # 13

Acciones a realizar en categoría

Programar las diferentes acciones segun los requerido.

Completo:

Se completa cuando todas las acciones requeridas funcionen adecuadamente

PRIORIDAD 3

Tarjeta # 14

Crear registro de proveedor

Programar que tipo de datos se registran

Completo:

Se completa cuando el el registro de datos sea exitoso con los datos necesarios

PRIORIDAD 3

Tarjeta # 15

Acciones a realizar en proveedores

Programar todas las acciones solicitadas que del CRUD

Completo:

Se completa cuando todas las acciones funcionen debidamente

PRIORIDAD 3

Tarjeta # 16

Crear registro de clientes

Programar los diferentes registros requeridos para el registro de clientes

Completo:

Se completa cuando los registros sean exitosos

PRIORIDAD 3

Tarjeta # 17

Acciones a realizar en clientes

Programar las acciones necesarias en que se tendra en clientes

Completo:

Se completa cuando todas las acciones requeridas cumplan las funciones que se debe realizar

PRIORIDAD 3

Tarjeta # 18

Crear registros de datos de las compras

Programar la forma de registrar de los datos de todas las compras a realizar

Completo:

Se completa cuando todas las formas de registros sean exitosos y correctamente las compras

PRIORIDAD 3

<p>Tarjeta # 19</p> <p><u>Acciones a realizar en compras</u></p> <p>Programar todas las acciones que tendra el modulo de compras</p> <p><u>Completo:</u></p> <p>Se completa cuando todos acciones necesarias que se debe emplear en las compras exitosamente</p> <p>PRIORIDAD 3</p>	<p>Tarjeta # 20</p> <p><u>Crear registro de ventas</u></p> <p>Programar las direfentes formas de registros requeridas para las diversas ventas que se realizarán</p> <p><u>Completo:</u></p> <p>Se completa cuando los registros de ventasn sean exitosos segun la necesidad del cliente</p> <p>PRIORIDAD 3</p>
<p>Tarjeta # 21</p> <p><u>Crear acciones de ventas</u></p> <p>Programar las diferentes acciones que se emplearan en las ventas</p> <p><u>Completo:</u></p> <p>Se completa cuando todas las acciones requeridas en ventas se cumplan</p> <p>PRIORIDAD 3</p>	<p>Tarjeta # 22</p> <p><u>Crear la generación de facturas o recibos</u></p> <p>Programar todo lo necesario para la generación de la factura o recibo y el formato en que se genera</p> <p><u>Completo:</u></p> <p>Se completa cuando la factura se generá con todos los datos adecuados según lo requerido para el proceso de compras tanto en ventas</p> <p>PRIORIDAD 3</p>
<p>Tarjeta # 23</p> <p><u>Crear registros de archivos de respaldo</u></p> <p>Programar la forma en que se respaldara los diferentes formatos de archivos</p> <p><u>Completo:</u></p> <p>Se completa cuando los diferentes formatos de archivos se puedan guardar</p> <p>PRIORIDAD 4</p>	<p>Tarjeta # 24</p> <p><u>Crear categoria de archivos y descargas</u></p> <p>Programar la forma de como se podra descargar los archivos almacenados</p> <p><u>Completo:</u></p> <p>Se completa cuando se pueda descargar los archivos guardados</p> <p>PRIORIDAD 5</p>

Fuente: Elaboración propia

### 3.6. Fase 3 Análisis y diseño - Base de datos

#### 3.6.1. Descripciones de los actores

Para comprender Figura N.º 3. 3 pagina 56 diagrama de caso de uso general, se detalla un listado de roles y tareas que los usuarios desarrollan en el sistema las cuales son descritas en las siguientes tablas.

Tabla N.º 3. 4 Descripción de los actores del sistema

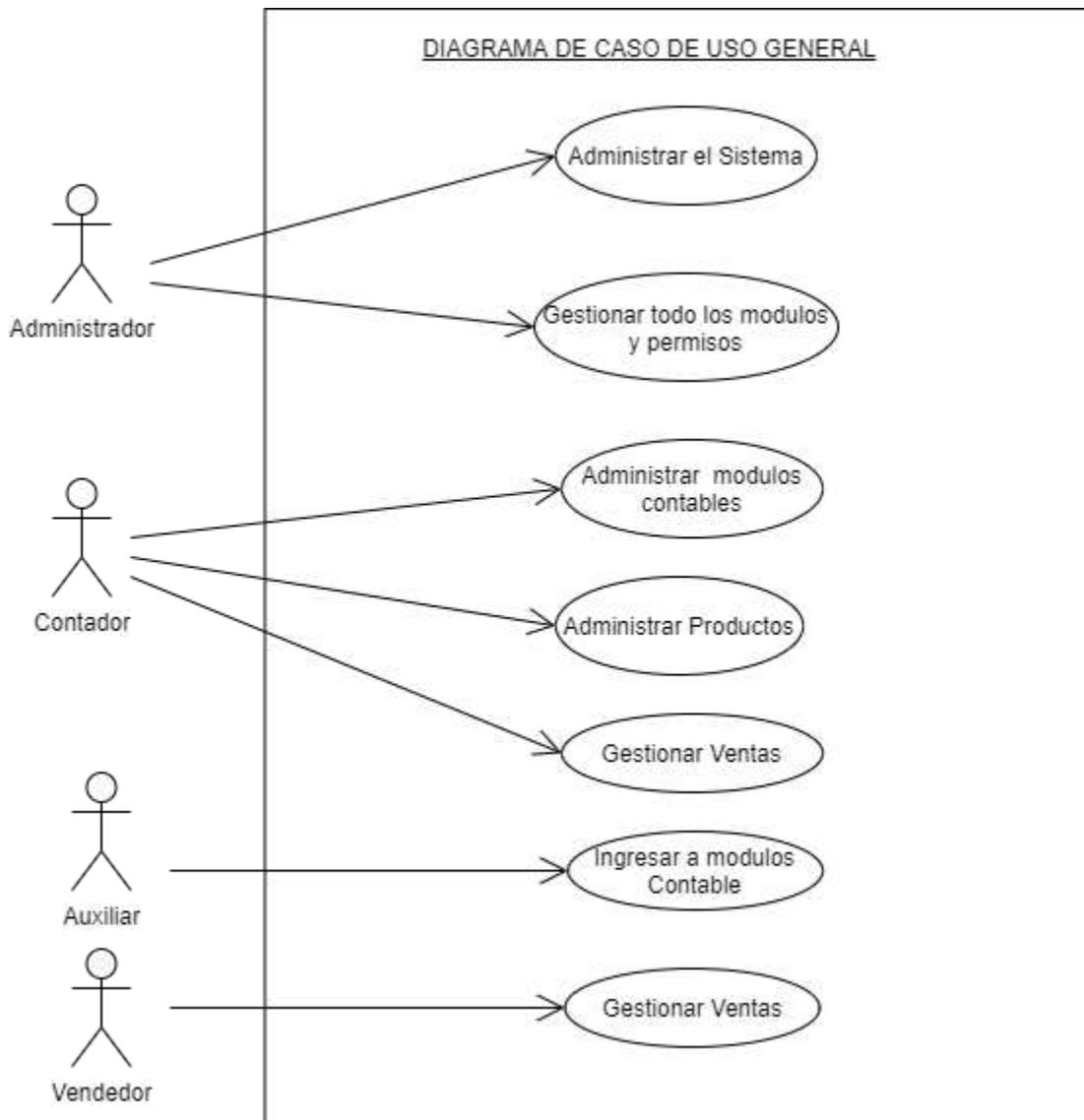
ACTOR	TAREAS
ADMINISTRADOR	Administrar a los usuarios. Autoriza los roles a los usuarios. Puede crear, ver, actualizar y eliminar cada dato de los módulos existentes en el sistema. Generar todo el reporte existente. Administrar todos los módulos contables como libro diario, libro mayor, plan de cuentas balance general y generar reportes. Administrar todos los módulos existentes en productos. Gestionar compras, ventas, proveedores, clientes, productos, categorías y todas las facturaciones o recibos. Manejo de archivos de respaldados y permisos a ellos.
CONTADOR	Puede realizar reportes, ver, editar, eliminar en los módulos autorizados. Administrar plan de cuentas. Administrar libros diarios y generar reportes. Administrar libros mayores. Administrar balance general. Administrar módulo de productos, categorías, ventas, compras, proveedores, clientes. Emite Factura/Recibos. Manejo de archivos de respaldo.
AUXILIAR	Ingresar a los módulos de libro diario y libro mayor. Generar reportes autorizados. Realizar registro de libro diario.
VENDEDOR	Puede realizar reportes y emitir facturas o recibos. Administrar módulo de ventas y clientes. Podrá ingresar al módulo de productos y generar reporte de stock.

Fuente: Elaboración Propia

### 3.6.2. Diagrama de caso de uso general

Para comprender el funcionamiento del sistema se utiliza el diagrama de caso de uso de alto nivel la cual nos permitirá observar ordenadamente las funciones que se realizan.

Figura N.º 3. 3 Diagrama de Caso de Uso General



Fuente: Elaboración propia

### 3.6.3. Diseño de base de datos e implementación

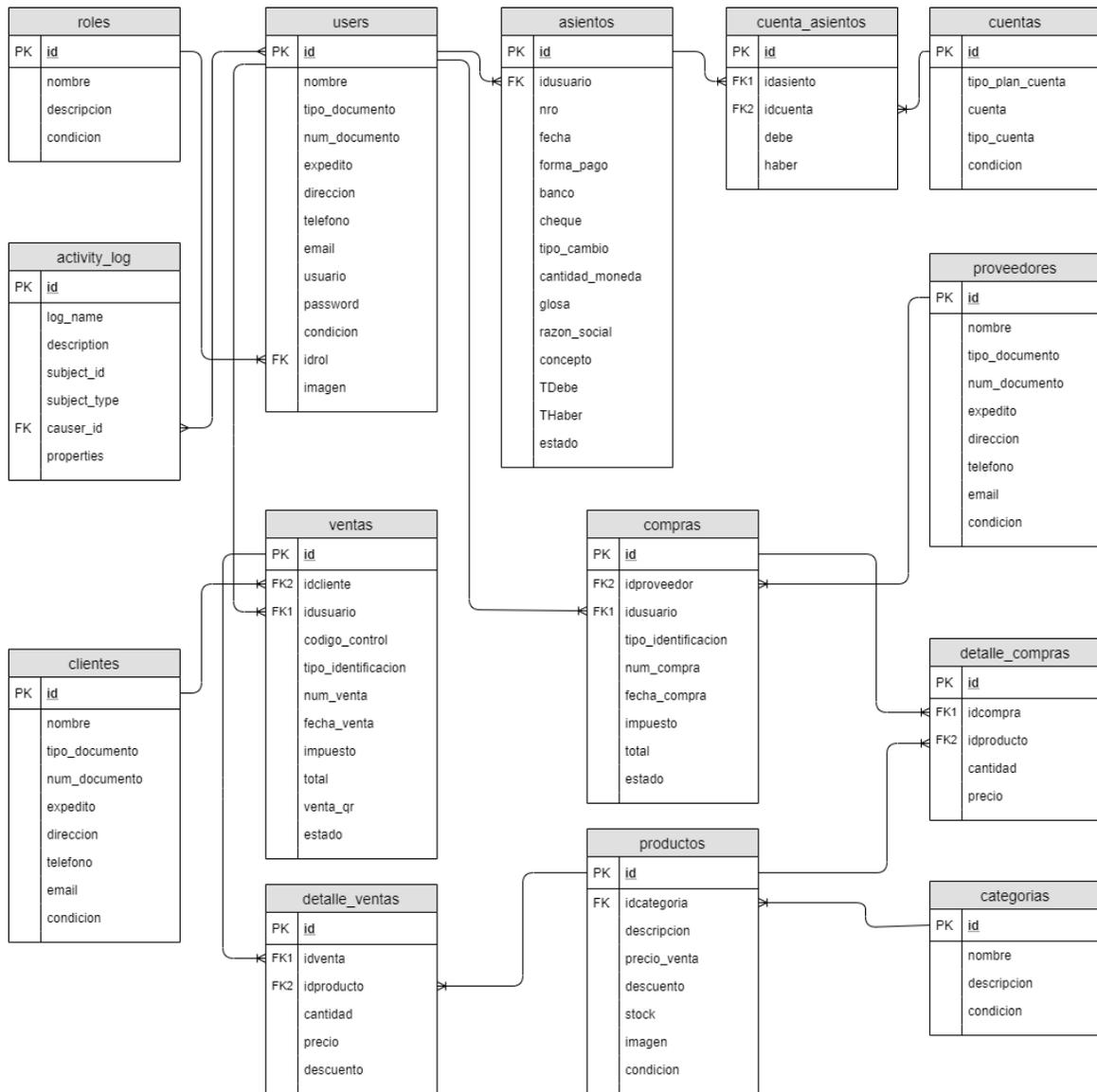
En esta parte de la etapa se realizó la tarea de diseñar la Base de Datos e implementación. En la cual se diseñó el modelo relacional de la Base de Datos pensando en cumplir todos los requerimientos.

Tabla N.º 3. 5 Modelo relacional de la Base de Datos

REQUISITO	TAREA	Inicio	Fin	Duración
		11/02/2020	17/02/2020	7 días
		Desde	Hasta	Estado
Diseño de la base de datos	Diseño del modelo relacional de la base de datos e Implementación	11/02/2020	17/02/2020	Completado

Fuente: Elaboración propia

Figura N.º 3. 4 Modelo relacional de la Base de Datos



Fuente: Elaboración propia

### 3.7. Fase 4 Programación – Base de datos

Después de diseñar el modelo relacional se realiza la programación de la base de datos y el diccionario de datos correspondiente, la cual contiene nombre de las tablas, atributos, tipo de dato y descripción de los atributos. Se puede observar en la siguiente tabla:

Tabla N.º 3. 6 Diccionario de datos de la Base de Datos

TABLAS	ATRIBUTOS	TIPO	DESCRIPCIÓN
roles	Id	INT(10)	Numero único del rol
	nombre	VARCHAR(30)	Nombre del rol
	descripcion	VARCHAR(100)	Descripción breve del rol
	condicion	TINYINT(1)	Estado de rol activo – no activo
activity_log(acciones)	id	INT(10)	Numero único de acción
	log_name	VARCHAR(191)	Nombre de registro
	description	TEXT	Descripción de acción que realiza
	subject_id	INT(11)	Numero único del modificado
	subject_type	VARCHAR(191)	URL donde fue causado
	causer_id	INT(11)	Numero único de quien modifico
	causer_type	VARCHAR(191)	Registro de quien modifico
	properties	TEXT	Registro de datos modificados
users	Id	BIGINT(20)	Numero único del usuario
	nombre	VARCHAR(100)	Nombre completo de usuario
	tipo_documento	VARCHAR(20)	Tipo de documento del usuario
	num_documento	VARCHAR(20)	Numero de documento del usuario
	expedito	VARCHAR(30)	Lugar de emisión del documento
	direccion	VARCHAR(70)	Dirección del usuario
	telefono	VARCHAR(20)	Teléfono del usuario
	Email	VARCHAR(50)	Correo electrónico del usuario
	usuario	VARCHAR(50)	Nombre de usuario para iniciar sesión
	password	VARCHAR(30)	Contraseña del usuario para iniciar sesión
	condicion	TINYINT(1)	Estado del usuario activo – no activo
	Idrol	INT(10)	Numero de rol asignado al usuario
	imagen	VARCHAR(200)	Foto del usuario
asientos	Id	BIGINT(20)	Numero único de asientos del libro diario
	idusuario	BIGINT(20)	Numero único de usuario que registra el libro diario
	Nro	BIGINT(20)	Numero de comprobante
	Fecha	DATE	Fecha de registro que corresponde
	forma_pago	VARCHAR(100)	Forma de pago
	Banco	VARCHAR(50)	Registro al banco que corresponde
	cheque	BIGINT(20)	Numero de cheque
	tipo_cambio	VARCHAR(100)	Tipo de cambio de la moneda
	tantidad_moneda	DECIMAL(12,2)	Cantidad de cambio en Bs

	Glosa	VARCHAR(200)	Glosa nota de descripción del comprobante.
	razon_social	VARCHAR(200)	Registrar nombre de la empresa
	concepto	VARCHAR(200)	Registrar el concepto del comprobante
	TDebe	DECIMAL(12,2)	Total, del debe
	THaber	DECIMAL(12,2)	Total, del haber
	estado	VARCHAR(20)	Estado de registrado o anulado
cuenta_asientos	Id	BIGINT(20)	Numero único de la tabla
	idasiento	BIGINT(20)	Numero único del asiento
	idcuenta	BIGINT(20)	Numero único de la cuenta
	Debe	DECIMAL(12,2)	Él debe del libro diario
	Haber	DECIMAL(12,2)	El haber del libro diario
cuentas	Id	BIGINT(20)	Numero único de la tabla de cuentas
	tipo_plan_cuenta	VARCHAR(100)	Tipo de cuanta activo, pasivo, patrimonio.
	cuenta	BIGINT(20)	Número cuenta
	tipo_cuenta	VARCHAR(100)	Nombre de la cuenta
	condicion	TINYINT(1)	Estado lógico activo – no activo
compras	Id	INT(10)	Numero único de la tabla
	idproveedor	INT(10)	Numero único del proveedor
	idusuario	BIGINT(20)	Numero único del usuario
	tipo_identificacion	VARCHAR(20)	Tipo de identificación del documento de compra
	num_compra	VARCHAR(10)	Número de la compra
	fecha_compra	DATETIME	Fecha y hora del registro de la compra
	impuesto	DECIMAL(4,2)	Registra el impuesto IVA
	Total	DECIMAL(11,2)	Total, a pagar de las compras
	estado	VARCHAR(20)	Estado de compra registrado o anulado
proveedores	Id	INT(10)	Numero único de la tabla
	nombre	VARCHAR(100)	Nombre completo
	tipo_documento	VARCHAR(20)	Tipo de documento
	num_documento	VARCHAR(20)	Numero de documento
	expedito	VARCHAR(30)	Lugar de emisión del documento
	direccion	VARCHAR(70)	Dirección
	telefono	VARCHAR(20)	Teléfono
	Email	VARCHAR(50)	Correo electrónico
	condicion	TINYINT(1)	Estado activo – no activo
	Id	INT(10)	Numero único de la tabla

detalle_compras	idcompra	INT(10)	Numero único de la tabla compras
	idproducto	INT(10)	Numero único de la tabla producto
	cantidad	INT(11)	Cantidad
	Precio	DECIMAL(11,2)	Precio
productos	Id	INT(10)	Numero único de la tabla
	idcategoria	INT(10)	Numero único de la categoría
	Código	VARCHAR(50)	Código del producto
	nombre	VARCHAR(100)	Nombre del producto
	descripcion	VARCHAR(100)	Descripción del producto
	precio_venta	DECIMAL(11,2)	Precio de venta
	descuento	INT(11)	Descuento en porcentaje
	Stock	INT(11)	Cantidad de producto existentes
	imagen	VARCHAR(200)	Foto del producto
	condicion	TINYINT(1)	Estado activo – no activo
categorias	Id	INT(11)	Numero único de la tabla
	nodmbre	VARCHAR(50)	Nombre de categoría
	descripcion	VARCHAR(256)	Descripción de categoría
	condicion	TINYINT(1)	Estado activo – no activo
detalle_ventas	Id	INT(10)	Numero único de la tabla
	idventa	INT(10)	Numero único de venta
	idproducto	INT(10)	Numero único de producto
	cantidad	INT(11)	Cantidad de productos
	Precio	DECIMAL(11,2)	Precio unitario del producto
	descuento	DECIMAL(4,2)	Descuento unitario del producto
ventas	Id	INT(10)	Numero único de la tabla
	idcliente	INT(10)	Numero único del cliente
	idusuario	BIGINT(20)	Numero único del usuario
	codigo_control	VARCHAR(20)	Código de control de la venta
	tipo_identificacion	VARCHAR(20)	Tipo de documento
	num_venta	VARCHAR(10)	Numero de venta
	fecha_venta	DATETIME	Fecha de la venta
	impuesto	DECIMAL(4,2)	Impuesto IVA
	total	DECIMAL(11,2)	Total, a pagar
	venta_qr	VARCHAR(200)	Código QR de la venta
	estado	VARCHAR(20)	Estado de la venta registrado o anulado
clientes	Id	INT(10)	Numero único de la tabla
	nombre	VARCHAR(100)	Nombre completo del cliente
	tipo_documento	VARCHAR(20)	Tipo de documento
	num_documento	VARCHAR(20)	Numero de documento
	expedito	VARCHAR(30)	Lugar de emisión del documento
	direccion	VARCHAR(70)	Dirección

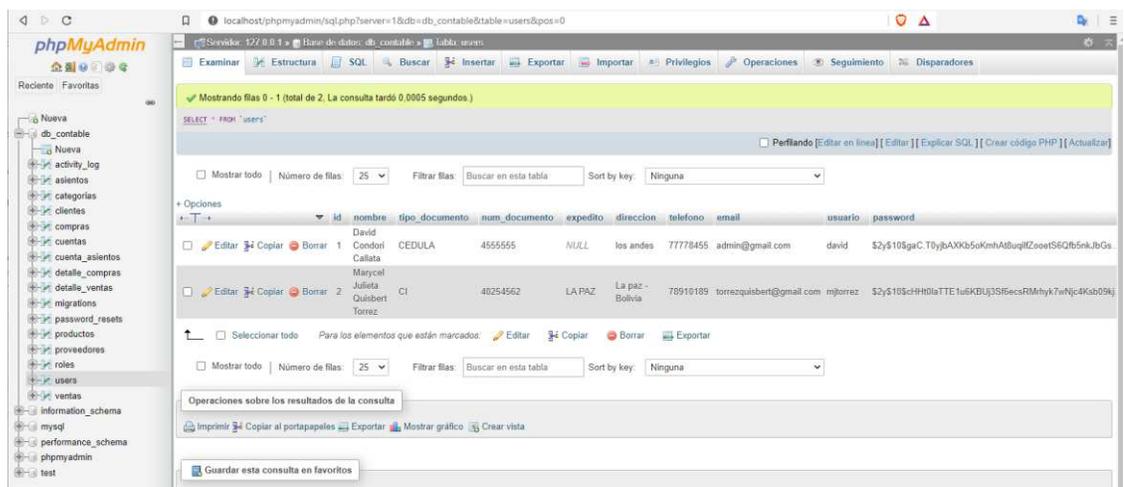
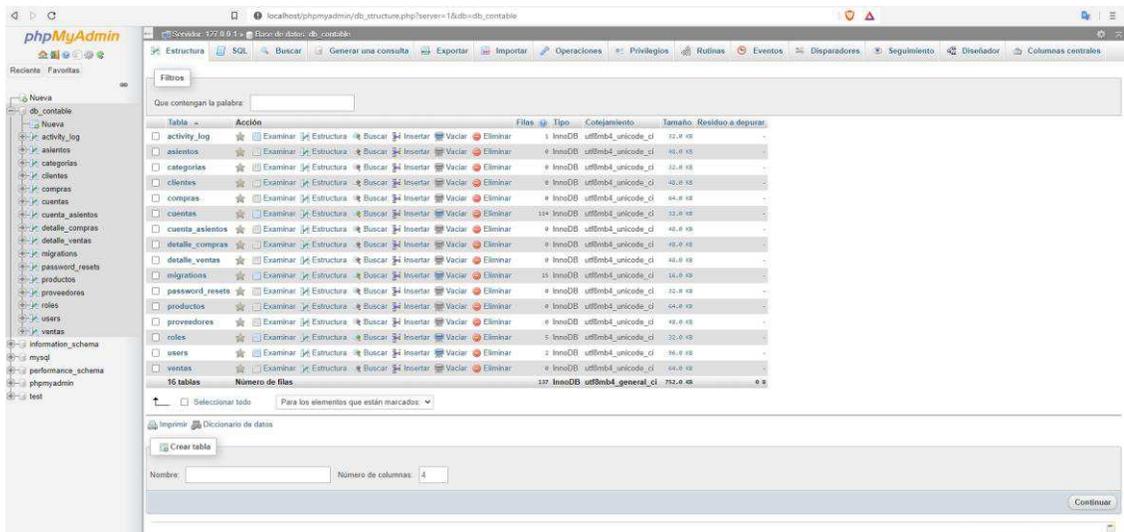
telefono	VARCHAR(20)	Teléfono
email	VARCHAR(50)	Correo electrónico
condicion	TINYINT(1)	Estado activo – no activo

Fuente: Elaboración propia

### 3.8. Fase 5 Implementación y pruebas – Base de datos

Se realizó las pruebas de integración en base de datos implementada en el servidor de pruebas.

Figura N.º 3. 5 Prueba e implementación de Base de datos en el servidor



Fuente: Elaboración propia

### 3.8.1. Resultados

Se realizó las pruebas necesarias en el servidor de pruebas de desarrollo.

Tabla N.º 3. 7 Resultados de la implementación de la Base de Datos

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	Crear: ¿cuándo un usuario guarda la información en la base de datos cumple?	Cumple
2	Recuperar: ¿Cuándo el usuario realiza la búsqueda de la información almacenada en la base de datos cumple?	Cumple
3	Actualizar: ¿Cuándo un usuario edita o modifica un registro la información se guarda con los nuevos cambios, cumple?	Cumple
4	Eliminar: ¿Cuándo un usuario elimina o borra una información en la base de datos cumple?	Cumple

Fuente: Elaboración propia

### 3.9. Fase 3 Análisis y diseño - Módulo usuario

Este es prioridad Muy Alto los cuales son: Autenticación de Usuario, Acciones de Usuarios y Roles de Usuarios que pertenecen al Módulo Usuario. Se diseñó los diagramas en la que nos proporciona UML para cada requerimiento, para luego poder codificarlo.

Tabla N.º 3. 8 Módulo de Usuario, seguridad y las acciones

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Autenticación de Usuario	Diseño del diagrama UML	18/02/2020	28/02/2020	11 días
	Codificación	20/02/2020	21/02/2020	Completado
	Pruebas	22/02/2020	22/02/2020	Completado
Acciones de Usuario	Diseño del diagrama UML	23/02/2020	23/02/2020	Completado
	Codificación	24/02/2020	24/02/2020	Completado
	Pruebas	25/02/2020	25/02/2020	Completado
Roles de usuario	Diseño del diagrama UML	26/02/2020	26/02/2020	Completado
	Codificación	27/02/2020	27/02/2020	Completado
	Pruebas	28/02/2020	28/02/2020	Completado

Fuente: Elaboración propia

### 3.9.1. Caso de Uso

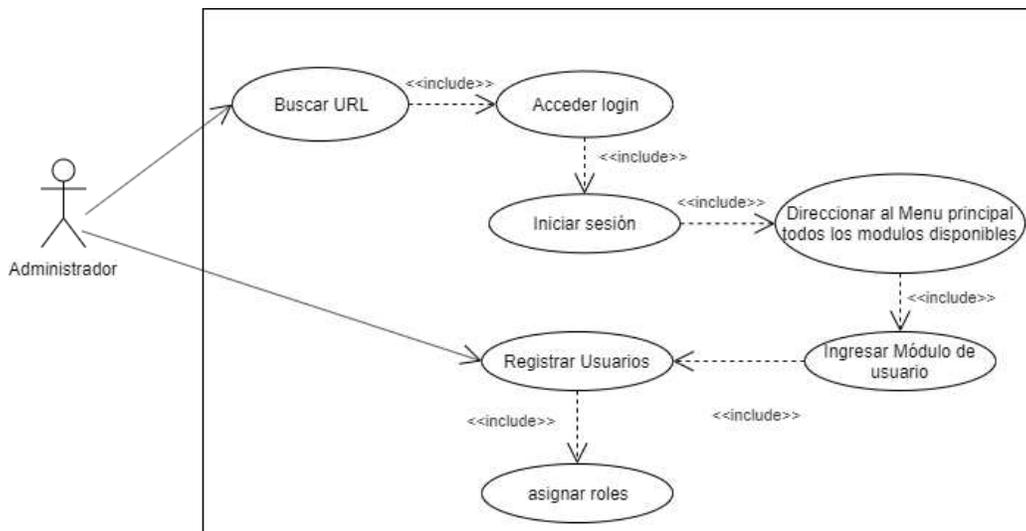
En la siguiente tabla se describe como se realiza el módulo.

Tabla N.º 3. 9 Descripción de caso de uso del Módulo Usuario en la seguridad

Nro.	Descripción
1	Para la autenticación se debe dirigir a la página principal del dominio a la opción login
2	Se autentica con los datos del usuario y se direcciona al menú principal
3	Asigna intervalos de tiempo en el acceso de usuario
4	Muestra todos los módulos disponibles para la administración ingresa al módulo usuario y registra usuarios con la debida asignación de roles y se registra las acciones.

Fuente: Elaboración propia

Figura N.º 3. 6 Caso de uso del Módulo Usuario en la seguridad



Fuente: Elaboración propia

Tabla N.º 3. 10 Inicio de sesión ingresa al módulo de usuario

<b>Nombre.</b>	<b>Iniciar sesión</b>
Actores	Administrador. Sistema.
Propósitos	Para el usuario pueda ingresar debe en navegador buscar la URL o dirección de la página web y dirigirse a la parte del login e iniciar la sesión con sus datos asignados y el sistema verifica que el usuario tengas los permisos para acceder, luego es direccionado al menú principal del sistema la cual el usuario podrá dirigirse y acceder al módulo de usuarios para los debidos registros. <ul style="list-style-type: none"> <li>• Evento del actor: Introduce el nombre de usuario y la contraseña</li> <li>• Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de usuario administrar y registra nuevos usuarios y se registra las acciones.</li> </ul>
Flujo principal	
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

Tabla N.º 3. 11 Usuario asignar roles, registro de usuarios

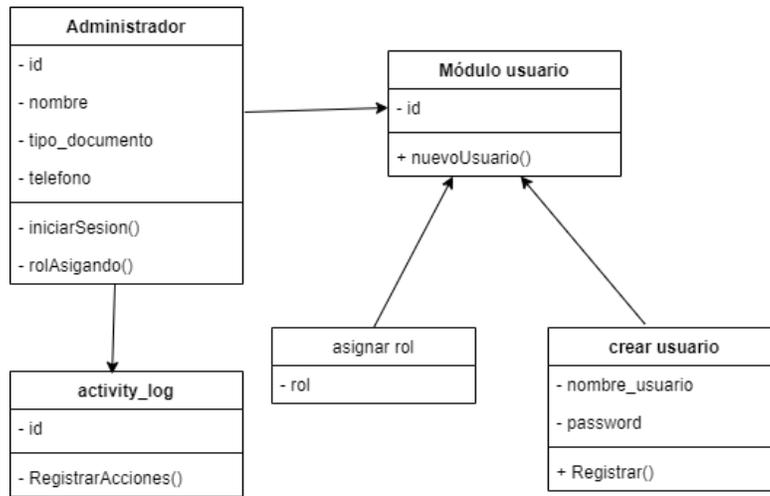
<b>Nombre.</b>	<b>Iniciar asignar roles / Registrar</b>
Actores	Administrador. Sistema.
Propósitos	Para que el usuario pueda ingresar se recupera los datos del tipo de rol asignado al usuario y puede crear nuevos usuarios designando roles a los nuevos. <ul style="list-style-type: none"> <li>• Evento del actor: El usuario deberá tener un rol asignado.</li> <li>• Evento del sistema: Recupera el rol asignado de la base de datos.</li> </ul>
Flujo principal	
Flujo alternativo	Si no se encuentra el rol del usuario no tendrá acceso al sistema.
Precondición	El usuario debe tener asignado un rol.
Postcondición	Solo con el rol autorizado podrá administrar las diferentes funciones del sistema.
Presunción	El administrador debe estar registrado en las bases de datos para poder ingresar.

Fuente: Elaboración propia

### 3.9.2. Diagrama de clases

Se diseña el diagrama de clases para el requisito para el Autenticación de Usuario, mostrar acciones del usuario y rol el cual se visualiza en la siguiente figura:

Figura N.º 3. 7 Diagrama de clases Modelo Usuario y acceso de seguridad

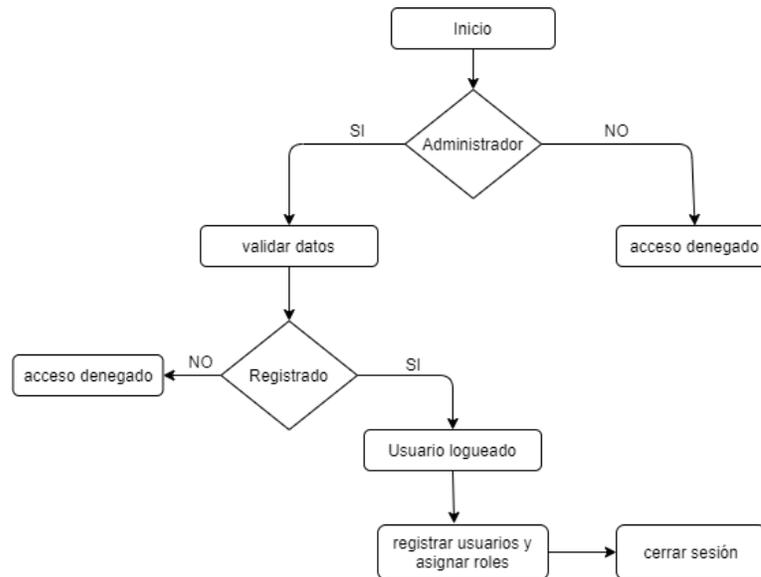


Fuente: Elaboración propia

### 3.9.3. Diagrama de actividades o de flujo

Se diseñó el diagrama de actividades los requisitos de autenticación de usuario y las acciones que tiene, lo cual se visualiza en la siguiente figura:

Figura N.º 3. 8 Diagrama de actividades y módulo de usuario y seguridad

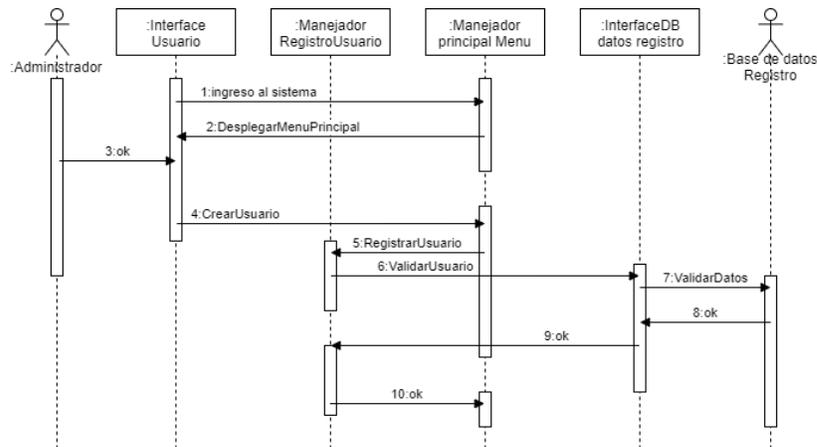


Fuente: Elaboración propia

### 3.9.4. Diagrama de secuencia

Se diseñó el diagrama de secuencias para la autenticación de usuario, acciones de usuario y asignación de roles la cual se puede observar en el siguiente gráfico:

Figura N.º 3. 9 Diagrama de secuencia del módulo de usuario y seguridad, registro

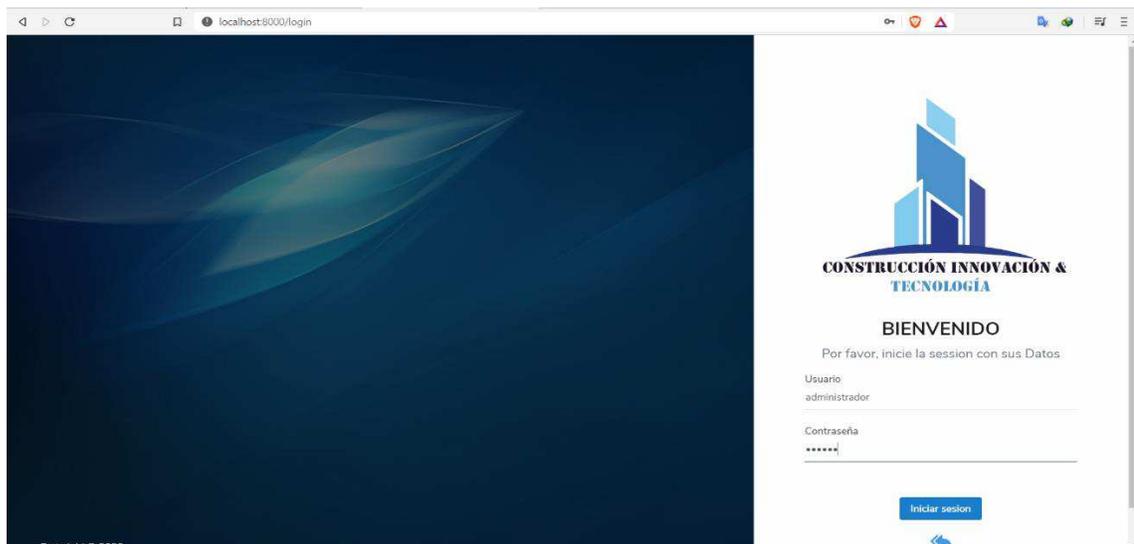


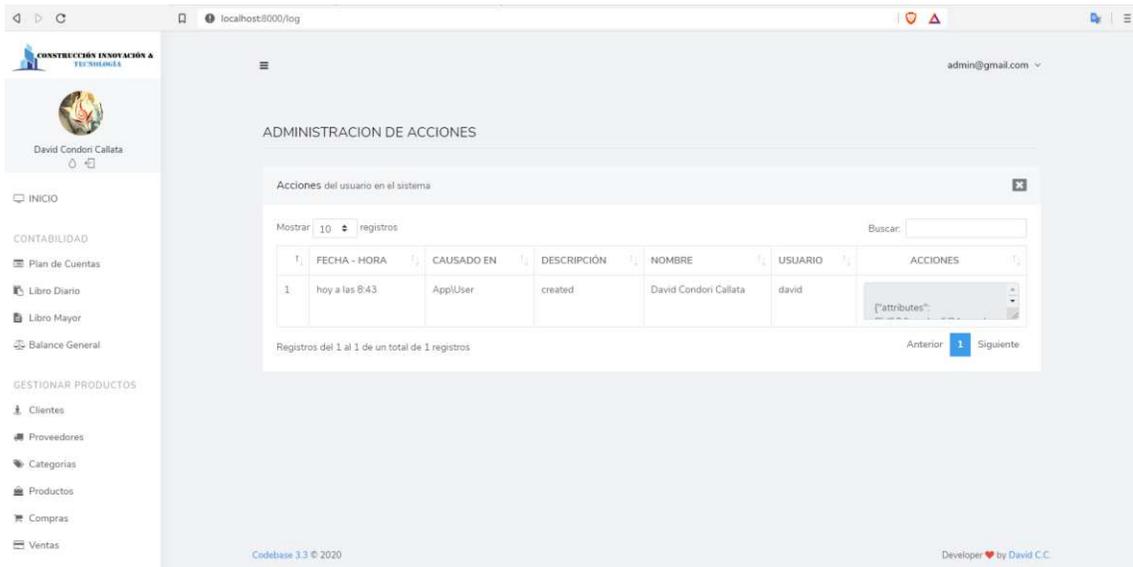
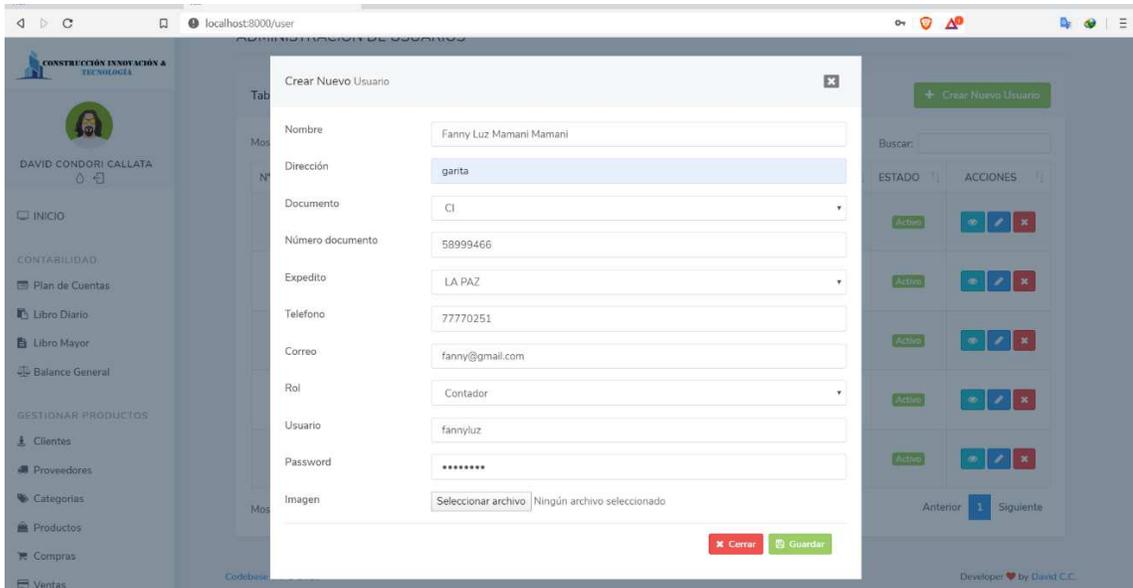
Fuente: Elaboración propia

### 3.10. Fase 4 Programación – Módulo usuario

Después de la programación se obtiene los siguientes resultados como se observa en las siguientes Figuras:

Figura N.º 3. 10 Resultados Modulo usuario y seguridad, acciones de registros





Fuente: Elaboración propia

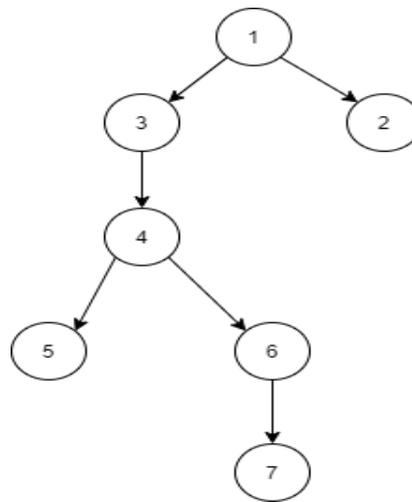
### 3.11. Fase 5 Implementación y pruebas – Modulo usuario

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 10 pagina 67, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.11.1. Prueba de Caja Blanca

Esta prueba se centra en el comportamiento interno y la estructura del programa, en el cual se examina la lógica interna del programa sin considerar los aspectos de rendimiento. La prueba de la caja blanca intenta garantizar que se ejecute al menos una vez todos los caminos independientes en cada módulo, se aplican las decisión verdadera o falsa ejecutando los bucles a sus límites. La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 8 pagina 66.

Figura N.º 3. 11 Grafo de flujo módulo de usuario y seguridad



Fuente: Elaboración propia

Tabla N.º 3. 12 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
7	6	3	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 6 - 7 + 2 = 1$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 3$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 13 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,3,4,6,7	-> Administrador	-> Si usuario es administrador está registrado en la base de datos. -> Si usuario está registrado, usuario logueado puede registra nuevos usuarios y asigna roles	Registro de usuarios y asignación de roles con éxito
1,3,4,5	->Administrador	->Si es administrador valida datos. ->No registrado en la base de datos acceso denegado	Acceso denegado
1,2	->Administrador	->No administrador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.11.2. Prueba de Caja Negra

Esta prueba de software verifica la funcionabilidad sin tomar en cuenta la estructura interna del código, la cual se enfoca solamente en las entradas y salidas, únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 14 Prueba de Caja negra del módulo usuario, seguridad y acciones

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿El formulario de login para iniciar sesión tiene validadores?	Cumple
2	¿Después de que el usuario inicia sesión nos direcciona al menú principal?	Cumple
3	¿Cuándo ingresa al módulo usuario registra nuevos usuarios y asignando roles?	Cumple
4	¿Al registrar un nuevo usuario es validado en la base de datos?	Cumple
5	¿Las acciones del usuario son registrados, cumple?	Cumple

Fuente: Elaboración propia

### 3.11.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de Usuario y de seguridad en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.12. Fase 3 Análisis y diseño - Modulo plan de cuentas

Es considerado como de prioridad alto la cual es Crear el Registro de Plan de Cuentas con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 15 Módulo de plan de cuentas

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de plan de cuentas.	Diseño del diagrama UML	29/02/2020	29/02/2020	Completado
	Codificación	01/03/2020	02/03/2020	Completado
	Pruebas	03/03/2020	03/03/2020	Completado

Fuente: Elaboración propia

#### 3.12.1. Caso de Uso

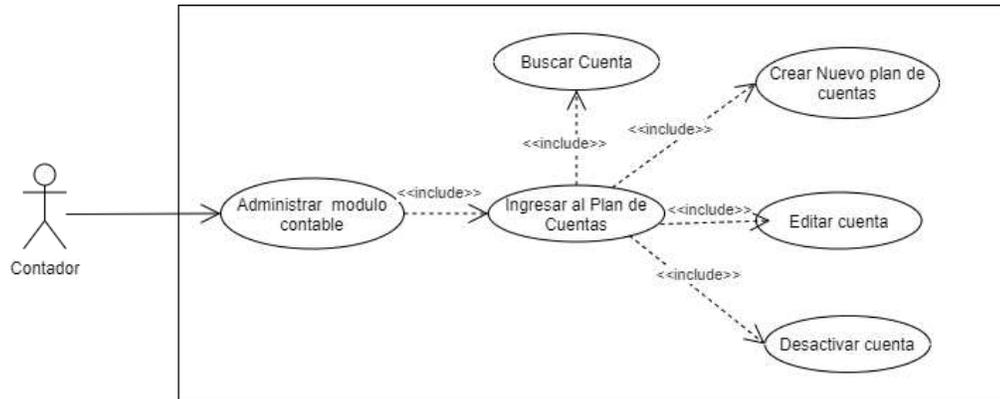
En la siguiente tabla se describe como realiza el registro del plan de cuentas.

Tabla N.º 3. 16 Descripción de caso de uso del Módulo plan de cuentas

Nro.	Descripción
1	Registro y actualización de plan de cuentas
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 12 Caso de uso del Módulo plan de cuentas



Fuente: Elaboración propia

Tabla N.º 3. 17 Inicio de sesión, ingresar al módulo de plan de cuentas

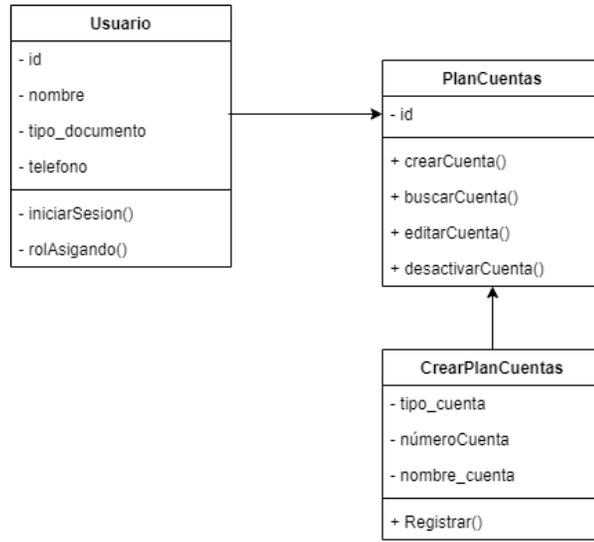
Nombre.	Registrar y actualizar plan de cuentas
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar un nuevo plan de cuentas deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de plan de cuentas para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.12.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar plan de cuentas

Figura N.º 3. 13 Diagrama de clases del Módulo de plan de cuentas

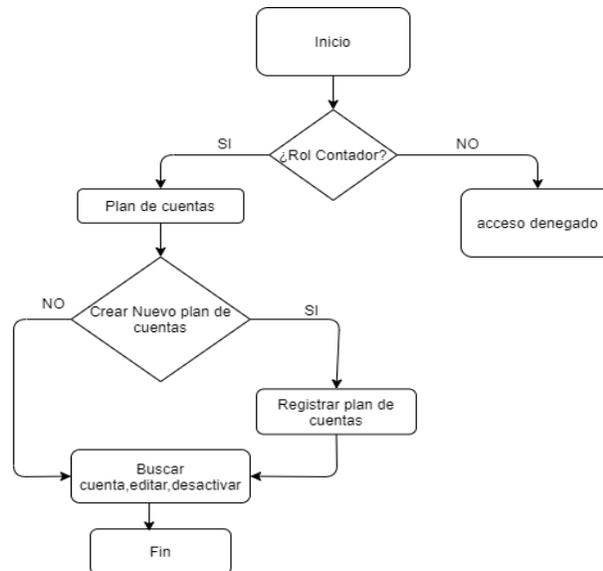


Fuente: Elaboración propia

### 3.12.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar modulo plan de cuentas y autenticación del usuario, el cual se observa en la siguiente figura:

Figura N.º 3. 14 Diagrama de actividades o flujo modelo plan de cuentas

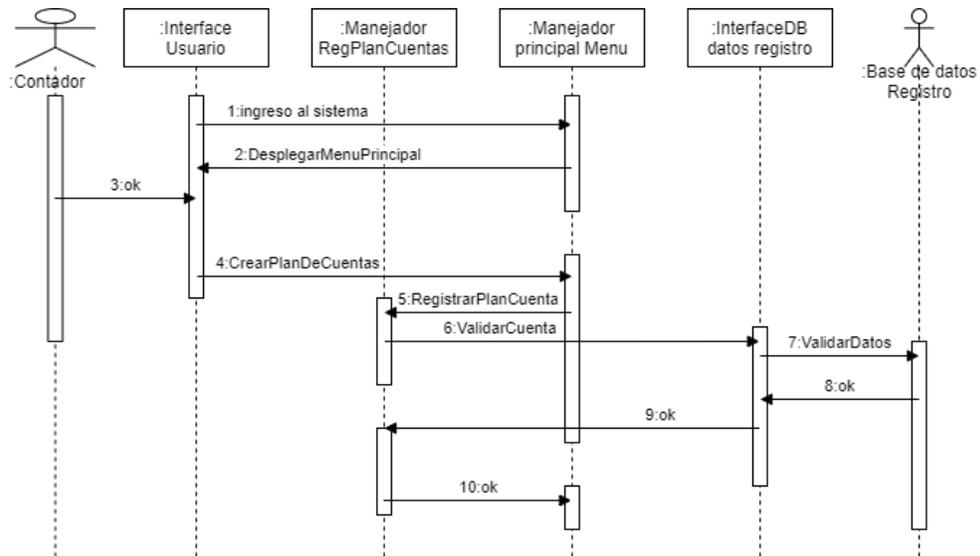


Fuente: Elaboración propia

### 3.12.4. Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de registro de plan de cuentas / actualizar como se observa en la siguiente figura:

Figura N.º 3. 15 Diagrama de secuencia Modulo de registro plan de cuentas

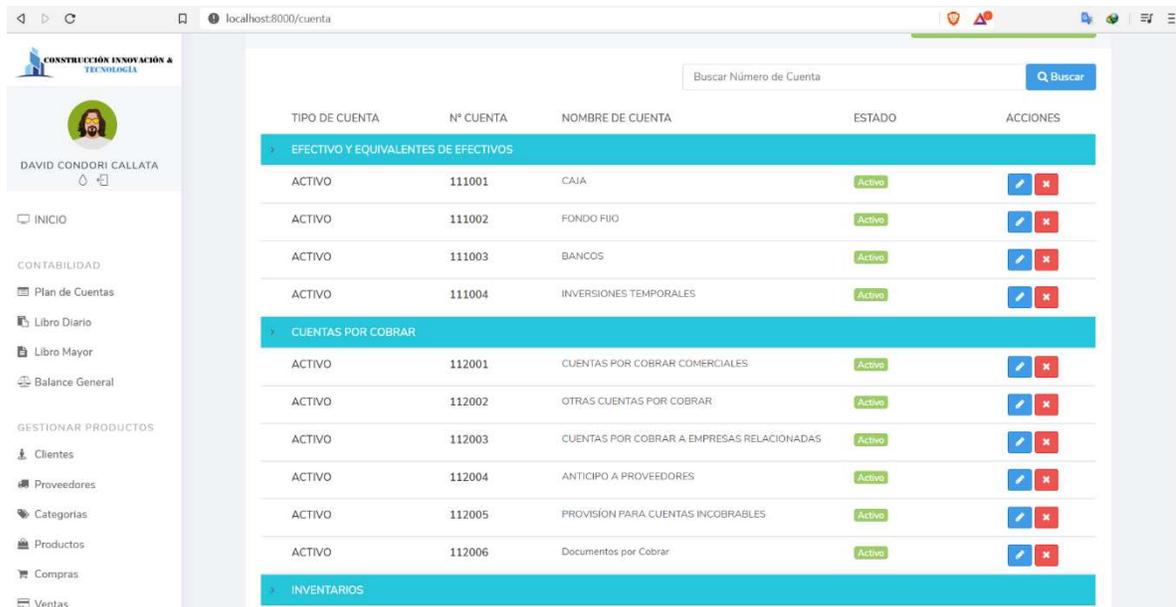
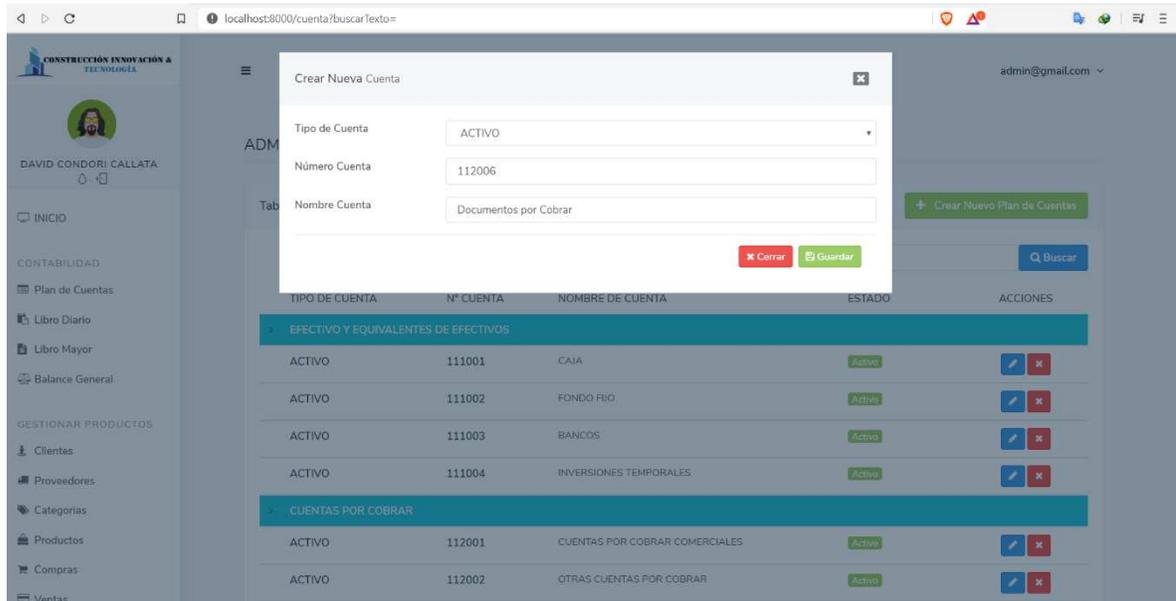


Fuente: Elaboración propia

### 3.13. Fase 4 Programación – Modulo plan de cuentas

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 16 Entornos gráficos del Módulo de plan de cuentas/Registros



Fuente: Elaboración propia

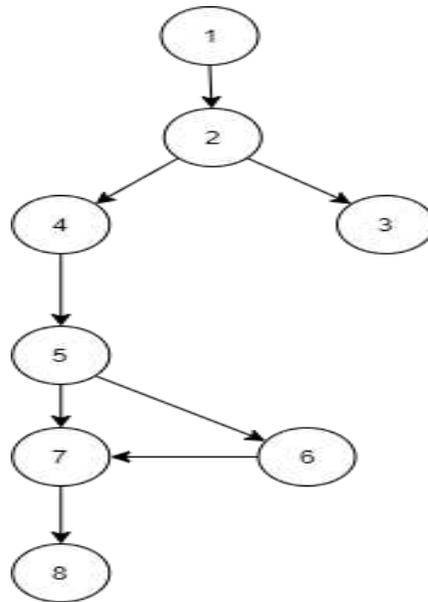
### 3.14. Fase 5 Implementación y pruebas – Modulo plan de cuentas

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 16 pagina 74, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.14.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 14 pagina 73.

Figura N.º 3. 17 Grafo de flujo módulo Plan de cuentas



Fuente: Elaboración propia

Tabla N.º 3. 18 Calculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 19 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevo plan de cuentas.	Registra nuevo plan de cuentas con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No crea nuevo plan de cuentas	Buscar cuentas, editar, desactivar
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.14.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 20 Resultados de la implementación del módulo plan de cuentas

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar los planes de cuentas?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo plan de cuentas?	Cumple
3	¿Cuándo ingresa al módulo plan de cuentas puede registra nuevas cuentas?	Cumple
4	¿Al registrar una nueva cuenta es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo plan de cuentas, cumple?	Cumple

Fuente: Elaboración propia

### 3.14.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de Plan de cuentas en el momento de la implementación pasando primeramente por la prueba

de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.15. Fase 3 Análisis y diseño - Módulo de libro diario

Es considerado como de prioridad alto la cual es Crear el Registro de Libro Diario con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 21 Módulo Libro Diario

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
		04/03/2020	12/03/2020	9 días
	Diseño del diagrama UML	04/03/2020	07/03/2020	Completado
Crear registro de libro diario	Codificación	08/03/2020	11/03/2020	Completado
	Pruebas	12/03/2020	12/03/2020	Completado

Fuente: Elaboración propia

#### 3.15.1. Caso de Uso

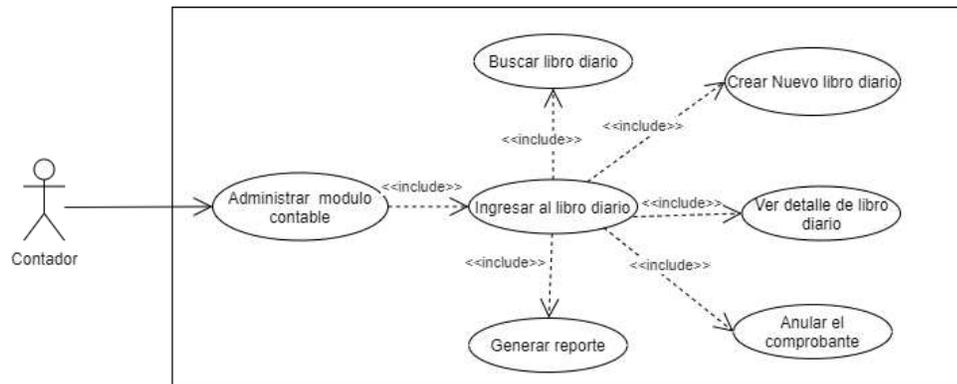
En la siguiente tabla se describe como realiza el registro del libro diario.

Tabla N.º 3. 22 Descripción de caso de uso del Módulo libro diario

Nro.	Descripción
1	Registro y actualización de libro diario
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 18 Caso de uso del Módulo libro diario



Fuente: Elaboración propia

Tabla N.º 3. 23 Inicio de sesión, ingresar al módulo de libro diario

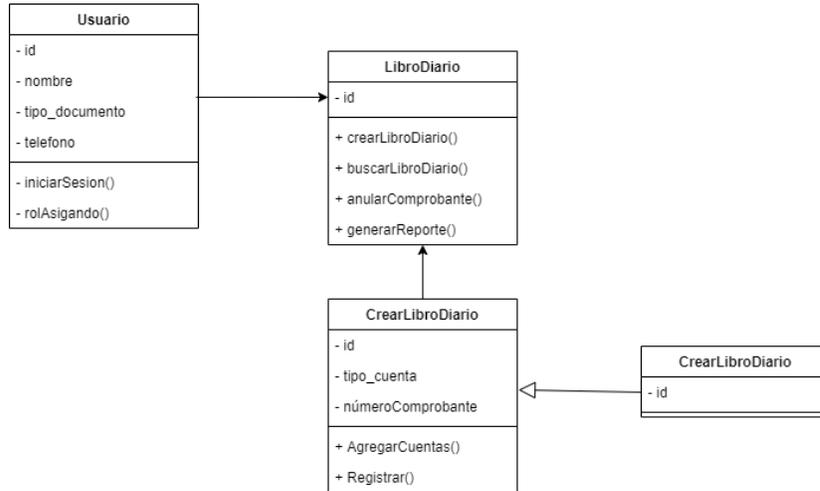
Nombre.	Registrar Libros diario y acciones a realizar
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar un nuevo libro diario deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de libro diario para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.15.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar Libro Diario

Figura N.º 3. 19 Diagrama de clases del Módulo de libro diario

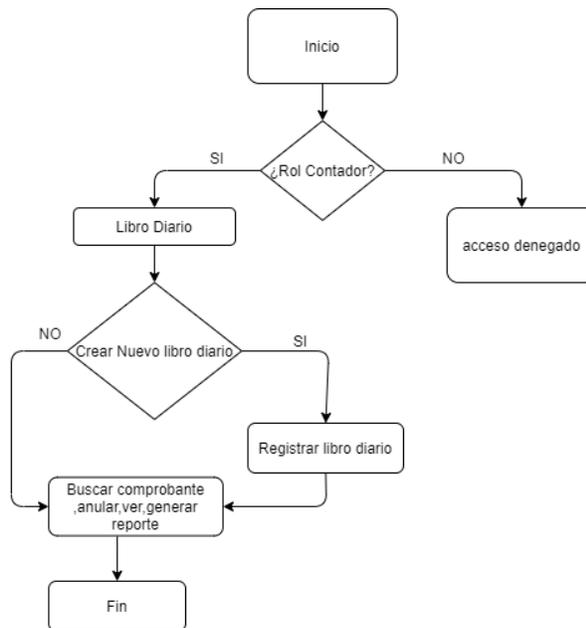


Fuente: Elaboración propia

### 3.15.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar modulo libro diario y autenticación del usuario, el cual se observa en la siguiente figura:

Figura N.º 3. 20 Diagrama de actividades o flujo modelo plan de cuentas

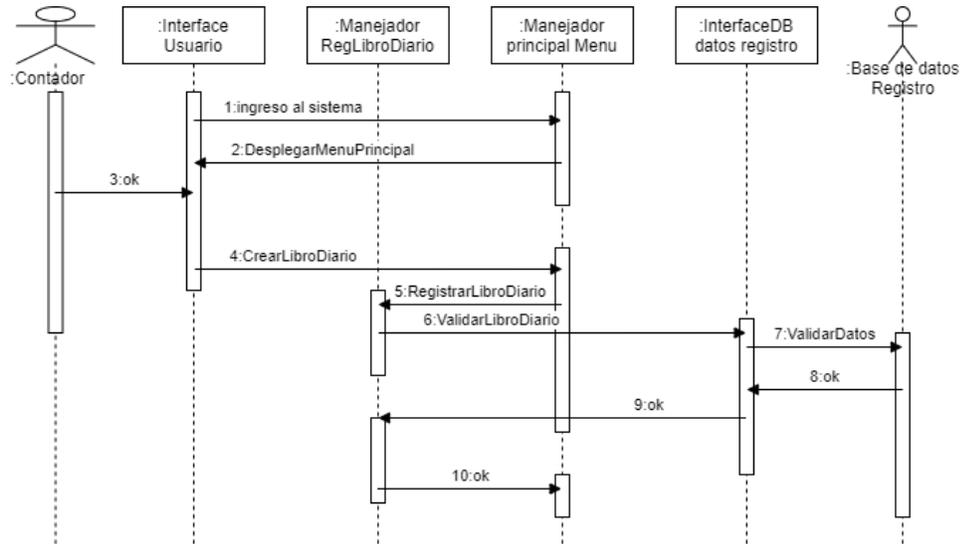


Fuente: Elaboración propia

### 3.15.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de registro de libro diario como se observa en la siguiente figura:

Figura N.º 3. 21 Diagrama de secuencia Modulo de registro libro diario

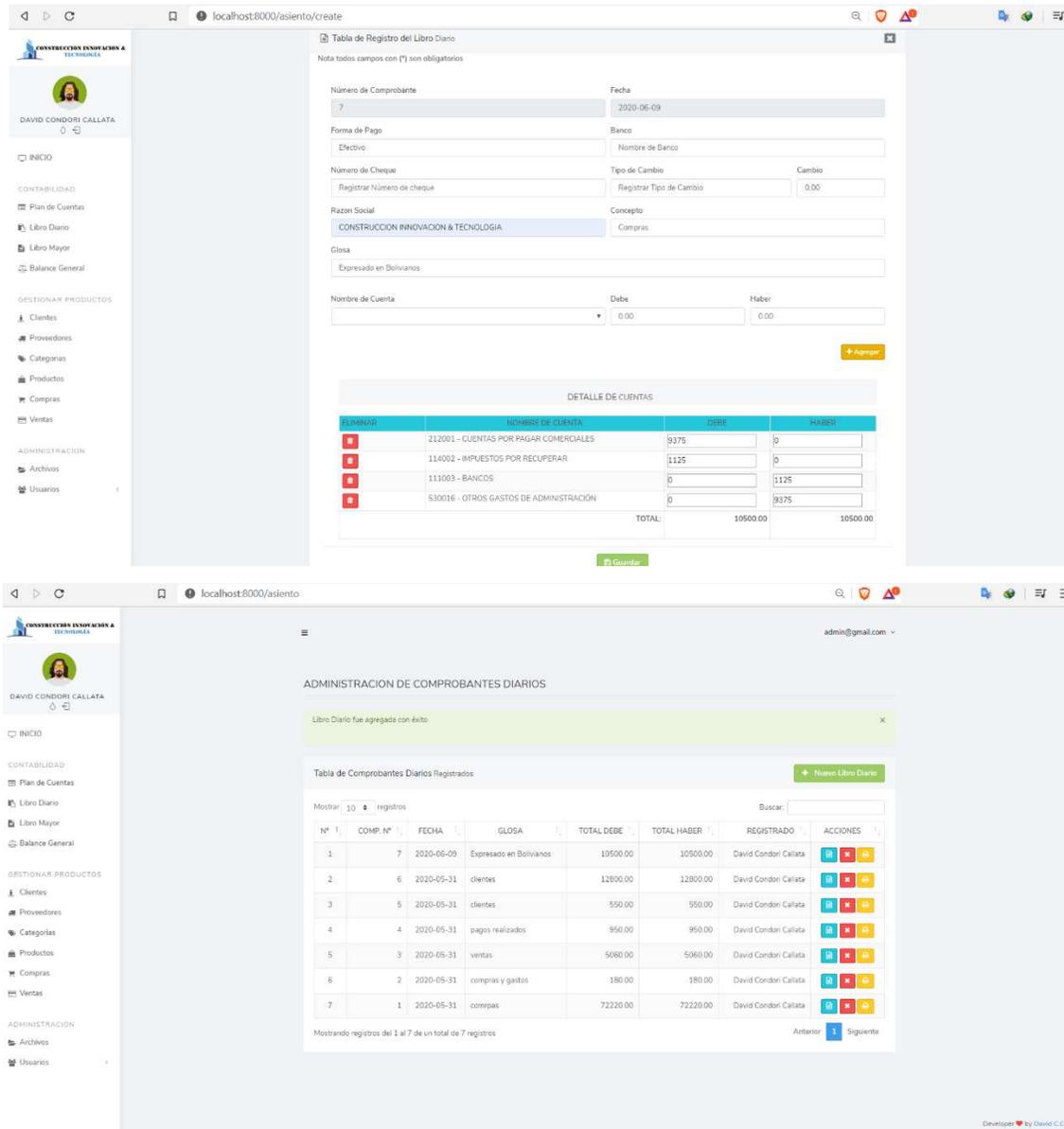


Fuente: Elaboración propia

### 3.16. Fase 4 Programación – Módulo de libro diario

Se obtiene el siguiente resultado como se observa en la siguiente figuras.

Figura N.º 3. 22 Entornos gráficos del Módulo de libro diario



Fuente: Elaboración propia

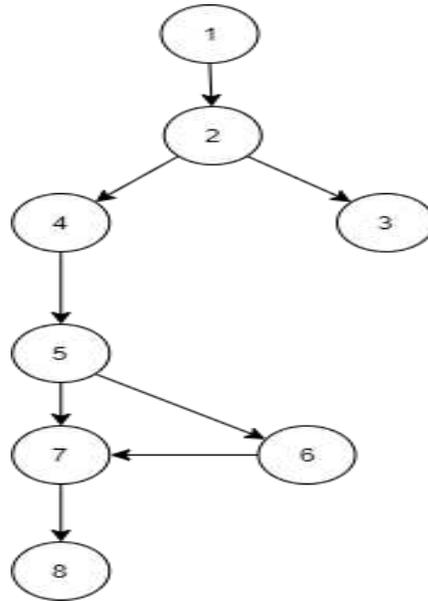
### 3.17. Fase 5 Implementación y pruebas – Módulo de libro diario

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 22 pagina 82, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.17.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 20 pagina 80.

Figura N.º 3. 23 Grafo de flujo módulo Libro diario



Fuente: Elaboración propia

Tabla N.º 3. 24 Calculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 25 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevo libro diario	Registra nuevo libro diario con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No crea nuevo libro diario	Buscar comprobantes, generar reporte, anular, ver.
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.17.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 26 Resultados de la implementación del módulo libro diario

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar un libro diario?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo de libro diario?	Cumple
3	¿Cuándo ingresa al módulo libro diario puede registra un nuevo libro diario?	Cumple
4	¿Al registrar puede generar un reporte del libro diario?	Cumple
5	¿Tiene el CRUD necesario el módulo libro diario, cumple?	Cumple

Fuente: Elaboración propia

### 3.17.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de libro diario en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.18. Fase 3 Análisis y diseño – Modulo Libro Mayor

Es considerado como de prioridad alto la cual es Libro Mayor con todas las acciones necesarias, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 27 Módulo Libro Mayor

REQUISITO	TAREA	Inicio	Fin	Duración	
		13/03/2020	17/03/2020	5 días	
		Desde	Hasta	Estado	
Generar Mayor	Libro	Diseño del diagrama UML	13/03/2020	14/03/2020	Completado
		Codificación	15/03/2020	16/03/2020	Completado
		Pruebas	17/03/2020	17/03/2020	Completado

Fuente: Elaboración propia

#### 3.18.1. Caso de Uso

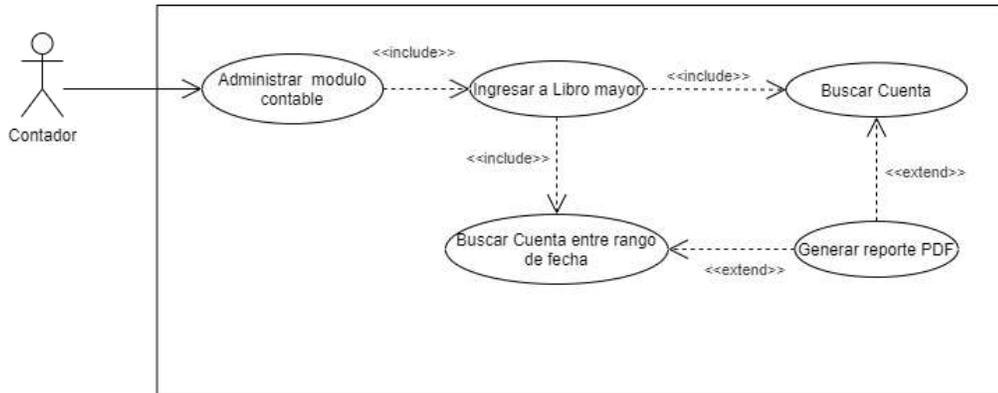
En la siguiente tabla se describe como realiza para generar el libro mayor.

Tabla N.º 3. 28 Descripción de caso de uso del Módulo Libro Mayor

Nro.	Descripción
1	Buscar libro mayor por cuentas y generar los cálculos.
2	Buscar entre rango de fechas y generar los cálculos.

Fuente: Elaboración propia

Figura N.º 3. 24 Caso de uso del Módulo Libro mayor



Fuente: Elaboración propia

Tabla N.º 3. 29 Inicio de sesión, ingresar al módulo de libro mayor

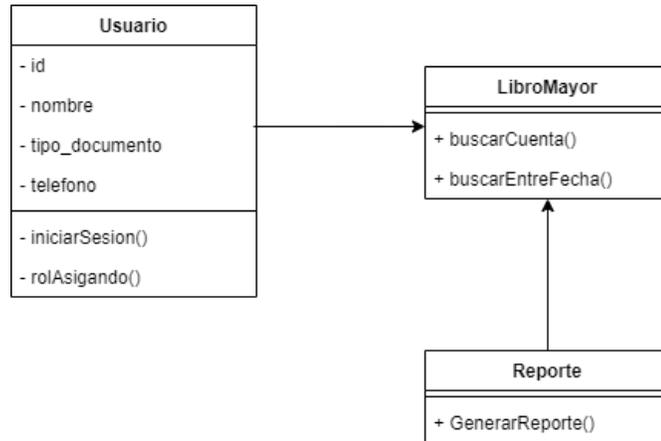
Nombre.	Generar libro mayor
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda generar un libro mayor deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de libro diario para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.18.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento de libro mayor.

Figura N.º 3. 25 Diagrama de clases del Módulo de Libro mayor

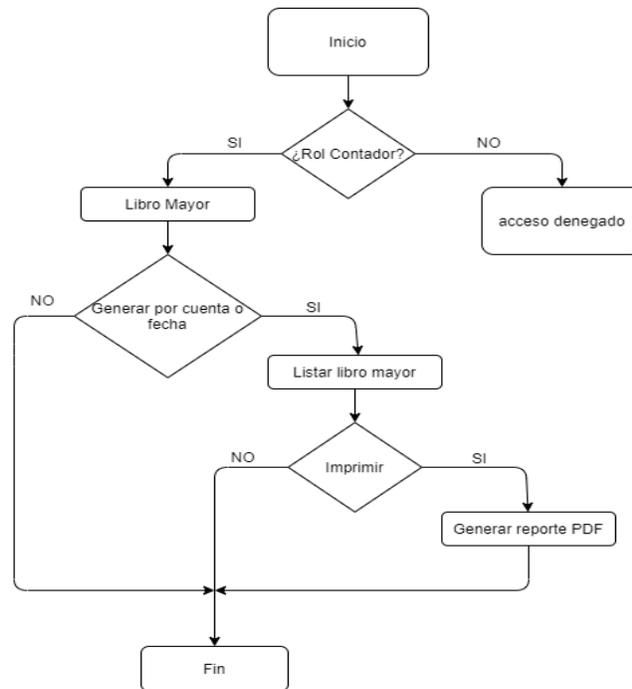


Fuente: Elaboración propia

### 3.18.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para generar libro mayor, el cual se observa en la siguiente figura:

Figura N.º 3. 26 Diagrama de actividades o flujo modelo libro mayor

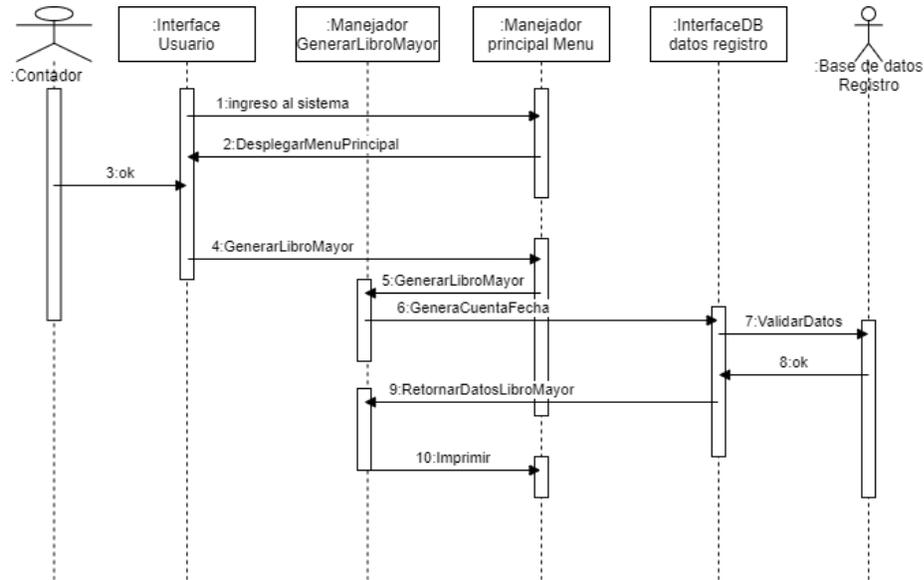


Fuente: Elaboración propia

### 3.18.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de libro mayor como se observa en la siguiente figura:

Figura N.º 3. 27 Diagrama de secuencia Modulo de libro mayor



Fuente: Elaboración propia

### 3.19. Fase 4 Programación – Modulo Libro Mayor

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 28 Entornos gráficos del Módulo de Libro Mayor

The screenshot displays the 'LIBRO MAYOR' (General Ledger) interface. On the left is a sidebar with the user profile 'DAVID CONDORI CALLATA' and a menu with options like 'INICIO', 'CONTABILIDAD', 'Plan de Cuentas', 'Libro Diario', 'Libro Mayor', 'Balance General', 'GESTIONAR PRODUCTOS', 'Clientes', 'Proveedores', 'Categorías', 'Productos', 'Compras', and 'Ventas'. The main area features a search bar with 'Seleccione Cuenta', 'Fecha Inicio', and 'Fecha Fin' fields, along with 'Imprimir' and 'Buscar' buttons. Below this is a table titled 'Tabla de Libro Mayor' with the following data:

FECHA	N°COMPR.	NOMBRE DE CUENTA	DEBE	HABER
2020-05-31	1	CAJA	1,050.00	0.00
2020-05-31	2	CAJA	0.00	180.00
2020-05-31	3	CAJA	5,060.00	0.00
2020-05-31	5	CAJA	550.00	0.00
2020-05-31	6	CAJA	12,800.00	0.00
TOTAL:			19,460.00	180.00
SALDO:			19,280.00	

Fuente: Elaboración propia

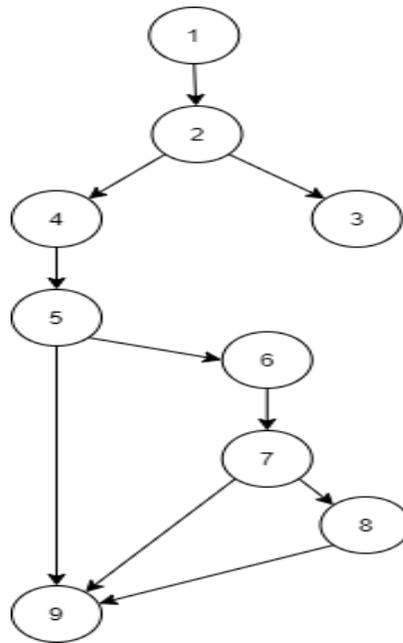
### 3.20. Fase 5 Implementación y pruebas – Modulo Libro Mayor

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 28 pagina 89, posteriormente se realizan las pruebas de caja blanca y negra.

#### 3.20.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 26 pagina 87.

Figura N.º 3. 29 Grafo de flujo módulo libro mayor



Fuente: Elaboración propia

Tabla N.º 3. 30 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	10	3	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 10 + 2 = 0$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 3 + 1 = 4$	
$V(G) = \text{Regiones}$		$V(G) = 3$	
Complejidad ciclomática de 4 que nos indica realizar 4 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 31 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8,9	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede generar libro mayor y reporte.	Generar reporte de libro mayor con éxito
1,2,4,5,6,7,9	-> Rol Contador	->Si es contador	Mostrar Libro mayor en pantalla.
1,2,4,5,9	-> Rol Contador	->No imprime libro mayor	Solo entra al módulo de libro mayor
1,2,3	-> Rol Contador	->No genera ningún libro mayor	Acceso denegado
		->No es contador	

Fuente: Elaboración propia

### 3.20.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 32 Resultados de la implementación del módulo libro mayor

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede generar reportes de libro mayor?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo libro mayor?	Cumple
3	¿Cuándo ingresa al módulo libro mayor puede registra generar reporte?	Cumple
4	¿Al buscar una cuenta para el libro mayor los datos son validado en la base de datos?	Cumple

Fuente: Elaboración propia

### 3.20.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de Libro Mayor en el momento de la implementación pasando primeramente por la prueba de

integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.21. Fase 3 Análisis y diseño - Módulo de Balance General

Es considerado como de prioridad alto la cual es Generar Balance general con todas las acciones necesarias para el reporte, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 33 Módulo Balance general

REQUISITO	TAREA	Inicio	Fin	Duración
		18/03/2020 Desde	22/03/2020 Hasta	5 días Estado
Generar balance general.	Diseño del diagrama UML	18/03/2020	19/03/2020	Completado
	Codificación	20/03/2020	21/03/2020	Completado
	Pruebas	22/03/2020	22/03/2020	Completado

Fuente: Elaboración propia

#### 3.21.1. Caso de Uso

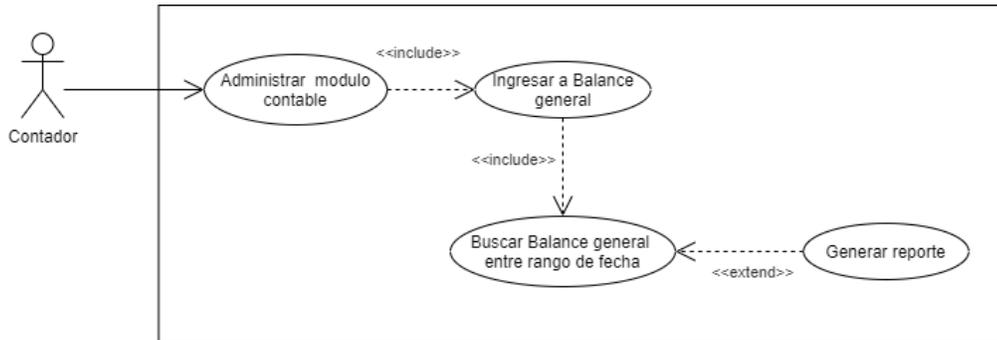
En la siguiente tabla se describe como realiza el balance general.

Tabla N.º 3. 34 Descripción de caso de uso del Módulo balance general

Nro.	Descripción
1	Generar y calcular balance general
2	Generar reporte del balance general

Fuente: Elaboración propia

Tabla N.º 3. 35 Caso de uso del Módulo balance general



Fuente: Elaboración propia

Tabla N.º 3. 36 Inicio de sesión, ingresar al módulo balance general

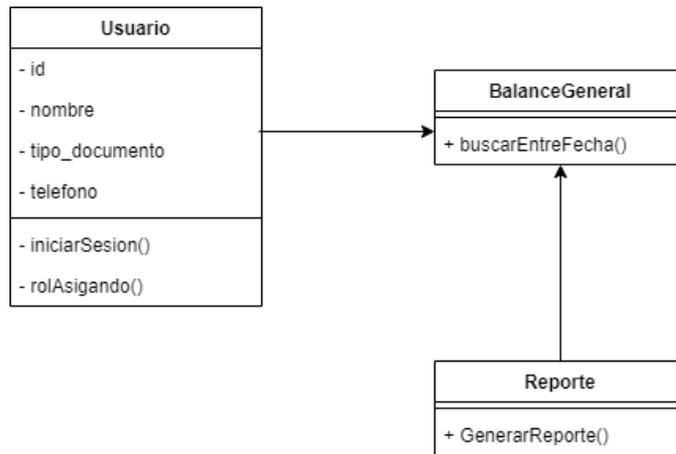
Nombre.	Generar balance general
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda genera un balance general deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de balance general para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.21.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento generar balance general.

Figura N.º 3. 30 Diagrama de clases del Módulo de balance general

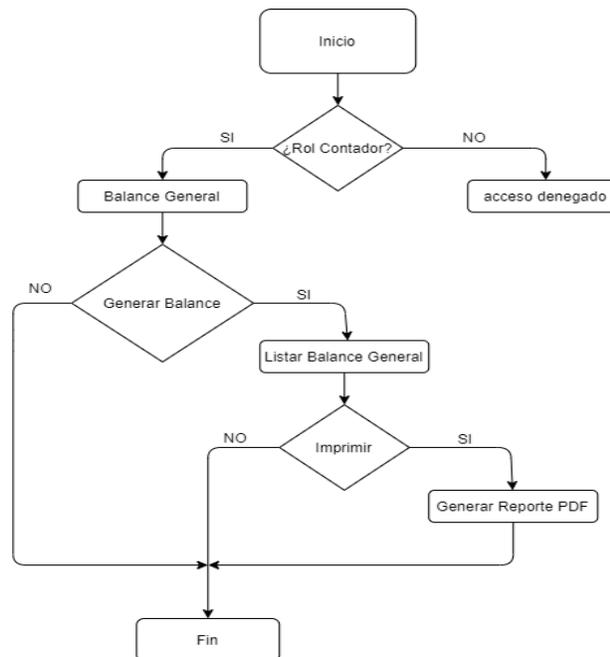


Fuente: Elaboración propia

### 3.21.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para generar balance general, el cual se observa en la siguiente figura:

Figura N.º 3. 31 Diagrama de actividades o flujo modelo balance general

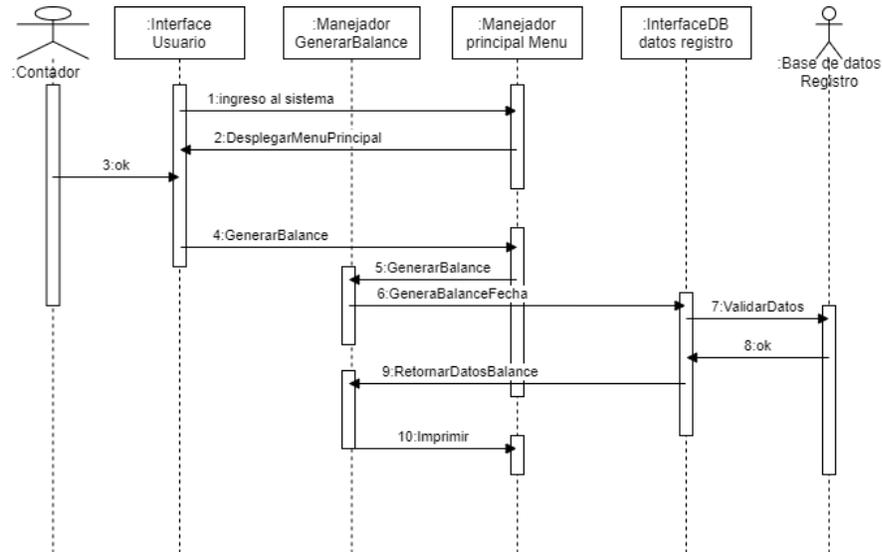


Fuente: Elaboración propia

### 3.21.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos para generar el balance general como se observa en la siguiente figura:

Figura N.º 3. 32 Diagrama de secuencia Modulo de balance general



Fuente: Elaboración propia

### 3.22. Fase 4 Programación – Módulo de Balance General

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 33 Entornos gráficos del Módulo de Balance general

The screenshot shows a web application interface for a general balance sheet. The browser address bar indicates the URL: localhost:8000/general?start=2020-04-29&end=2020-07-12. The user is logged in as admin@gmail.com. The sidebar on the left contains navigation links for 'INICIO', 'CONTABILIDAD' (Plan de Cuentas, Libro Diario, Libro Mayor, Balance General), and 'GESTIONAR PRODUCTOS' (Clientes, Proveedores, Categorías, Productos, Compras, Ventas). The main content area is titled 'BALANCE GENERAL' and features a search bar with 'Fecha Inicio' and 'Fecha Fin' fields, and buttons for 'Imprimir' and 'Buscar'. Below the search bar is a table titled 'Tabla de Balance General'.

NOMBRE DE CUENTA	DEBE	HABER
<b>ACTIVO</b>		
ANTICIPO A PROVEEDORES	848.21	0.00
	<b>848.21</b>	
BANCOS	38,500.00	2,075.00
	<b>36,425.00</b>	
CAJA	19,460.00	180.00
	<b>19,280.00</b>	
<b>PATRIMONIO</b>		
OTROS GASTOS DE ADMINISTRACIÓN	0.00	22,575.00
		<b>22,575.00</b>
PRÉSTAMOS BANCARIOS	0.00	32,000.00
		<b>32,000.00</b>
AJUSTE DE CAPITAL	0.00	24,520.00
		<b>24,520.00</b>
VENTAS	0.00	4,517.86
		<b>4,517.86</b>
	<b>TOTAL ACTIVO:</b>	<b>71,240.72</b>
	<b>TOTAL PASIVO:</b>	<b>42,202.86</b>
	<b>TOTAL PATRIMONIO:</b>	<b>29,037.86</b>
<b>BALANCE GENERAL:</b>		<b>ACTIVO   PASIVO + PATRIMONIO</b>
		<b>71,240.72    71,240.72</b>

Developer ❤️ by David C.C.

Fuente: Elaboración propia

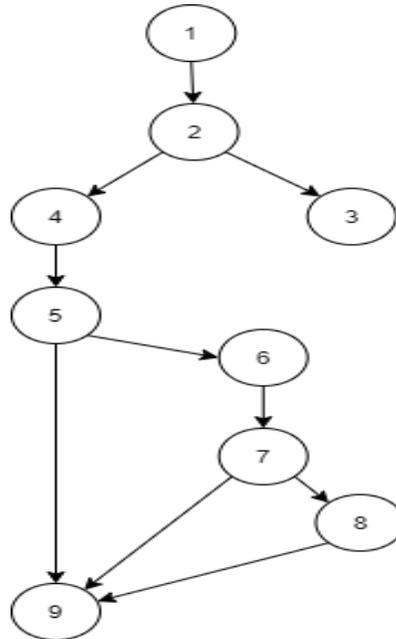
### 3.23. Fase 5 Implementación y pruebas – Módulo de Balance General

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 33 pagina 96, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.23.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 31 pagina 94.

Figura N.º 3. 34 Grafo de flujo módulo balance general



Fuente: Elaboración propia

Tabla N.º 3. 37 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
9	10	3	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$			$V(G) = 9 - 10 + 2 = 1$
$V(G) = \text{Nodos Predicados} + 1$			$V(G) = 3 + 1 = 4$
$V(G) = \text{Regiones}$			$V(G) = 3$
Complejidad ciclomática de 4 que nos indica realizar 4 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 38 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8,9	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede generar balance general y hace el reporte.	Genera balance general y reporte con éxito
1,2,4,5,7,9	-> Rol Contador	->Si es contador ->No genera el reporte	Solo genera la vista de balance general
1,2,4,5,9	-> Rol Contador	->Solo ingresa al módulo sin realizar nada	Solo ingresa al módulo
1,2,3	-> Rol Contador	->No es contador	Acceso denegado

Fuente: Elaboración propia

### 3.23.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 39 Resultados de la implementación del módulo Balance general

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede generar balance general?	Cumple
2	¿Después de que el usuario autorizado inicia sesión puede ingresar al módulo de balance general?	Cumple
3	¿Cuándo ingresa al módulo balance general puede generar el reporte?	Cumple

Fuente: Elaboración propia

### 3.23.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de Balance General en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.24. Fase 3 Análisis y diseño - Módulo de Productos

Es considerado como de prioridad alto la cual es Crear el Registro de productos con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 40 Módulo Productos

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de productos.	Diseño del diagrama UML	23/03/2020	24/03/2020	6 días
	Codificación	25/03/2020	27/03/2020	Completado
	Pruebas	28/03/2020	28/03/2020	Completado

Fuente: Elaboración propia

#### 3.24.1. Caso de Uso

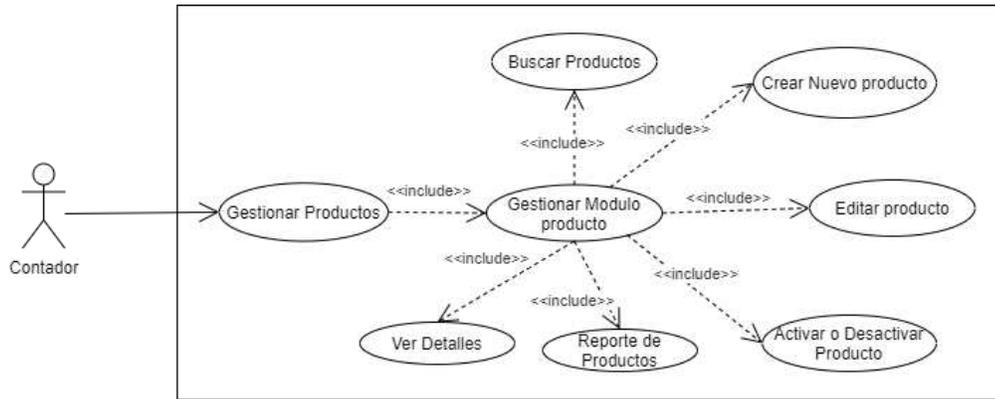
En la siguiente tabla se describe como realiza el registro de productos

Tabla N.º 3. 41 Descripción de caso de uso del Módulo productos

Nro.	Descripción
1	Registro y actualización productos
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 35 Caso de uso del Módulo productos



Fuente: Elaboración propia

Tabla N.º 3. 42 Inicio de sesión, ingresar al módulo de productos

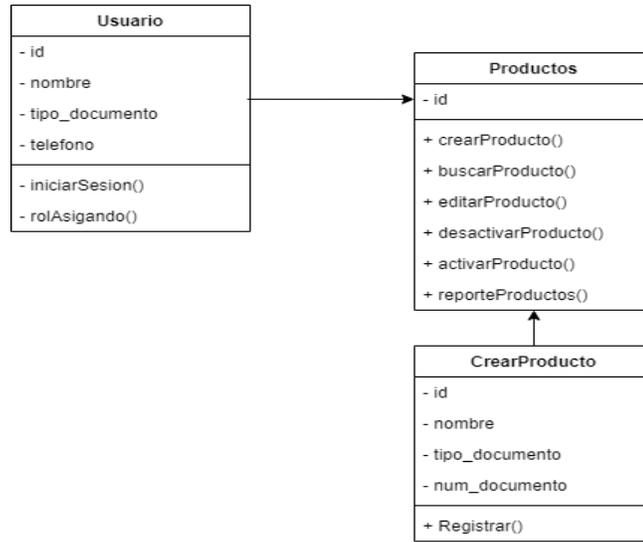
Nombre.	Registrar y actualizar productos
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar un nuevo producto deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de productos para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.24.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar productos.

Figura N.º 3. 36 Diagrama de clases del Módulo de productos

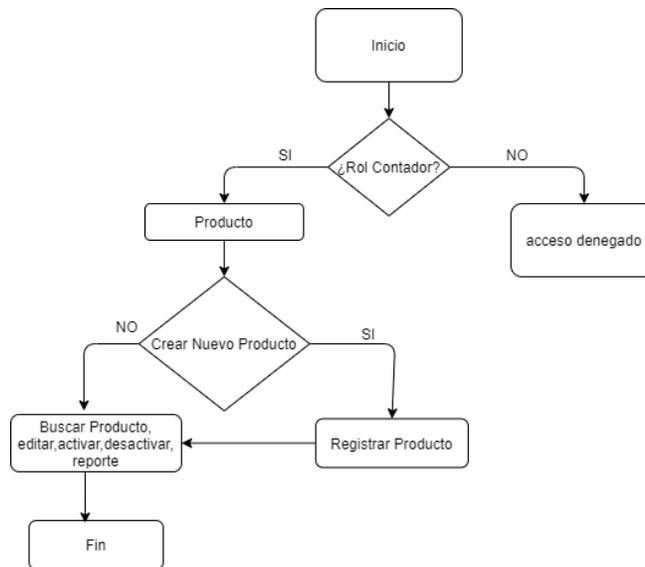


Fuente: Elaboración propia

### 3.24.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar modulo productos, el cual se observa en la siguiente figura:

Figura N.º 3. 37 Diagrama de actividades o flujo modulo productos

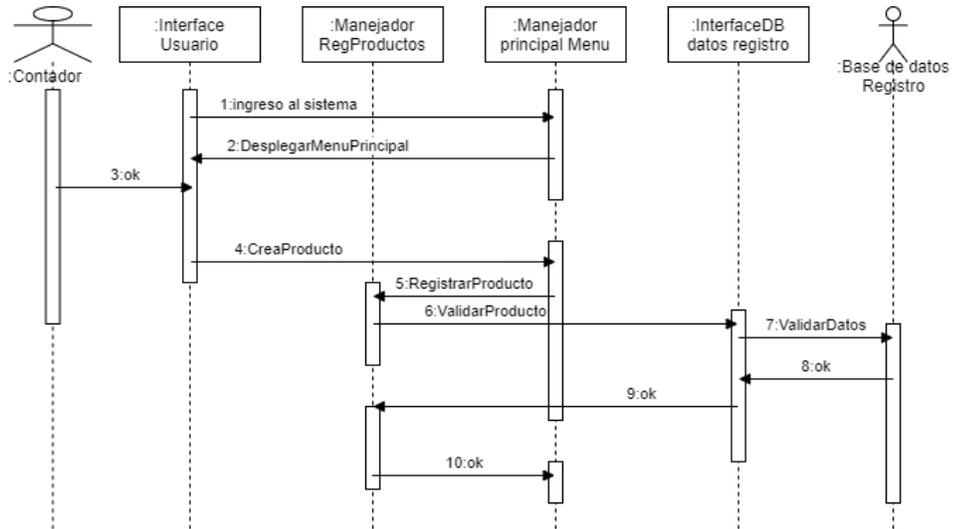


Fuente: Elaboración propia

### 3.24.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de registro de productos / actualizar como se observa en la siguiente figura:

Figura N.º 3. 38 Diagrama de secuencia Modulo de productos

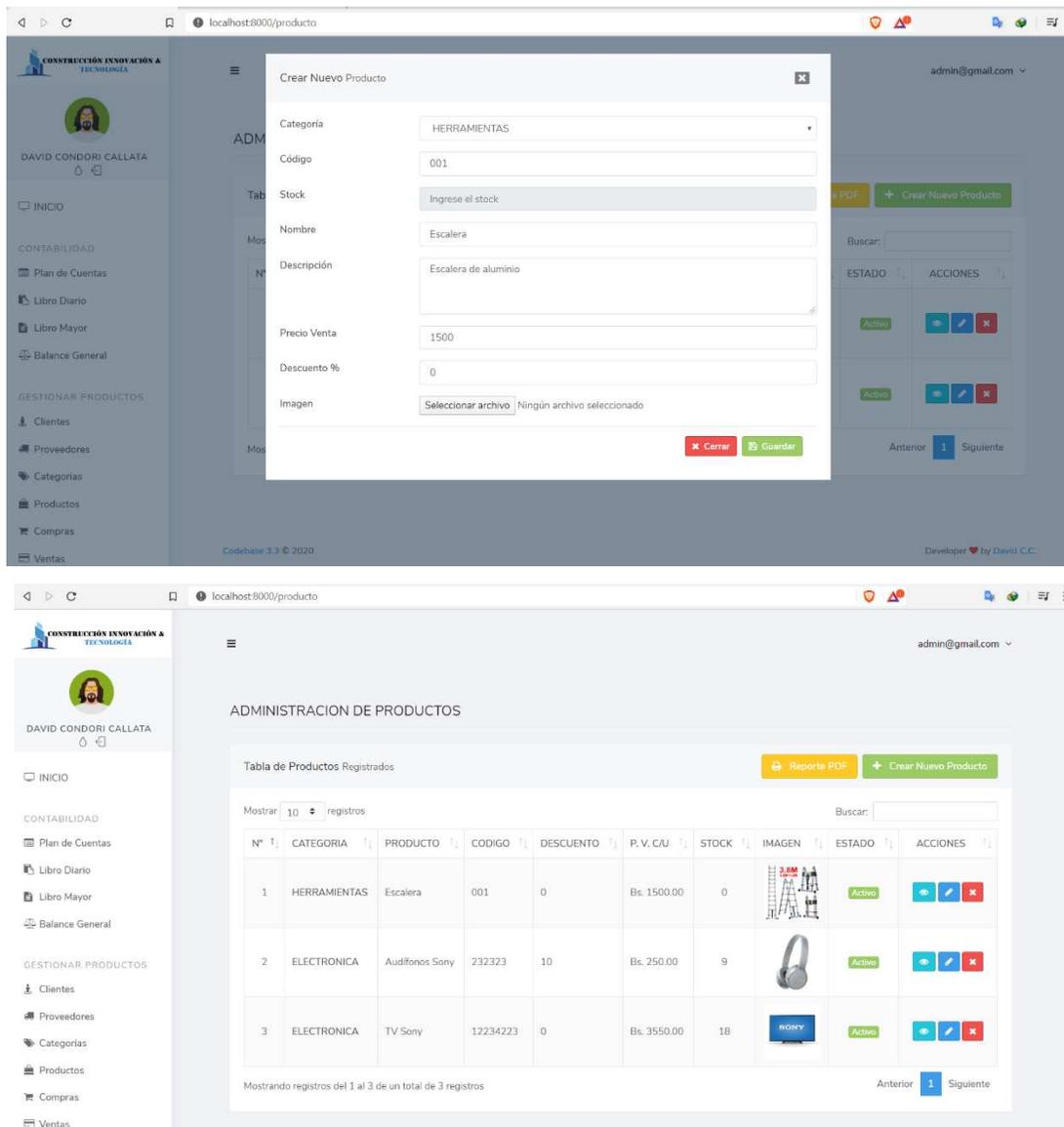


Fuente: Elaboración propia

### 3.25. Fase 4 Programación – Módulo de Productos

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 39 Entornos gráficos del Módulo de productos/Registros



Fuente: Elaboración propia

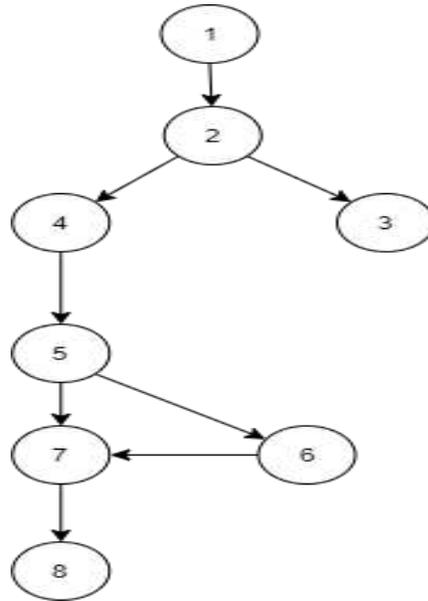
### 3.26. Fase 5 Implementación y pruebas – Módulo de Productos

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 39 pagina 103, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.26.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 37 pagina 101.

Figura N.º 3. 40 Grafo de flujo módulo de productos



Fuente: Elaboración propia

Tabla N.º 3. 43 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 44 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevos productos.	Registra nuevos productos con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No crea nuevos productos	Buscar productos, editar, desactivar, desactivar y reporte.
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.26.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 45 Resultados de la implementación del módulo plan de cuentas

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar los productos?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo de productos?	Cumple
3	¿Cuándo ingresa al módulo productos puede registra?	Cumple
4	¿Al registrar un producto es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo productos, cumple?	Cumple

Fuente: Elaboración propia

### 3.26.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de productos en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.27. Fase 3 Análisis y diseño - Módulo de categorías

Es considerado como de prioridad alto la cual es Crear el Registro de categorías con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 46 Módulo categorías

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de categorías.	Diseño del diagrama UML	29/03/2020	29/03/2020	Completado
	Codificación	30/03/2020	31/03/2020	Completado
	Pruebas	01/03/2020	01/03/2020	Completado

Fuente: Elaboración propia

#### 3.27.1. Caso de Uso

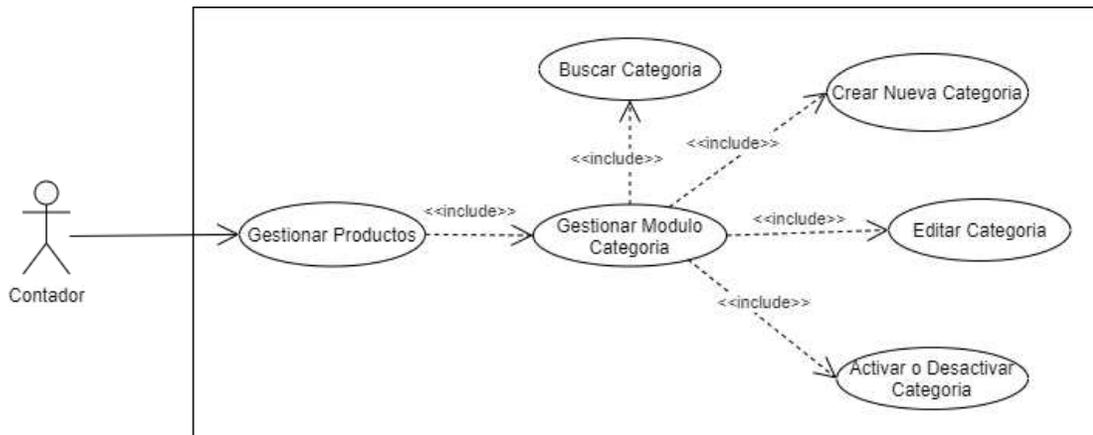
En la siguiente tabla se describe como realiza el registro de las categorías.

Tabla N.º 3. 47 Descripción de caso de uso del Módulo de categorías

Nro.	Descripción
1	Registro y actualización de categorías.
2	Y las acciones necesarias que contendrá.

Fuente: Elaboración propia

Figura N.º 3. 41 Caso de uso del Módulo de categorías



Fuente: Elaboración propia

Tabla N.º 3. 48 Inicio de sesión, ingresar al módulo de categorías

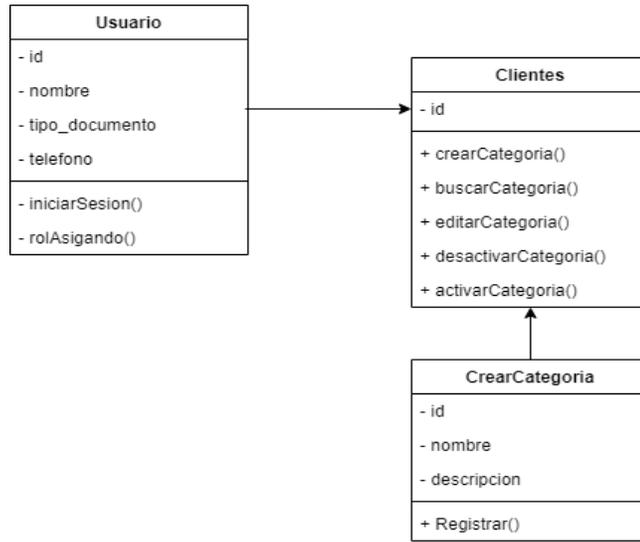
Nombre.	Registrar y actualizar categorías
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar una nueva categoría deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de categorías para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.27.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar categorías.

Figura N.º 3. 42 Diagrama de clases del Módulo de categorías

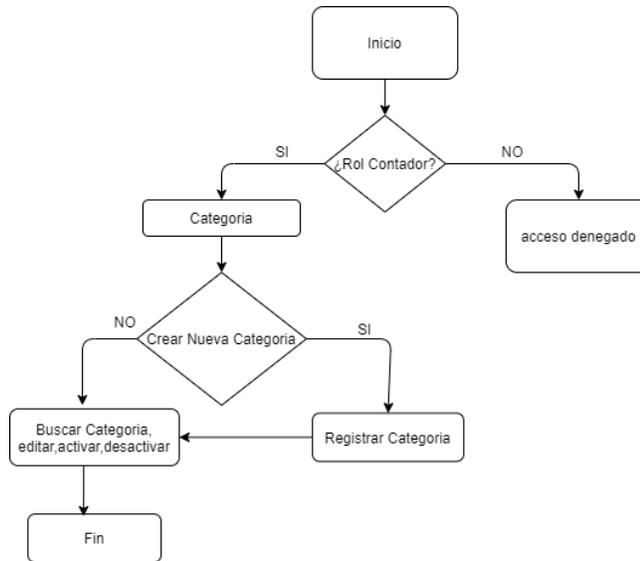


Fuente: Elaboración propia

### 3.27.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar categorías, el cual se observa en la siguiente figura:

Figura N.º 3. 43 Diagrama de actividades o flujo módulo de categorías

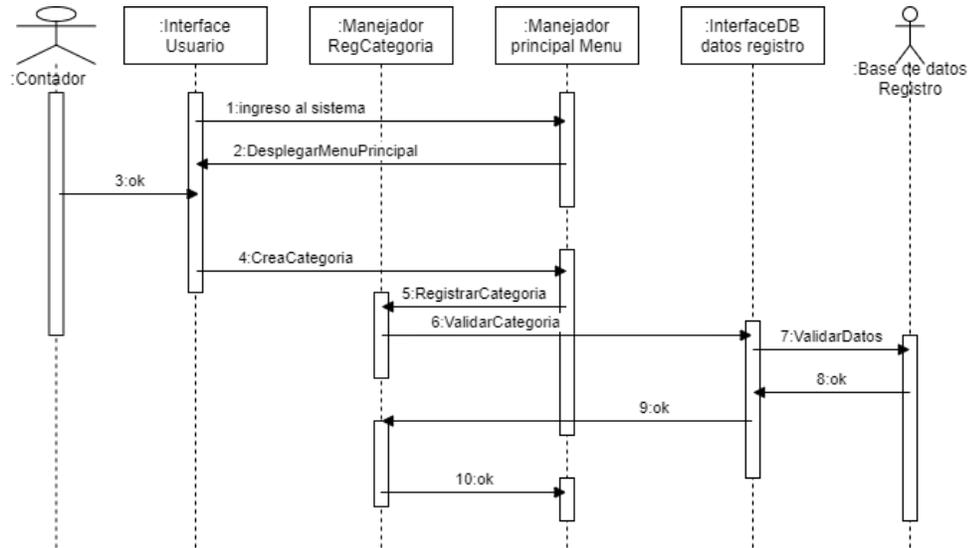


Fuente: Elaboración propia

### 3.27.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de categorías / actualizar como se observa en la siguiente figura:

Figura N.º 3. 44 Diagrama de secuencia Modulo de registro categorías

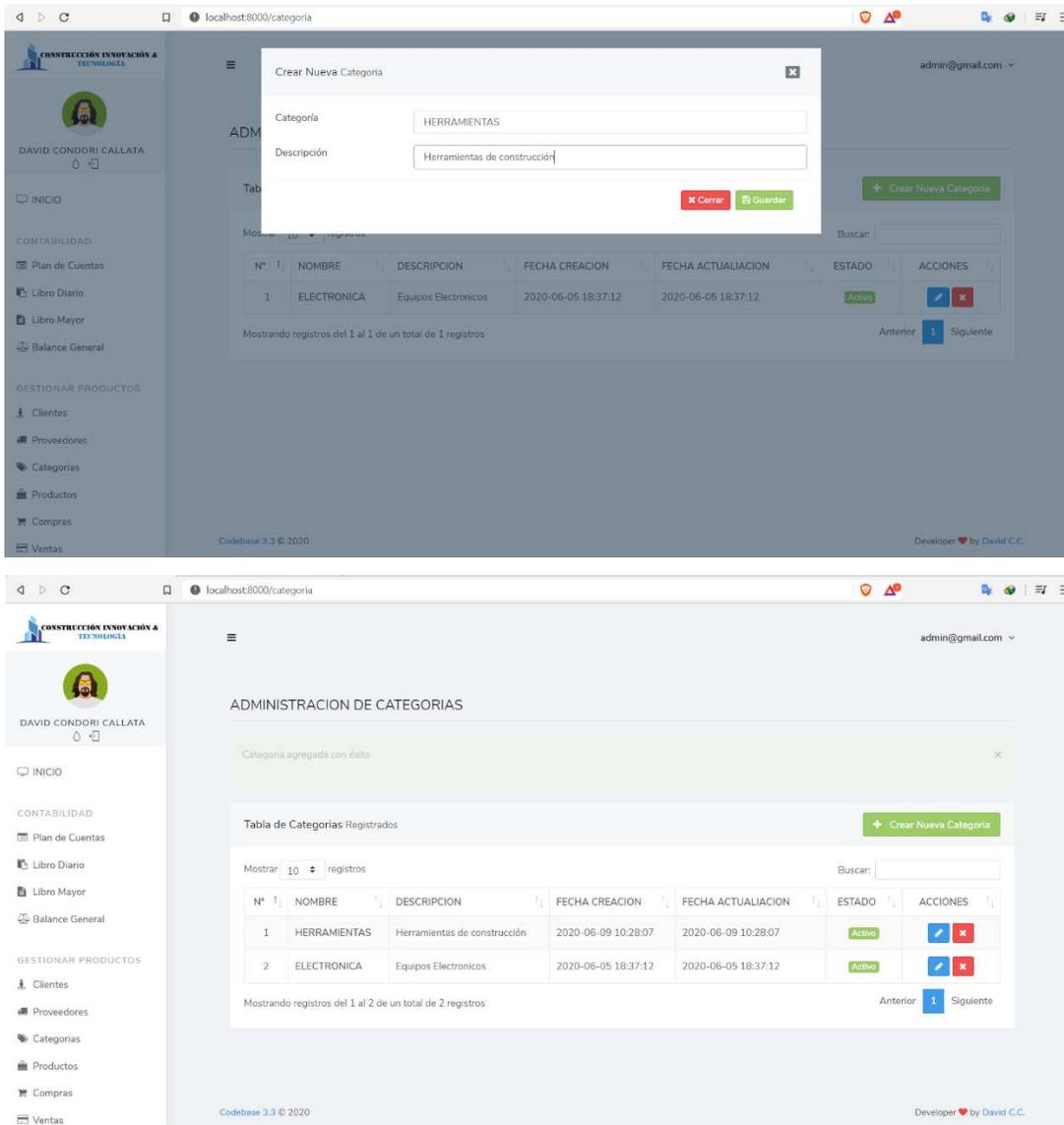


Fuente: Elaboración propia

### 3.28. Fase 4 Programación – Módulo de categorías

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 45 Entornos gráficos del Módulo de Categorías /Registros



Fuente: Elaboración propia

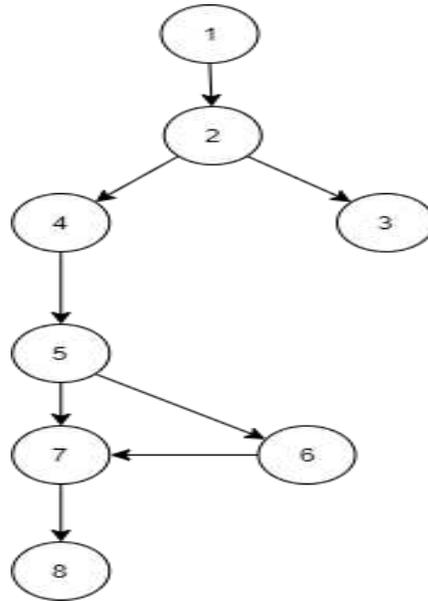
### 3.29. Fase 5 Implementación y pruebas – Módulo de categorías

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 45 pagina 110, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.29.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 43 pagina 108.

Figura N.º 3. 46 Grafo de flujo módulo de categorías



Fuente: Elaboración propia

Tabla N.º 3. 49 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 50 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevas categorías.	Registra nuevas categorías con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No crea nueva categoría	Buscar categoría, editar, desactivar, activar.
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.29.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 51 Resultados de la implementación del módulo de categorías

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar nuevas categorías?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo de categorías?	Cumple
3	¿Cuándo ingresa al módulo de categorías puede registra realizar nuevos registros o modificar?	Cumple
4	¿Al registrar una categoría es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo de categorías, cumple?	Cumple

Fuente: Elaboración propia

### 3.29.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de categorías en el momento de la implementación pasando primeramente por la prueba de

integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.30. Fase 3 Análisis y diseño - Módulo de Proveedor

Es considerado como de prioridad alto la cual es Crear el Registro de proveedores con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 52 Módulo de Proveedor

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de proveedores.	Diseño del diagrama UML	02/04/2020	03/04/2020	Completado
	Codificación	04/04/2020	05/04/2020	Completado
	Pruebas	06/04/2020	06/04/2020	Completado

Fuente: Elaboración propia

#### 3.30.1. Caso de Uso

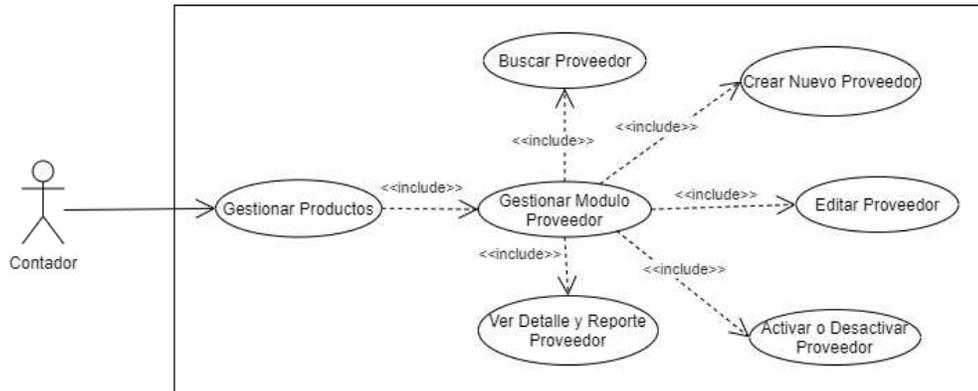
En la siguiente tabla se describe como realiza el registro de proveedores.

Tabla N.º 3. 53 Descripción de caso de uso del Módulo proveedor

Nro.	Descripción
1	Registro y actualización de proveedores
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 47 Caso de uso del Módulo de proveedor



Fuente: Elaboración propia

Tabla N.º 3. 54 Inicio de sesión, ingresar al módulo de proveedor

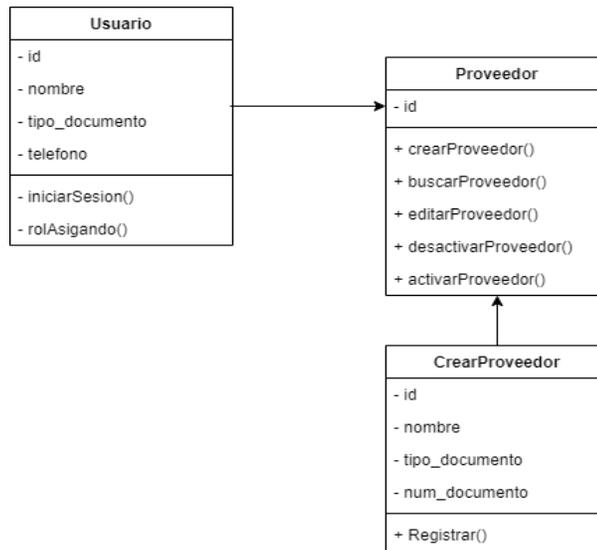
Nombre.	Registrar y actualizar proveedor
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar un nuevo proveedor deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de proveedor para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.30.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar proveedores.

Figura N.º 3. 48 Diagrama de clases del Módulo de proveedor

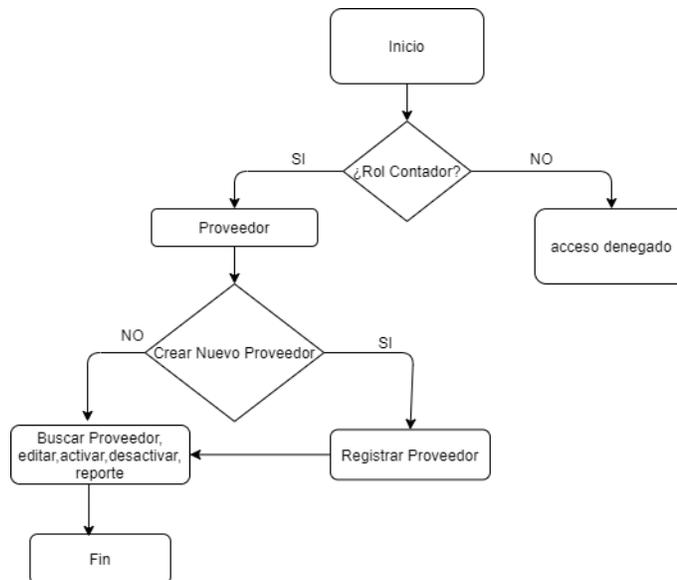


Fuente: Elaboración propia

### 3.30.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar proveedores, el cual se observa en la siguiente figura:

Figura N.º 3. 49 Diagrama de actividades o flujo módulo de proveedores

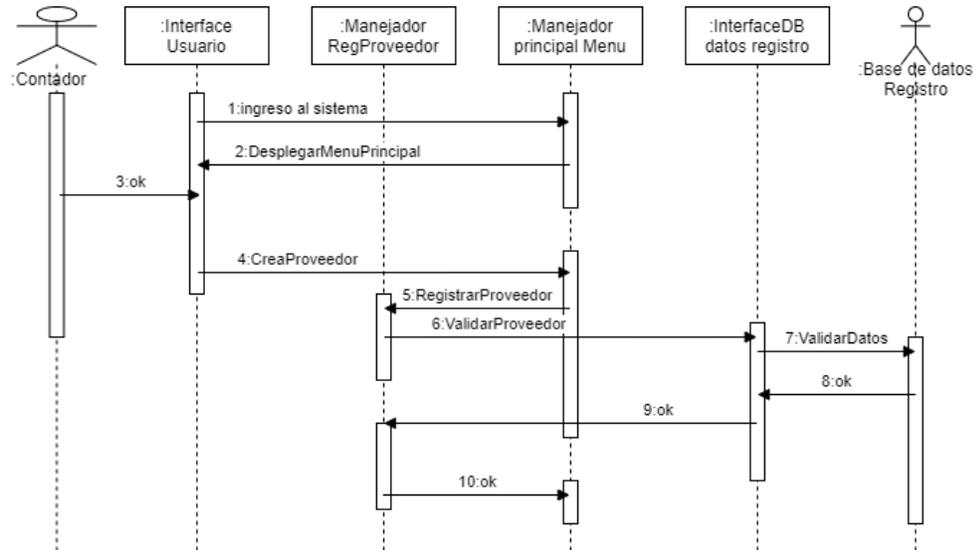


Fuente: Elaboración propia

### 3.30.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de proveedor como se observa en la siguiente figura:

Figura N.º 3. 50 Diagrama de secuencia Modulo de proveedor

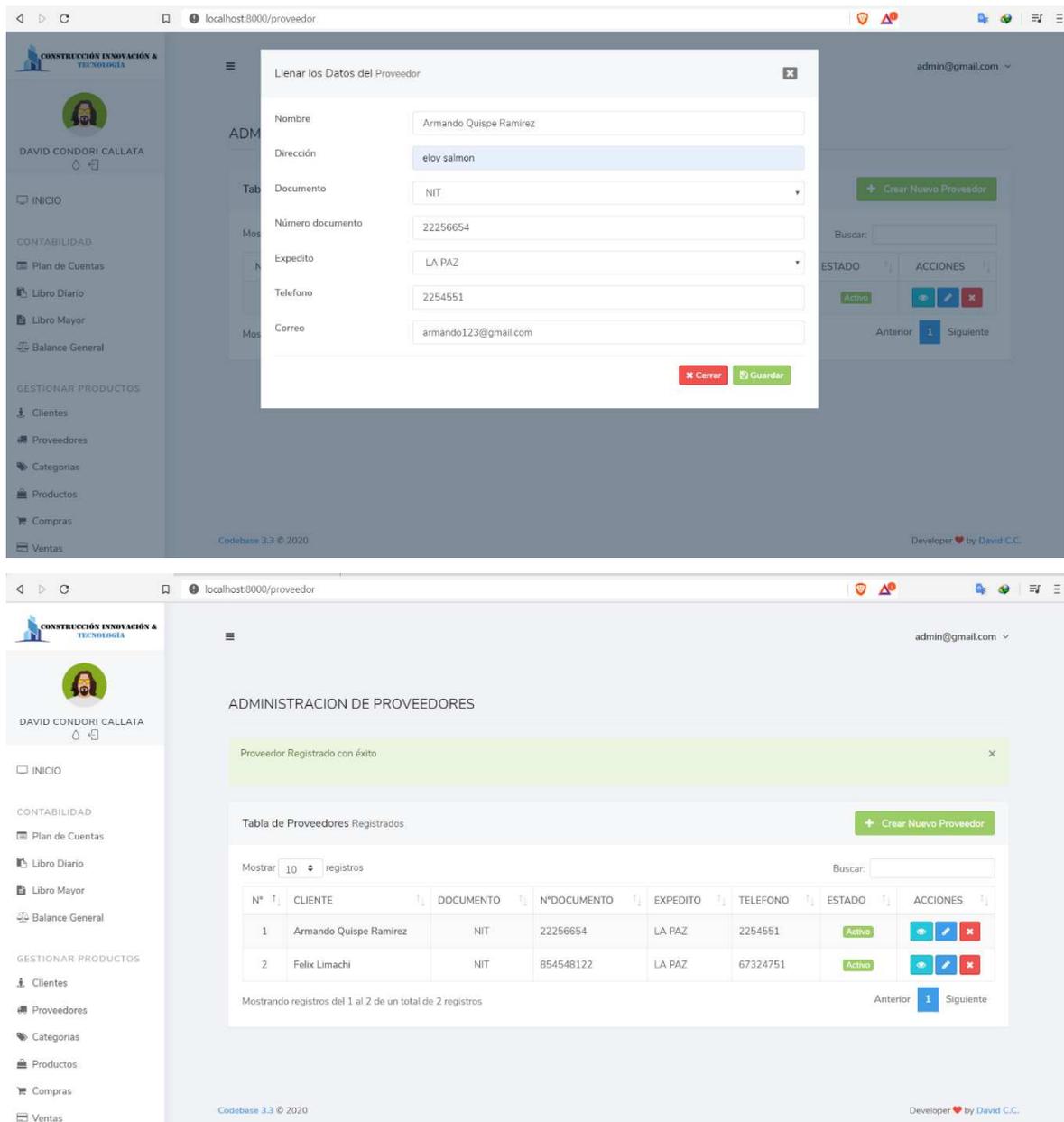


Fuente: Elaboración propia

### 3.31. Fase 4 Programación – Módulo de Proveedor

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 51 Entornos gráficos del Módulo de proveedor/Registros



Fuente: Elaboración propia

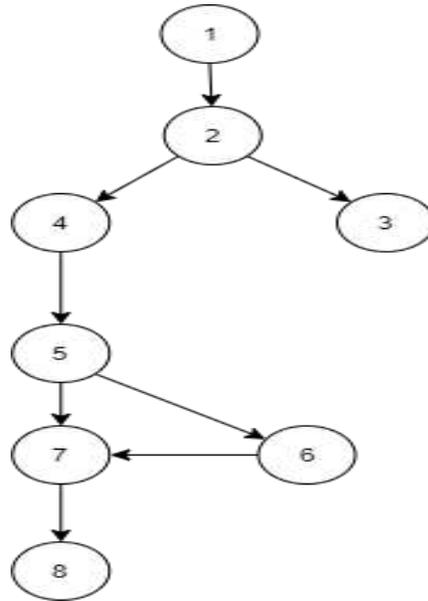
### 3.32. Fase 5 Implementación y pruebas – Módulo de Proveedor

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 51 pagina 117, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.32.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 49 pagina 115.

Figura N.º 3. 52 Grafo de flujo módulo de proveedor



Fuente: Elaboración propia

Tabla N.º 3. 55 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 56 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevo proveedor.	Registra nuevo proveedor con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No crea nuevo proveedor.	Buscar proveedor, editar, desactivar y reporte
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.32.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 57 Resultados de la implementación del módulo proveedor

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar proveedores?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo de proveedor?	Cumple
3	¿Cuándo ingresa al módulo proveedor puede registra generar reporte de un proveedor?	Cumple
4	¿Al registrar un nuevo proveedor es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo proveedor, cumple?	Cumple

Fuente: Elaboración propia

### 3.32.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de proveedor en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.33. Fase 3 Análisis y diseño - Módulo de Clientes

Es considerado como de prioridad alto la cual es Crear el Registro de Clientes con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 58 Módulo Cliente

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de clientes.	Diseño del diagrama UML	07/04/2020	11/04/2020	5 días
	Codificación	09/04/2020	10/04/2020	Completado
	Pruebas	11/04/2020	11/04/2020	Completado

Fuente: Elaboración propia

#### 3.33.1. Caso de Uso

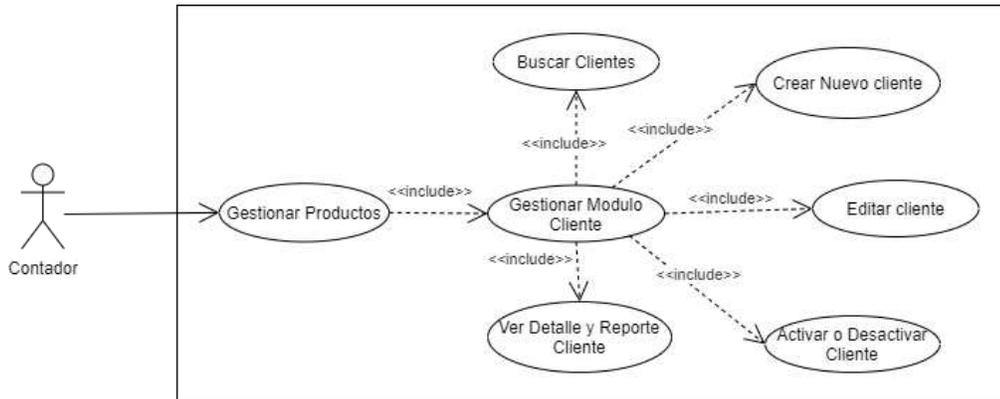
En la siguiente tabla se describe como realiza el registro de clientes.

Tabla N.º 3. 59 Descripción de caso de uso del Módulo cliente

Nro.	Descripción
1	Registro y actualización de clientes
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 53 Caso de uso del Módulo cliente



Fuente: Elaboración propia

Tabla N.º 3. 60 Inicio de sesión, ingresar al módulo cliente

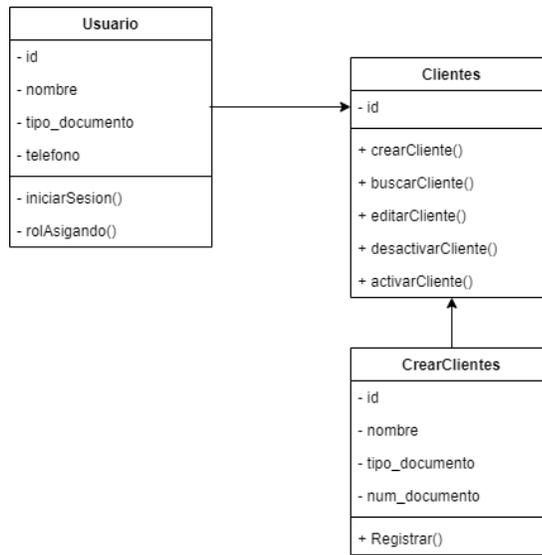
Nombre.	Registrar y actualizar plan de cuentas
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar un nuevo cliente deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de clientes para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.33.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar clientes

Figura N.º 3. 54 Diagrama de clases del Módulo de cliente

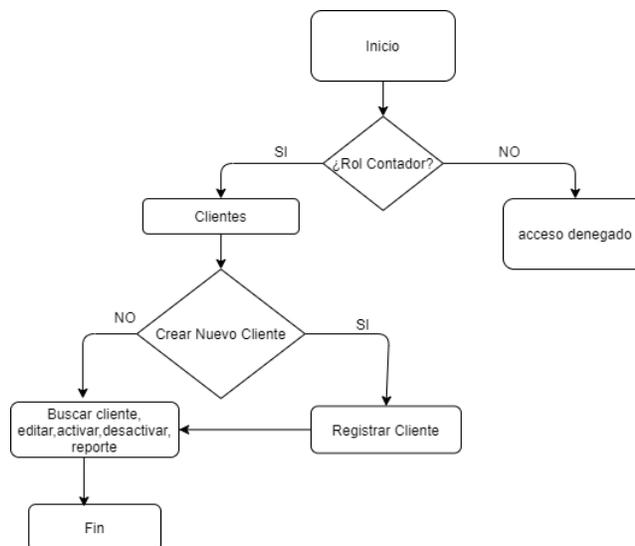


Fuente: Elaboración propia

### 3.33.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar modulo cliente, el cual se observa en la siguiente figura:

Figura N.º 3. 55 Diagrama de actividades o flujo modulo cliente

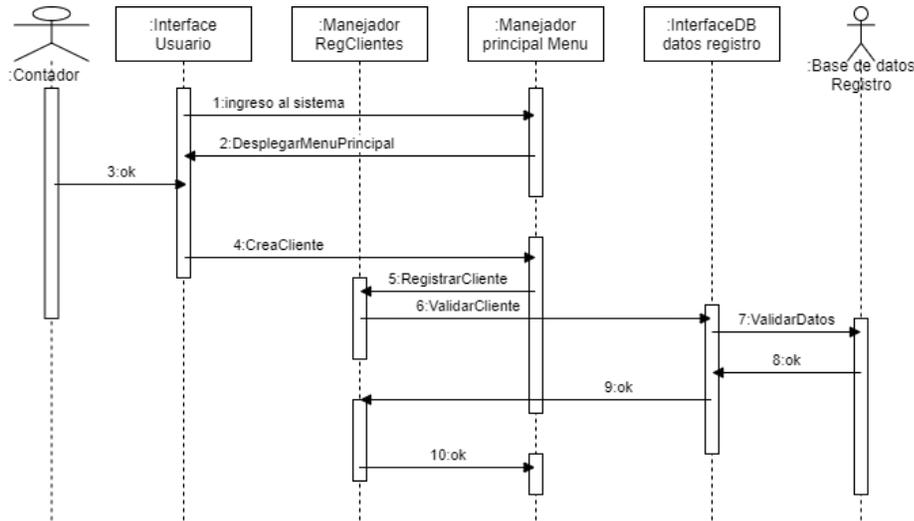


Fuente: Elaboración propia

### 3.33.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de registro de clientes / actualizar como se observa en la siguiente figura:

Figura N.º 3. 56 Diagrama de secuencia Modulo de registro cliente

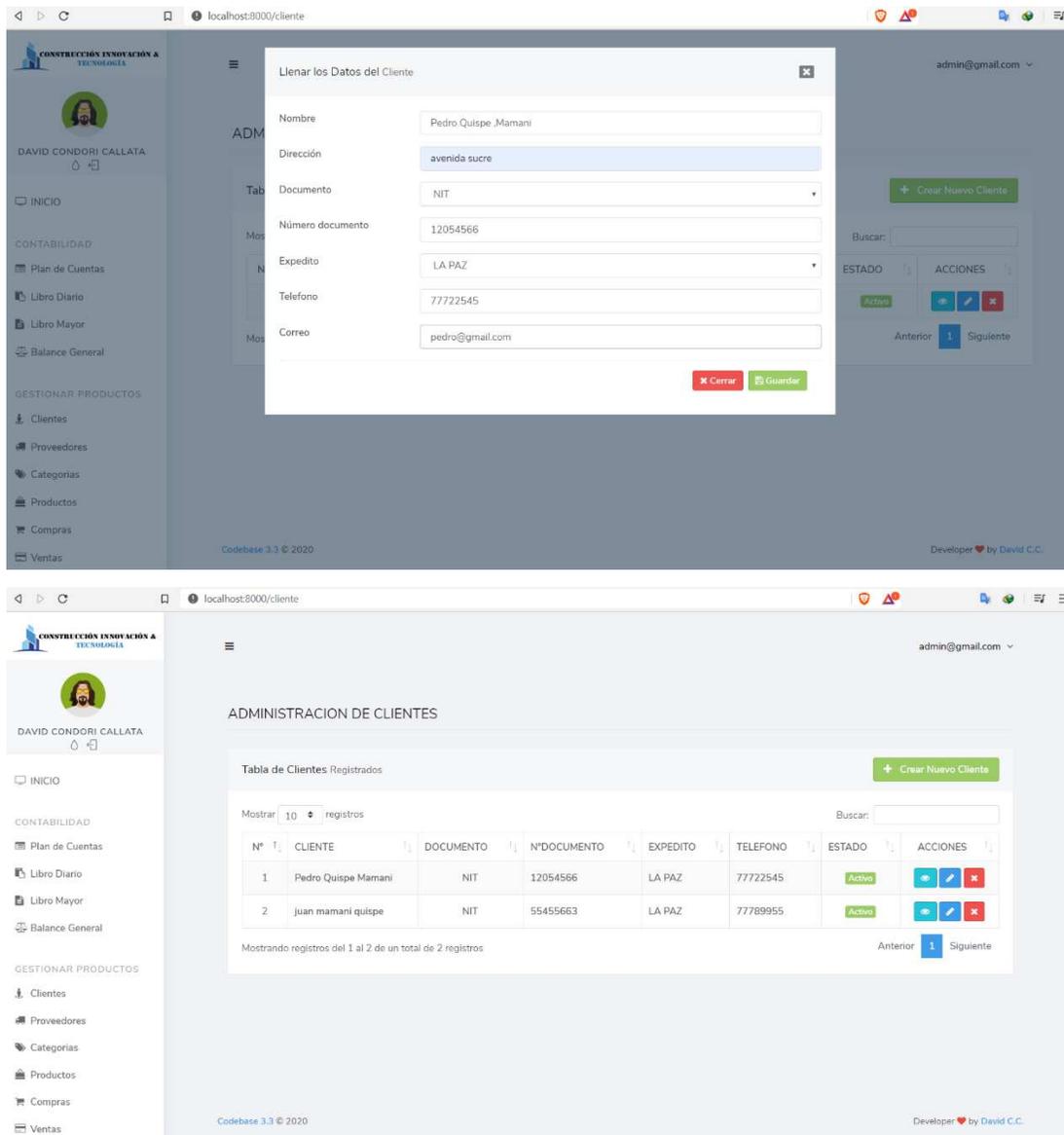


Fuente: Elaboración propia

### 3.34. Fase 4 Programación – Módulo de Clientes

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 57 Entornos gráficos del Módulo de clientes/Registros



Fuente: Elaboración propia

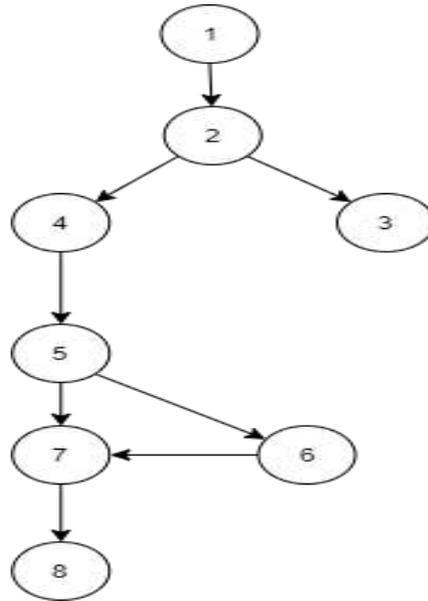
### 3.35. Fase 5 Implementación y pruebas – Módulo de Clientes

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 57 pagina 124, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.35.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 55 pagina 122.

Figura N.º 3. 58 Grafo de flujo módulo cliente



Fuente: Elaboración propia

Tabla N.º 3. 61 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 62 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevo cliente.	Registra nuevo cliente con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No registra nuevo cliente	Buscar cliente, editar, desactivar, activar y reporte
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.35.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 63 Resultados de la implementación del módulo cliente

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar clientes?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo del cliente?	Cumple
3	¿Cuándo ingresa al módulo de cliente puede generar reporte?	Cumple
4	¿Al registrar un nuevo cliente es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo cliente, cumple?	Cumple

Fuente: Elaboración propia

### 3.35.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de cliente en el momento de la implementación pasando primeramente por la prueba de integración,

luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.36. Fase 3 Análisis y diseño - Módulo de Compras

Es considerado como de prioridad alto la cual es Crear el Registro de las compras con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 64 Módulo de Compras

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de compras	Diseño del diagrama UML	12/04/2020	14/04/2020	Completado
	Codificación	15/04/2020	18/04/2020	Completado
	Pruebas	19/04/2020	19/04/2020	Completado

Fuente: Elaboración propia

#### 3.36.1. Caso de Uso

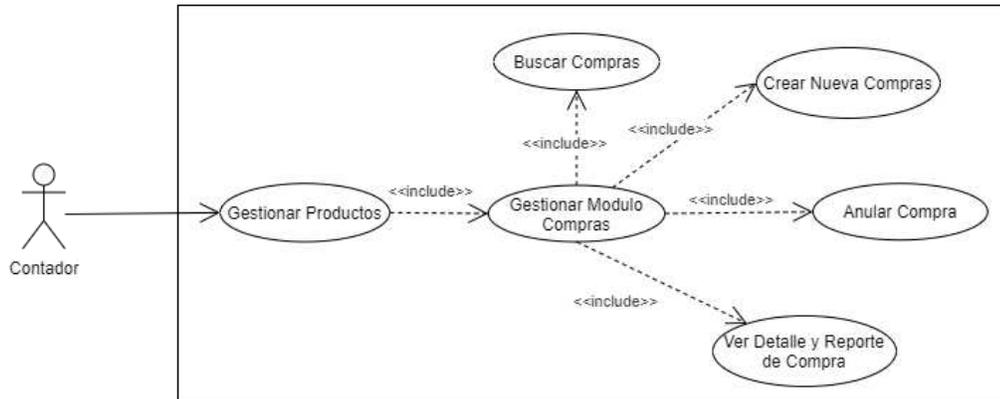
En la siguiente tabla se describe como realiza el registro de compras.

Tabla N.º 3. 65 Descripción de caso de uso del Módulo de compras

Nro.	Descripción
1	Registro y anulación las compras
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 59 Caso de uso del Módulo de compras



Fuente: Elaboración propia

Tabla N.º 3. 66 Inicio de sesión, ingresar al módulo de compras

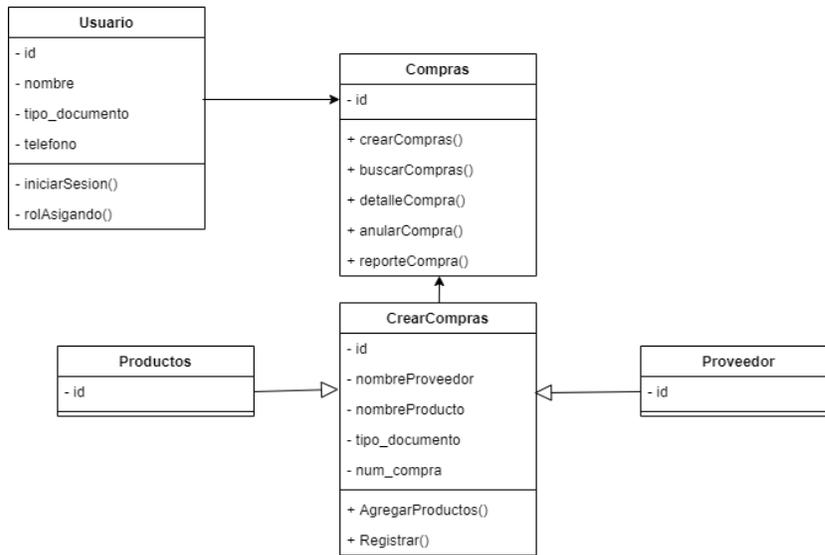
Nombre.	Registrar y actualizar plan de cuentas
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar una nueva compra deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de compras para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.36.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar compras.

Figura N.º 3. 60 Diagrama de clases del Módulo de compras

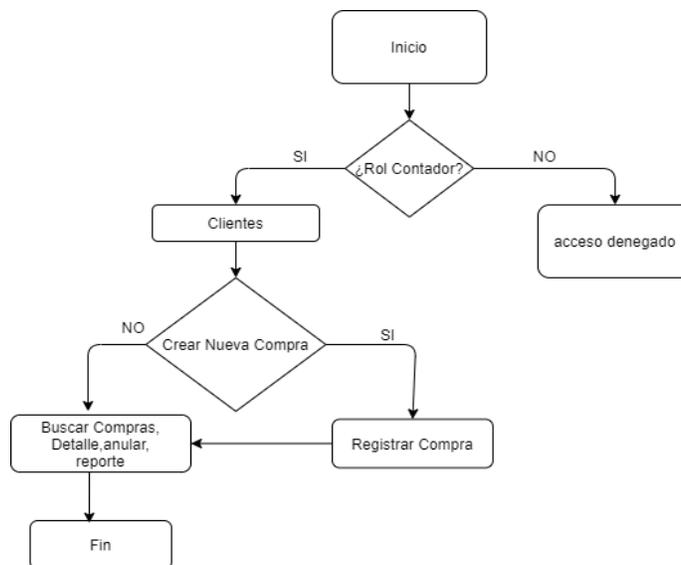


Fuente: Elaboración propia

### 3.36.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar módulo de compras y autenticación del usuario, el cual se observa en la siguiente figura:

Figura N.º 3. 61 Diagrama de actividades o flujo módulo de compras

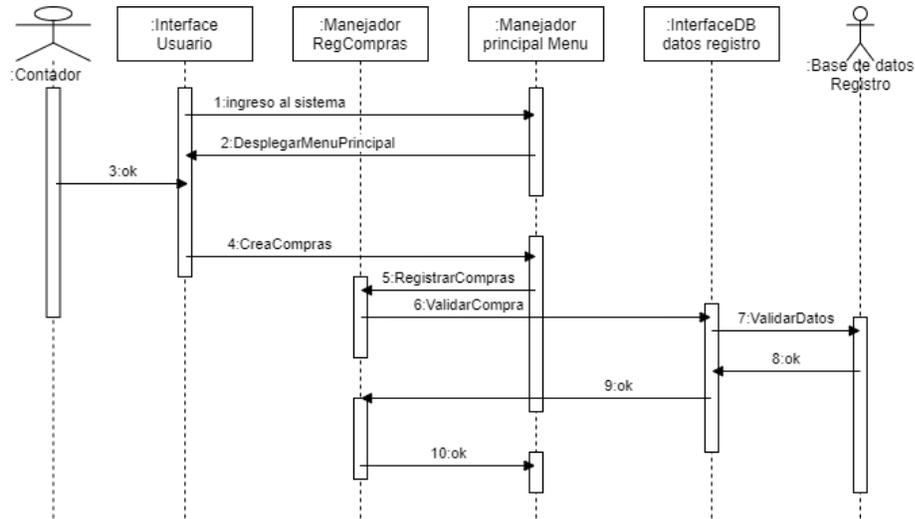


Fuente: Elaboración propia

### 3.36.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de registro de compras como se observa en la siguiente figura:

Figura N.º 3. 62 Diagrama de secuencia módulo de compras

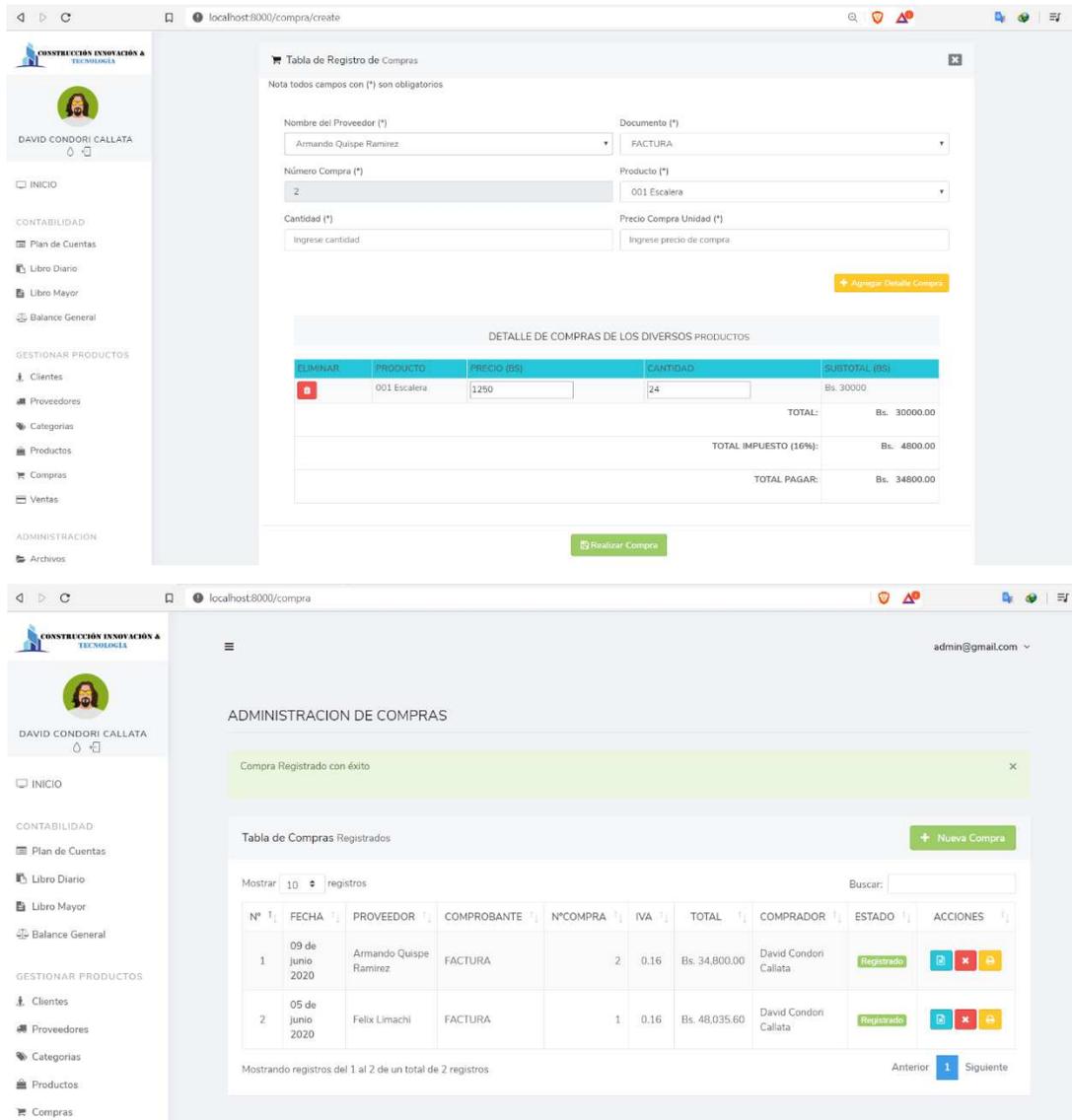


Fuente: Elaboración propia

### 3.37. Fase 4 Programación – Módulo de Compras

Se obtiene el siguiente resultado como se observa en la siguiente Figura.

Figura N.º 3. 63 Entornos gráficos del Módulo de compras



Fuente: Elaboración propia

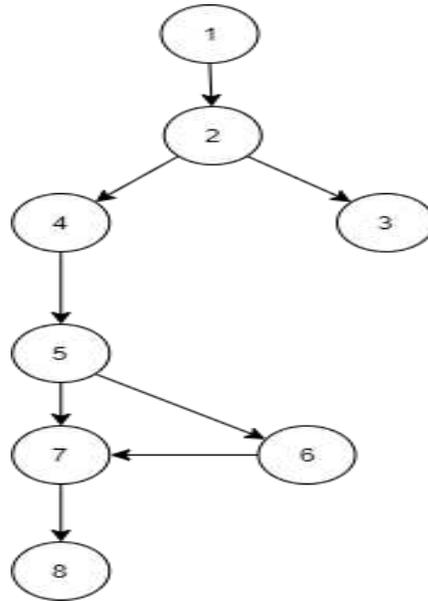
### 3.38. Fase 5 Implementación y pruebas – Módulo de Compras

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 63 pagina 131, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.38.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 61 pagina 129.

Figura N.º 3. 64 Grafo de flujo módulo de compras



Fuente: Elaboración propia

Tabla N.º 3. 67 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 68 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevas compras.	Registra nueva compra con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No realiza una compra	Buscar compras, detalle, anular y reporte
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.38.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 69 Resultados de la implementación del módulo de compras

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar nuevas compras?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo de compras?	Cumple
3	¿Cuándo ingresa al módulo compras puede ser los detalles y generar reporte?	Cumple
4	¿Al registrar una nueva compra es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo de compras, cumple?	Cumple

Fuente: Elaboración propia

### 3.38.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de compras en el momento de la implementación pasando primeramente por la prueba de

integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.39. Fase 3 Análisis y diseño - Módulo de Ventas

Es considerado como de prioridad alto la cual es Crear el Registro de ventas con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 70 Módulo de Ventas

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
		20/04/2020	27/04/2020	4 días
Crear registro de ventas.	Diseño del diagrama UML	20/04/2020	22/04/2020	Completado
	Codificación	23/04/2020	26/04/2020	Completado
	Pruebas	27/04/2020	27/04/2020	Completado

Fuente: Elaboración propia

#### 3.39.1. Caso de Uso

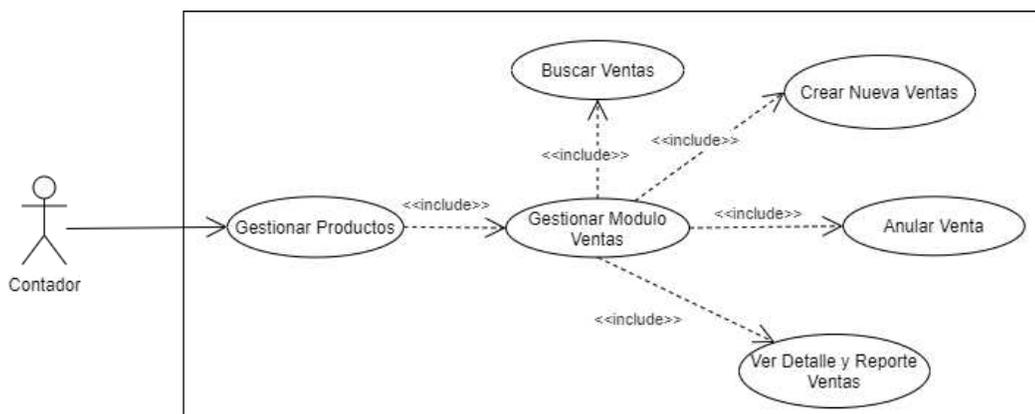
En la siguiente tabla se describe como realiza el registro de ventas.

Tabla N.º 3. 71 Descripción de caso de uso del Módulo de ventas

Nro.	Descripción
1	Registro y anulación de las ventas.
2	Y las acciones necesarias que contendrá

Fuente: Elaboración propia

Figura N.º 3. 65 Caso de uso del Módulo de ventas



Fuente: Elaboración propia

Tabla N.º 3. 72 Inicio de sesión, ingresar al módulo de ventas

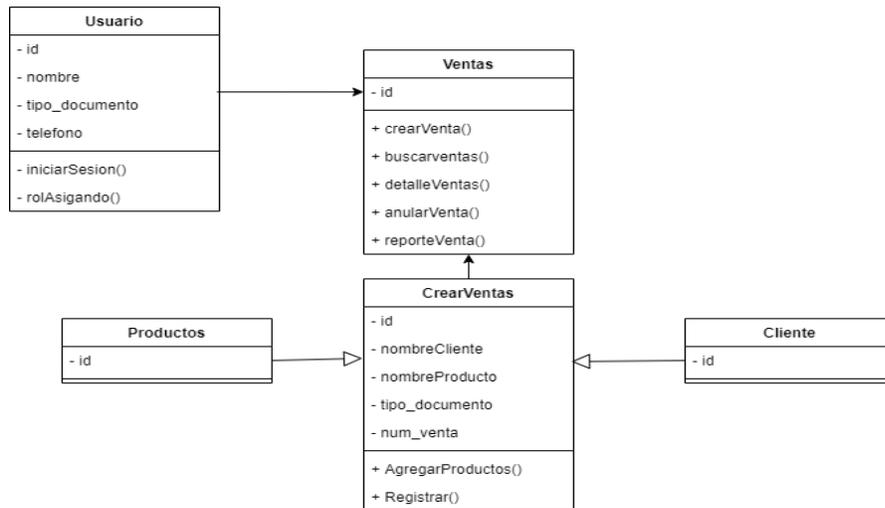
Nombre.	Registrar y anular ventas
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda registrar una nueva venta o anular deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de ventas para que pueda realizar los debidos registros.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.39.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento Registrar ventas realizadas.

Figura N.º 3. 66 Diagrama de clases del Módulo de ventas

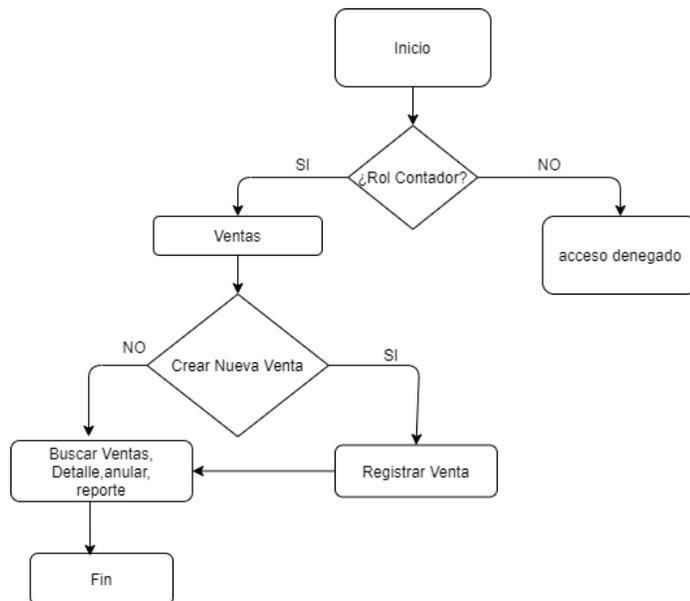


Fuente: Elaboración propia

### 3.39.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar módulo de ventas, el cual se observa en la siguiente figura:

Figura N.º 3. 67 Diagrama de actividades o flujo módulo de ventas

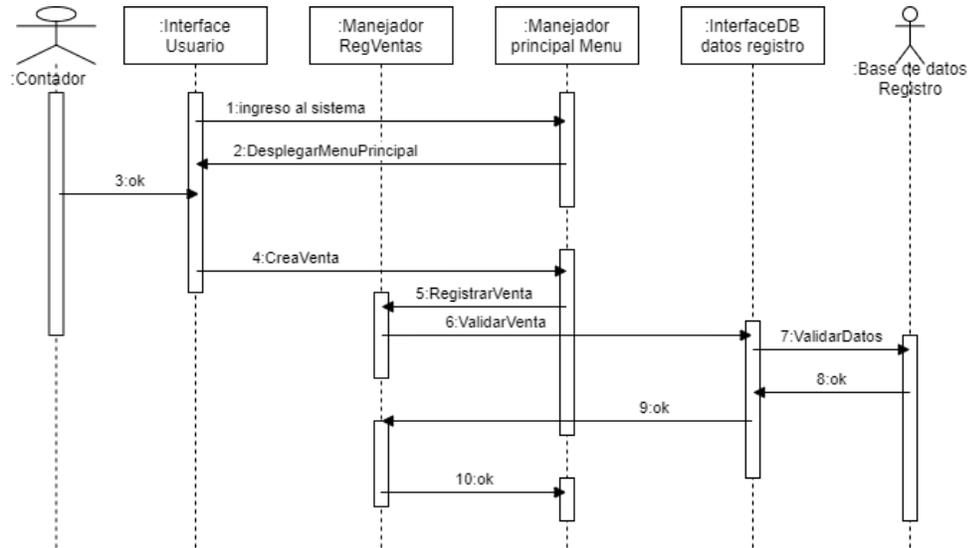


Fuente: Elaboración propia

### 3.39.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de registro de nuevas compras como se observa en la siguiente figura:

Figura N.º 3. 68 Diagrama de secuencia Modulo de registro ventas

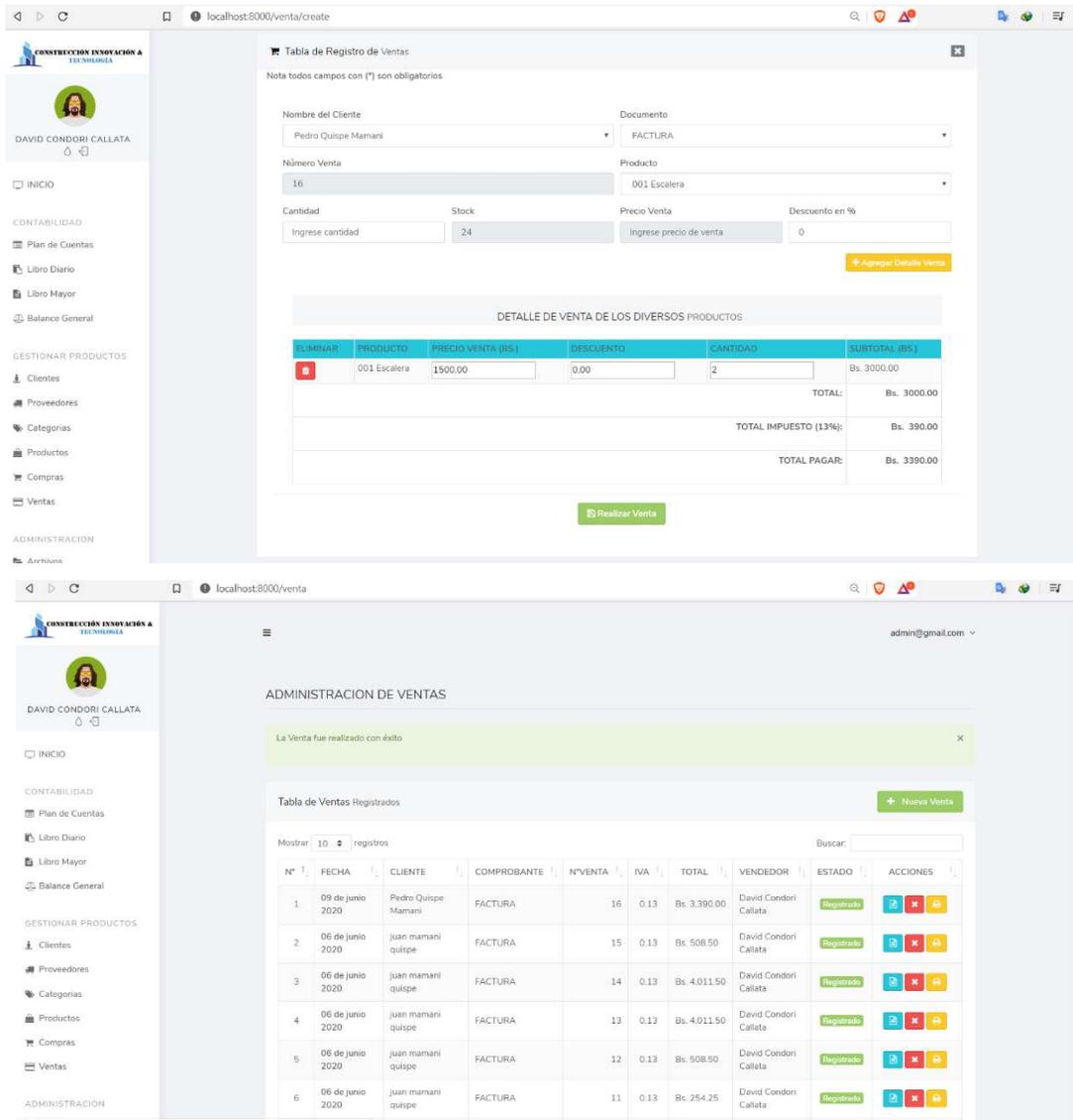


Fuente: Elaboración propia

### 3.40. Fase 4 Programación – Módulo de Ventas

Se obtiene el siguiente resultado como se observa en la siguiente Figura.

Figura N.º 3. 69 Entornos gráficos del Módulo de ventas



Fuente: Elaboración propia

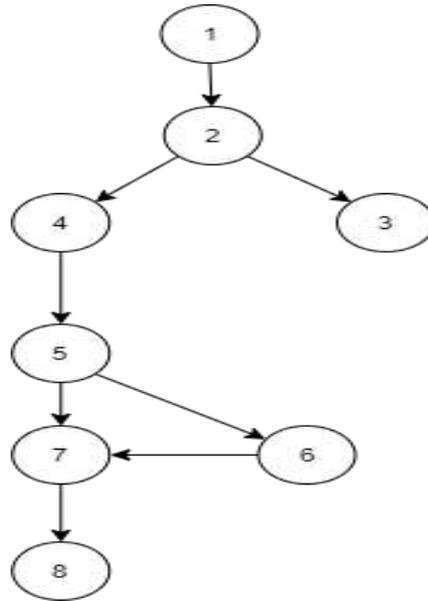
### 3.41. Fase 5 Implementación y pruebas – Módulo de Ventas

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 69 pagina 138, posteriormente se realizan las pruebas de caja blanca y negra.

### 3.41.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 67 pagina 136.

Figura N.º 3. 70 Grafo de flujo módulo de ventas



Fuente: Elaboración propia

Tabla N.º 3. 73 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$		$V(G) = 8 - 8 + 2 = 2$	
$V(G) = \text{Nodos Predicados} + 1$		$V(G) = 2 + 1 = 3$	
$V(G) = \text{Regiones}$		$V(G) = 2$	
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 74 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede registra nuevas ventas.	Registra nueva venta con éxito
1,2,4,5,7,8	-> Rol Contador	->Si es contador ->No registra nueva venta.	Buscar ventas, detalle, anular y facturar
1,2,3	-> Rol Contador	->No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.41.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 75 Resultados de la implementación del módulo de ventas

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede registrar nuevas ventas?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo ventas?	Cumple
3	¿Cuándo el usuario ingresa al módulo ventas puede emitir facturas?	Cumple
4	¿Al registrar una nueva venta es validado en la base de datos?	Cumple
5	¿Tiene el CRUD necesario el módulo de ventas, cumple?	Cumple

Fuente: Elaboración propia

### 3.41.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de ventas en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.42. Fase 3 Análisis y diseño - Generar Facturas o Recibos

Es considerado como de prioridad alto la cual es generar las facturas o recibos en los módulos de compras y ventas, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 76 Generar facturas o Recibos

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear la generación de facturas o recibos	Diseño del diagrama UML	28/04/2020	28/04/2020	Completado
	Codificación	29/04/2020	30/04/2020	Completado
	Pruebas	01/05/2020	01/05/2020	Completado

Fuente: Elaboración propia

#### 3.42.1. Caso de Uso

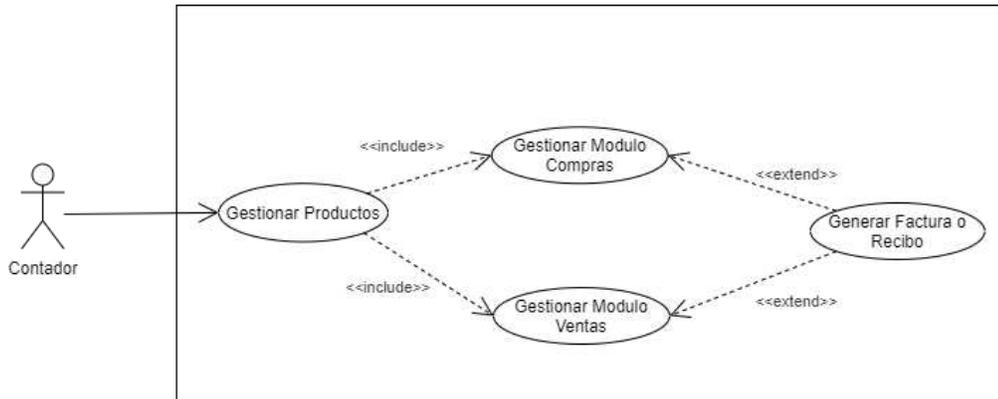
En la siguiente tabla se describe como generar las facturas para los módulos correspondientes.

Tabla N.º 3. 77 Descripción de caso de uso Facturas o Recibos

Nro.	Descripción
1	Generar facturas o recibos en los módulos de compras y ventas.

Fuente: Elaboración propia

Figura N.º 3. 71 Caso de uso Generar Facturas o Recibos



Fuente: Elaboración propia

Tabla N.º 3. 78 Usuario Genera Facturas y Recibos

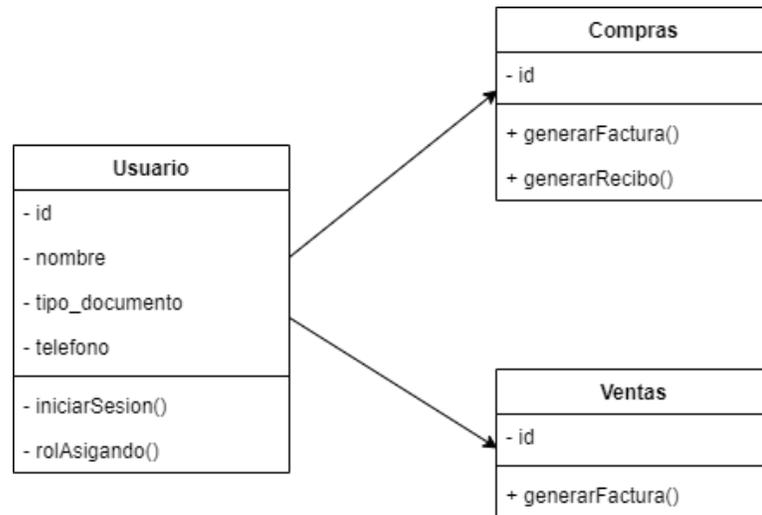
Nombre.	Generar Facturas o Recibos
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda generar una factura o recibo deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de compras o ventas para poder generar las facturas o recibos.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.42.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento facturas o recibos.

Figura N.º 3. 72 Diagrama de clases de facturas o recibos

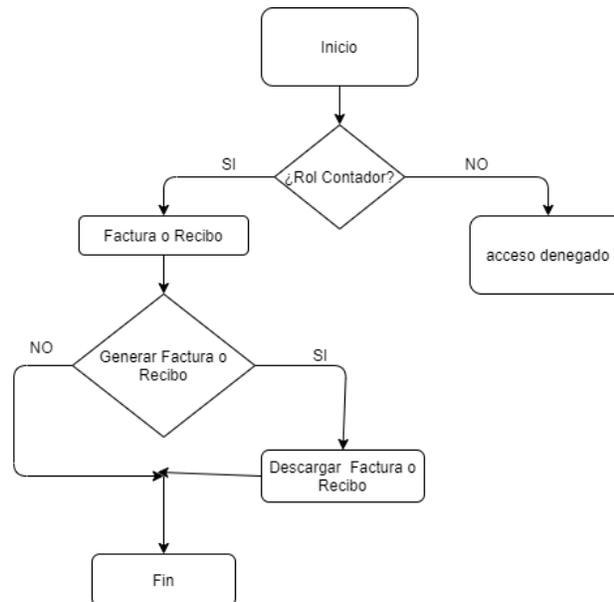


Fuente: Elaboración propia

### 3.42.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para generar las facturas o recibos, el cual se observa en la siguiente figura:

Figura N.º 3. 73 Diagrama de actividades o flujo de facturas o recibos

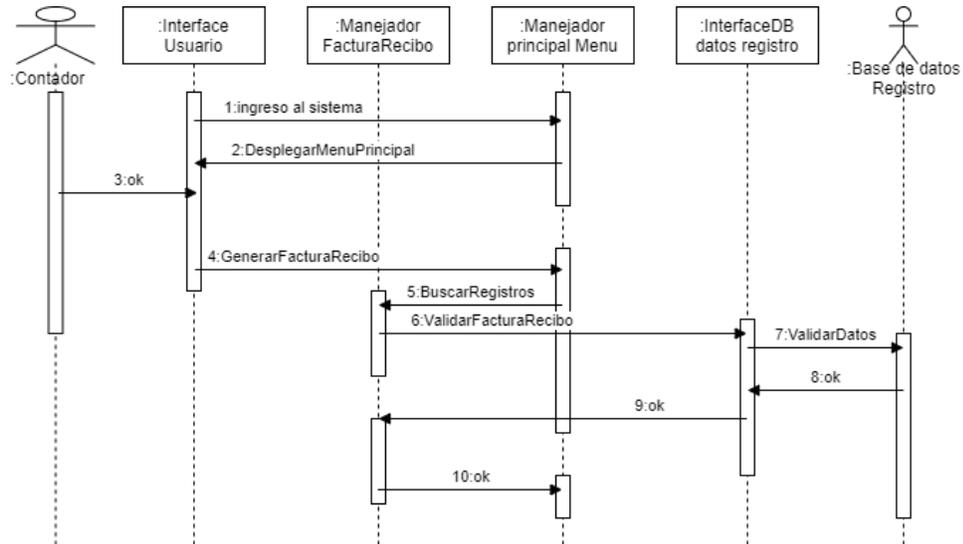


Fuente: Elaboración propia

### 3.42.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos de facturas o recibos como se observa en la siguiente figura:

Figura N.º 3. 74 Diagrama de secuencia para generar facturas o recibos



Fuente: Elaboración propia

### 3.43. Fase 4 Programación – Generar Facturas o Recibos

Se obtiene el siguiente resultado como se observa en la siguiente figura.

Figura N.º 3. 75 Entornos gráficos de facturas o recibos



De Construcción Innovación & Tecnología  
 Casa Matriz  
 Av. San José Nro. 2217, esquina Av. Florida  
 Zona/Barrio: Zona Munaypata  
 Teléfono/Celular: 78910189  
 La Paz-Bolivia

NIT:	4331911018
FACTURA N°:	<b>16</b>
AUTORIZACION N°:	255554583201

ORIGINAL

Actividades de contabilidad, teneduría de libro y auditoría; asesoramientos en materia de impuestos

## FACTURA

<b>Lugar:</b> La Paz . 09 de junio 2020		<b>NIT/CI:</b> 12054566		
<b>Señor(es):</b> Pedro Quispe Mamani				
CANTIDAD	DETALLE	PRECIO UNITARIO	DESCUENTO(%)	SUBTOTAL
2	Escalera	Bs. 1500.00	0.00	Bs. 3,000.00
<b>TOTAL:</b>				<b>Bs. 3,000.00</b>
<b>IVA:</b>				<b>Bs. 390.00</b>
<b>TOTAL PAGAR:</b>				<b>Bs. 3,390.00</b>
Son: TRES MIL TRESCIENTOS NOVENTA CON 00/100 BOLIVIANOS				

**Código de Control:** EE-4A-DF-57-AE  
**Fecha Límite de Emisión:** 17/01/2021



ESTA FACTURA CONTRIBUYE AL DESARROLLO DEL PAIS. EL USO ILICITO DE ESTA SERA SANCIONADO DE ACUERDO A LA LEY.  
 Ley N° 453: La interrupción del servicio debe comunicarse con anterioridad a las Autoridades que correspondan y a los afectados.

CONSTRUCCION INNOVACION & TECNOLOGIA  
 LOS ANDES-EL ALTO

Fecha: 2020-06-09 10:48:35

### COMPROBANTE DE COMPRAS

**Nombre de Proveedor :** Armando Quispe Ramirez. **Fecha :** 2020-06-09 10:46:20 **FACTURA N° :** 2  
**Telefono/Celular :** 2254551 **Correo :** armando123@gmail.com  
**Dirección :** eloy salmon  
**Comprador :** David Condori Callata

CANTIDAD	DETALLE	PRECIO UNITARIO	SUBTOTAL
24	Escalera	Bs. 1250.00	Bs. 30,000.00
<b>TOTAL:</b>			<b>Bs. 30,000</b>
<b>IVA (16%):</b>			<b>Bs. 4,800.00</b>
<b>TOTAL PAGAR:</b>			<b>Bs. 34,800.00</b>
Son: TREINTA Y CUATRO MIL OCHOCIENTOS CON 00/100 BOLIVIANOS			

CONTADOR

GERENTE

Fuente: Elaboración propia

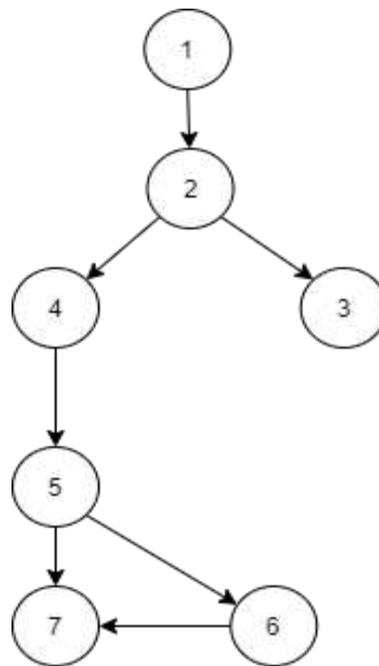
### 3.44. Fase 5 Implementación y pruebas – Generar Facturas o Recibos

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 75 pagina 145, posteriormente se realizan las pruebas de caja blanca y negra.

#### 3.44.1. Prueba de Caja Blanca

La prueba que se realizó para generar facturas o recibos es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 73 pagina 143.

Figura N.º 3. 76 Grafo de flujo de facturas o recibos



Fuente: Elaboración propia

Tabla N.º 3. 79 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
7	7	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$			$V(G) = 7 - 7 + 2 = 2$
$V(G) = \text{Nodos Predicados} + 1$			$V(G) = 2 + 1 = 3$
$V(G) = \text{Regiones}$			$V(G) = 2$
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 80 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7	-> Rol Contador	-> Si usuario debe estar registrado y tener el rol de contador. -> Si usuario está registrado, usuario logueado puede generar facturas o recibos.	Generar facturas o recibos con éxito
1,2,4,5,7	-> Rol Contador	-> Si es contador -> No genera factura o recibo	Ver detalles de factura o recibo
1,2,3	-> Rol Contador	-> No es contador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.44.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 81 Resultados de la implementación de facturas o recibos

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede generar las facturas o recibos?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar a los respectivos módulos para generar facturas o recibos?	Cumple
3	¿Cuándo ingresa al módulo ventas o compras genera facturas o recibos?	Cumple
4	¿Al registrar una nueva venta emite una factura?	Cumple

Fuente: Elaboración propia

### 3.44.3. Resultados

Después de codificar, se realizó las pruebas necesarias para generar las respectivas facturas o recibos en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

### 3.45. Fase 3 Análisis y diseño - Módulo de Archivos

Es considerado como de prioridad media y baja la cual es módulo de archivos con todas las acciones necesarias de CRUD, se diseñó los diagramas que UML nos proporciona en cada requisito, para luego pasar a la codificación.

Tabla N.º 3. 82 Módulo de archivos

REQUISITO	TAREA	Inicio	Fin	Duración
		Desde	Hasta	Estado
Crear registro de archivos de respaldo.	Diseño del diagrama UML	02/05/2020	02/05/2020	4 días
	Codificación	03/05/2020	04/05/2020	Completado
	Pruebas	05/05/2020	05/05/2020	Completado

Fuente: Elaboración propia

### 3.45.1. Caso de Uso

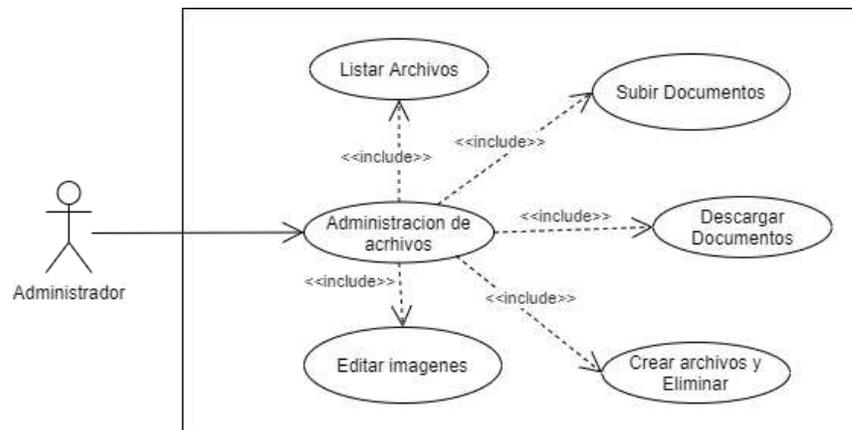
En la siguiente tabla se describe como realiza el almacenamiento de archivos.

Tabla N.º 3. 83 Descripción de caso de uso del Módulo archivos

Nro.	Descripción
1	Subir documentos a los archivos creados en el servidor.
2	Descargar los archivos según se soliciten.

Fuente: Elaboración propia

Figura N.º 3. 77 Caso de uso del Módulo de archivos



Fuente: Elaboración propia

Tabla N.º 3. 84 Inicio de sesión, Modulo de archivos

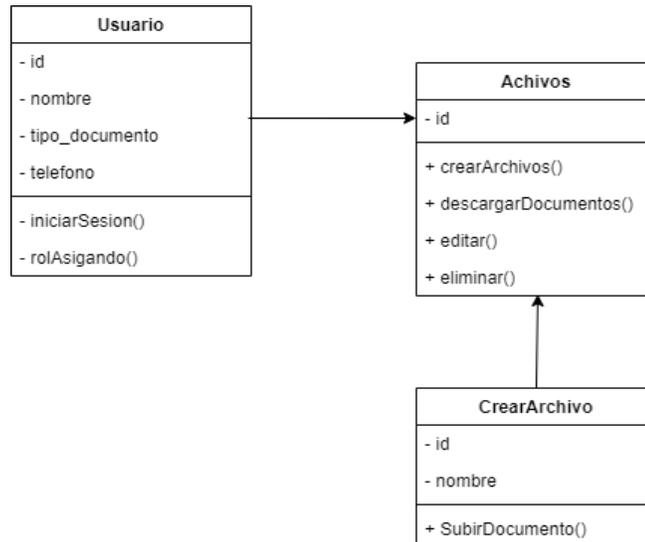
Nombre.	Registrar y actualizar plan de cuentas
Actores	Usuario. Sistema.
Propósitos	Para que el usuario pueda almacenar un documento deberá estar autorizado y con el rol asignado e iniciar sesión en el sistema. <ul style="list-style-type: none"> <li>Evento del actor: deberá iniciar sesión con el nombre de usuario y la contraseña</li> </ul>
Flujo principal	<ul style="list-style-type: none"> <li>Evento del sistema: Valida ambos parámetros enviados, ingresa al sistema con los permisos correspondientes y puede ingresar al módulo de archivos para que pueda subir o descargar los diversos documentos.</li> </ul>
Flujo alternativo	Si no se encuentra el usuario o contraseña no coinciden, se manda un mensaje "Usuario o Contraseña incorrecta"
Precondición	El usuario debe estar registrado en la base de datos, tanto su contraseña.
Postcondición	Solo un usuario autorizado ingresa al sistema.
Presunción	Si el usuario no se encuentra registrado no podrá ingresar al sistema.

Fuente: Elaboración propia

### 3.45.2. Diagrama de clases

Se diseña el diagrama de clases para el requerimiento de almacenamiento de documentos en diversos archivos.

Figura N.º 3. 78 Diagrama de clases del Módulo archivos

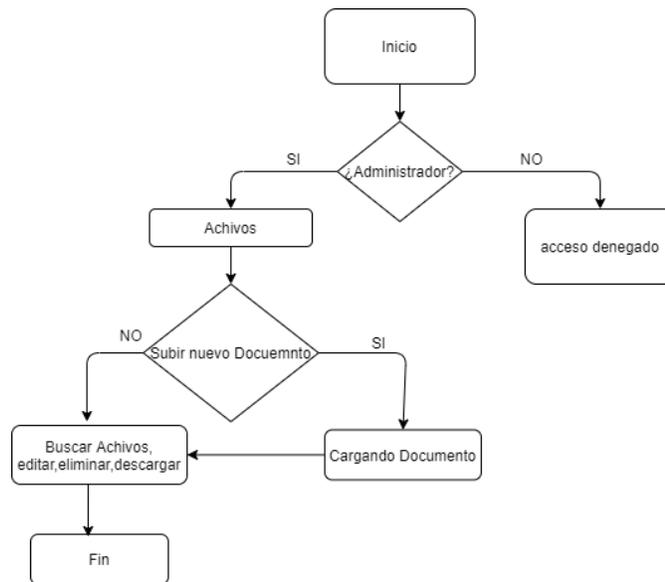


Fuente: Elaboración propia

### 3.45.3. Diagrama de actividades o de flujo

Se diseña el diagrama de actividades o flujo para registrar los documentos en cada archivo correspondiente, el cual se observa en la siguiente figura:

Figura N.º 3. 79 Diagrama de actividades o flujo modelo Archivos

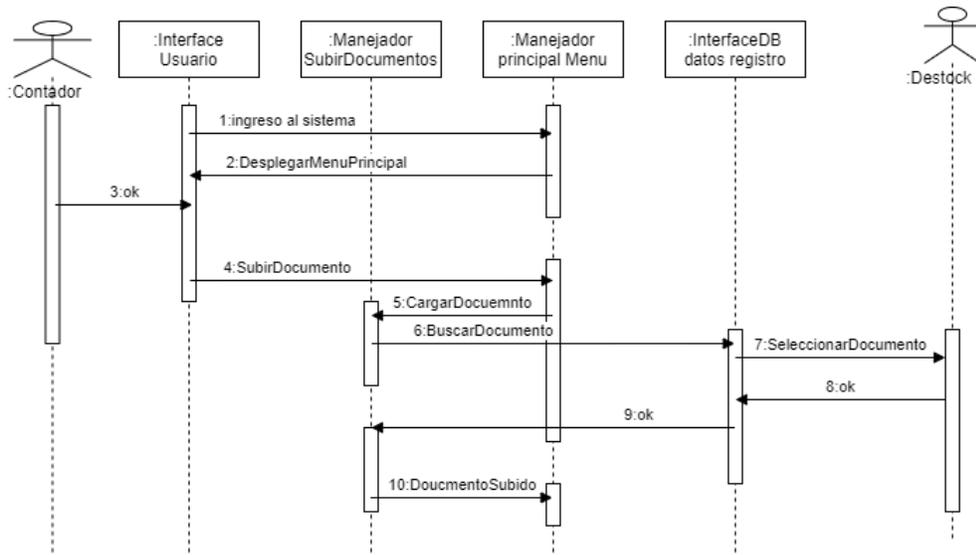


Fuente: Elaboración propia

### 3.45.4 Diagrama de secuencia

Se diseñó el diagrama de secuencias para los requerimientos del módulo de archivos como se observa en la siguiente figura:

Figura N.º 3. 80 Diagrama de secuencia Modulo de archivos

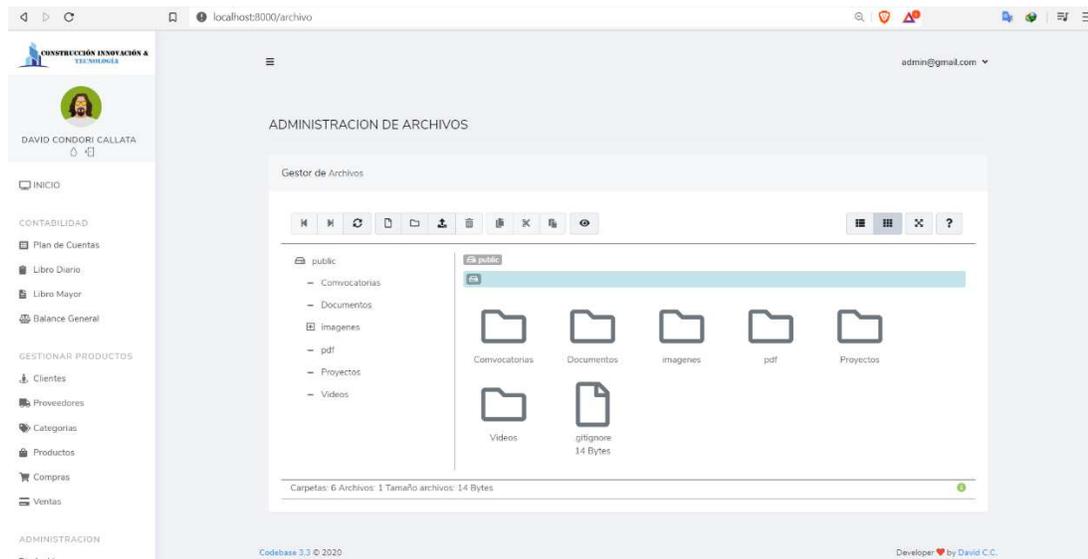


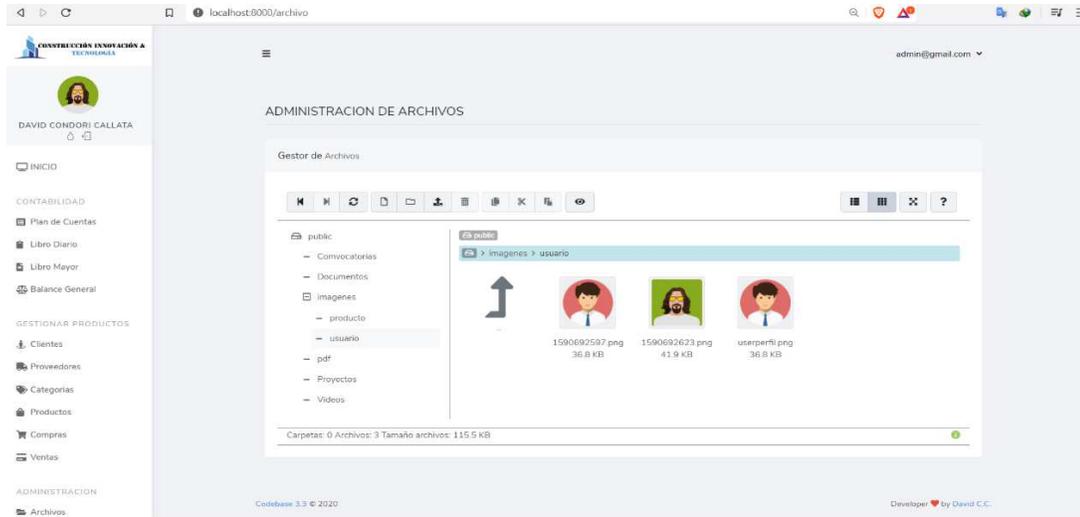
Fuente: Elaboración propia

### 3.46. Fase 4 Programación – Módulo de Archivos

Se obtiene el siguiente resultado como se observa en la síguete figura.

Figura N.º 3. 81 Entornos gráficos del Módulo de archivos





Fuente: Elaboración propia

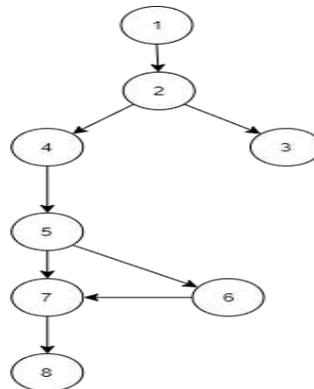
### 3.47. Fase 5 Implementación y pruebas – Módulo de Archivos

Se realiza la implementación una vez finalizado la programación como se puede observar en las Figuras N.º 3. 91 pagina 153, posteriormente se realizan las pruebas de caja blanca y negra.

#### 3.47.1. Prueba de Caja Blanca

La prueba que se realizó para este módulo es calcular la complejidad ciclomática del diagrama de flujo de la Figura N.º 3. 79 pagina 151.

Figura N.º 3. 82 Grafo de flujo módulo de archivos



Fuente: Elaboración propia

Tabla N.º 3. 85 Cálculo de complejidad

NODOS	ARISTAS	REGIONES	NODOS PREDICADO
8	8	2	1
$V(G) = \text{Aristas} - \text{Nodos} + 2$			$V(G) = 8 - 8 + 2 = 2$
$V(G) = \text{Nodos Predicados} + 1$			$V(G) = 2 + 1 = 3$
$V(G) = \text{Regiones}$			$V(G) = 2$
Complejidad ciclomática de 3 que nos indica realizar 3 pruebas			

Fuente: Elaboración propia

Tabla N.º 3. 86 Caminos independientes

CAMINO	ENTRADA DATOS	PRUEBA	SALIDA
1,2,4,5,6,7,8	->Administrador	-> Si usuario debe estar registrado y tener el rol de Administrador. -> Si usuario está registrado, usuario logueado puede subir nuevos documentos a respaldar.	Sube documentos con éxito
1,2,4,5,7,8	-> Administrador	->Si es Administrador. ->No sube documentos	Buscar archivos, documentos existentes, editar, eliminar y descargar
1,2,3	-> Administrador	->No es Administrador acceso denegado	Acceso denegado

Fuente: Elaboración propia

### 3.47.2. Prueba de Caja Negra

En esta prueba únicamente nos enfocamos en los requerimientos de software y especificaciones funcionales.

Tabla N.º 3. 87 Resultados de la implementación del módulo plan de cuentas

NRO.	DESCRIPCIÓN DE LOS CASOS DE PRUEBA	RESULTADOS
1	¿Solo un usuario autorizado puede subir o descargar documentos?	Cumple
2	¿Después de que el usuario inicia sesión puede ingresar al módulo de archivos?	Cumple
3	¿Cuándo ingresa al módulo de archivos puede editar o eliminar archivos o documentos?	Cumple

Fuente: Elaboración propia

### 3.47.3. Resultados

Después de codificar, se realizó las pruebas necesarias al Módulo de archivos en el momento de la implementación pasando primeramente por la prueba de integración, luego se aplicaron las pruebas de caja blanca y caja negra, obteniendo resultados satisfactorios.

## **4. MÉTRICAS DE CALIDAD, SEGURIDAD Y ESTIMACIÓN DE COSTOS**

## 4.1. Norma ISO/IEC 25010

### 4.1.1. Adecuación funcional

Es la capacidad del software de proveer las funciones para satisfacer las necesidades explícitas e implícitas, este atributo del sistema no puede ser calculado de forma directa, por esa razón se utiliza el cálculo de métrica de punto función, que contiene cinco características de información. Los valores de información son definidos de la siguiente forma:

- **Número de entradas de usuario:** Cada entrada de usuario proporciona diferentes datos orientados a la aplicación. Las entradas se diferencian de las peticiones de forma separadas.
- **Número de salidas de usuario:** Cada salida proporciona al usuario información orientada a la aplicación. Las salidas se refieren a informes, pantallas, mensajes de error entre otros.
- **Número de peticiones de usuario:** Es una entrada interactiva que es producida por la respuesta del software inmediata en forma de salida interactiva. Las peticiones son contadas de forma separada.
- **Numero de archivos:** Un grupo lógico de datos que puede ser parte de una base de datos (tablas de base de datos).
- **Numero de interfaces externas:** Se cuenta todas las interfaces legibles que se utilizan para transmitir la información.

Para calcular los puntos función se emplea la siguiente formula:

$$PF = \text{Cuenta Total} * (0.65 + 0.01 * \sum F_i)$$

Donde:

**PF:** Medida de la adecuación funcional.

**Cuenta Total:** Es la suma de los siguientes datos: N° de entradas, N° de salidas, N° de peticiones, N° de archivos y N° de interfaces externas.

**0.65:** Confiabilidad del proyecto, varía del 1% al 100% (0 a 1).

**0.01:** Error mínimo aceptable de complejidad.

$F_i$ : Son los valores de ajuste de complejidad, donde ( $1 \leq i \leq 14$ ).

Analizando todas las interfaces que tiene el sistema se obtuvieron los siguientes datos:

Tabla N.º 4. 1 Parámetros de medida y su cantidad

NRO.	PARAMETROS DE MEDIDA	CANTIDAD
1	N° de entradas de usuario	26
2	N° de salidas de usuario	16
3	N° de peticiones de usuario	31
4	N° de archivos	18
5	N° de interfaces externas	2

Fuente: Elaboración propia

Una vez obtenida la información se procede a calcular la cuenta total con el factor de ponderación media que se muestra en la siguiente tabla:

Tabla N.º 4. 2 Parámetros de medida y su cantidad

PARAMETROS DE MEDIDA	CANTIDAD		FACTOR DE PONDERACIÓN		TOTAL
Nº de entradas de usuario	26	*	4	=	104
Nº de salidas de usuario	16	*	5	=	80
Nº de peticiones de usuario	31	*	5	=	155
Nº de archivos	18	*	10	=	180
Nº de interfaces externas	2	*	7	=	14
<b>CUENTA TOTAL</b>				=	<b>533</b>

Fuente: Elaboración propia

La cuenta total de los puntos función obtenida se debe ajustar en función a las características ambientales del sistema. Los valores de ajuste de complejidad  $F_i$  basados en las respuestas a las preguntas formuladas de la siguiente tabla:

Tabla N.º 4. 3 Valores de ajuste de complejidad

NRO	FACTORES						Fi	
		0: Sin Influencia	1: Incidental	2: Moderado	3: Medio	4: Significativo		5: Esencial
1	¿Requiere el sistema copias de seguridad y de recuperación fiables?						X	5
2	¿Se requiere comunicación de datos?					X		4
3	¿Existen funciones de procesos distribuidos?				X			3
4	¿Es crítico el rendimiento?			X				2
								158

5	¿Será ejecutado el sistema en un entorno operativo existente y fuertemente utilizado?	X	4
6	¿Requiere el sistema entrada de datos interactiva?	X	4
7	¿Requiere la entrada de datos interactiva que se utilicen varias pantallas o varias operaciones?	X	3
8	¿Se utilizaron los archivos maestros de forma interactiva?	X	4
9	¿Son complejas las entradas, las salidas y/o peticiones?	X	4
10	¿Es complejo el procesamiento interno?	X	5
11	¿Se ha diseñado el código para ser reutilizable?	X	3
12	¿Están incluidas en el diseño la conversión y la instalación?	X	3
13	¿Se ha diseñado el sistema para soportar diferentes instalaciones en diferentes organizaciones?	X	5
14	¿Se ha diseñado la aplicación para facilitar los cambios y para ser fácilmente utilizada por el usuario?	X	5

Factor ajuste de complejidad

54

Fuente: Elaboración propia

Una vez que se consiguió los valores correspondientes a las variables de la fórmula de los puntos función se procedió a realizar el cálculo del mismo.

$$PF = \text{Cuenta Total} * (0.65 + 0.01 * \sum F_i)$$

$$PF = 533 * (0.65 + 0.01 * 54)$$

$$PF = 533 * (1.19)$$

$$PF = 634.27$$

Para comparar los puntos función con su valor máximo, se calculó los puntos función con valores de ajuste de complejidad al máximo que es un total de 70:

$$PF = \text{Cuenta Total} * (0.65 + 0.01 * \sum F_i)$$

$$PF = 533 * (0.65 + 0.01 * 70)$$

$$PF = 533 * (1.35)$$

$$PF = 719.55$$

Después de haber calculado ambos valores se tiene que la funcionalidad real es:

$$\text{Adecuación Funcional} = \left( \frac{634.27}{719.55} \right) * 100 \%$$

$$\text{Adecuación Funciona} = 88.15 \%$$

Este resultado demuestra que la funcionabilidad del sistema web es de un 88.15 %, esto quiere decir que el sistema tiene un 88.15 % de funcionar sin riesgos a fallar con una operatividad constante y un 11.85 % aproximadamente que suceda un colapso de sistema.

#### **4.1.2. Usabilidad**

Es la capacidad del software que su comprensión sea fácil e intuitivo para el uso, satisfaciendo los requerimientos establecidos por el usuario, se realiza una evaluación en base a encuestas a 10 personas del uso del sistema, las cuales se evalúan es ponderada al 100 %, los usuarios tienen el conocimiento de los procesos y los resultados son reflejados en la siguiente tabla:

Tabla N.º 4. 4 Encuesta sobre Usabilidad del Sistema

Nro.	PREGUNTAS	RESPUESTAS		% de SI
		SI	NO	
1	¿Aprendió a usar rápido el sistema?	8	2	80
2	¿Las vistas de pantalla que vio fueron de su agrado?	9	1	90
3	¿Las pantallas que vio fueron fáciles de comprender?	10	0	100
4	¿El sistema responde rápido a sus solicitudes?	9	1	90
5	¿El sistema le facilitara el trabajo?	9	1	90
6	¿El sistema reduce su tiempo de trabajo?	10	0	100
7	¿Es fácil navegar por las distintas opciones?	10	0	100
8	¿Las operaciones que se realizan no son complicadas?	10	0	100
9	¿El sistema le proporciono las respuestas requeridas?	9	1	90
10	¿El sistema no presento errores?	9	1	90
<b>RESULTADO DE LA USABILIDAD ES DE:</b>				<b>93 %</b>

Fuente: Elaboración propio

### 4.1.3. Fiabilidad

Es la capacidad del software para asegurar un nivel de funcionamiento adecuado cuando es utilizado en condiciones específicas por cierto tiempo. Para la fiabilidad del sistema se considera, calcula la fiabilidad de cada módulo o subsistema de forma independiente.

Para calcular la fiabilidad de cada módulo se usó la fórmula:  $R(t) = e^{-\lambda t}$

Donde:

R (t): Fiabilidad de un componente o subsistema t.

$\lambda$ : Tasa de constantes de fallo ( $\lambda = N^\circ$  de fallas de acceso /  $N^\circ$  total de accesos al sistema).

t: Periodo de operación de tiempo.

$e^{-\lambda t}$  : Probabilidad de falla de un componente o subsistema en el tiempo t.

$$e = 2,72$$

Luego de realizar pruebas de cada módulo en un tiempo de 4 horas continuas se obtiene la siguiente tabla:

Tabla N.º 4. 5 Valores de fiabilidad de cada modulo

NRO.	MODULO	$\lambda$	t	R (t)
1	Módulo Usuario y Seguridad	0.012	4 Horas.	0.95
2	Módulo Plan de Cuentas	0.022	4 Horas.	0.92
3	Módulo Libro Diario	0.018	4 Horas	0.93
4	Módulo Libro Mayor	0.012	4 Horas	0.95

5	Módulo Balance general	0.012	4 Horas	0.95
6	Módulo de Productos	0.025	4 Horas.	0.90
7	Módulo de Categorías	0.022	4 Horas	0.92
8	Módulo de Proveedores	0.022	4 Horas	0.92
9	Módulo de Clientes	0.022	4 Horas	0.92
10	Módulo de Compras	0.018	4 Horas	0.93
11	Módulo de Ventas	0.018	4 Horas	0.93
12	Generar Facturas o Recibos	0.005	4 Horas.	0.99
13	Módulo de Archivos	0.005	4 Horas.	0.99

Fuente: Elaboración propia

Para calcular la fiabilidad del sistema completo, en el módulo de usuario y seguridad se vio que si una autenticado de ingreso falla no se puede acceder al resto de los módulos por lo tanto está conectado en serie según el rol que se los asigna y los demás módulos tienen función independiente. Es por eso que la fiabilidad del sistema estaría dada por la siguiente fórmula:

$$\text{Fiabilidad} = R_s * R_p$$

Donde:

$$R_s = R_1 = 0.95 \quad \text{y} \quad R_p = \frac{\sum_{i=2}^5 (R_i * R_i)}{\sum_{i=2}^5 P_i}$$

La fórmula de  $R_P$ , la variable  $P_i$  es la participación en el equipo de desarrollo del módulo y como la participación fue al 100% entonces  $P_i = 1$ , así se tiene el siguiente resultado:

$$R_P = \frac{\sum_{i=2}^5 R_i}{12} = \frac{0.92+0.93+0.95+0.95+0.90+0.92+0.92+0.92+0.93+0.93+0.99+0.99}{12} = \frac{11.25}{12} = 0.94$$

Se comprueba que la fiabilidad del sistema está dada por:

$$\text{Fiabilidad} = 0.95 * 0.94 = 0.893 * 100 = 89.3 \%$$

De lo cual se puede establecer que un 10.7 % de que el sistema presente algún fallo, que pueda ser debido a que exista fallas con la conexión del sistema a la base de datos, servidor o uso incorrecto por parte del usuario.

#### 4.1.4. Mantenibilidad

Es la cualidad que tiene el software para poder ser modificado, incluyendo correcciones o mejoras del software, en los cambios de entorno o especificaciones de requerimientos funcionales. Para calcular la calidad de mantenimiento del sistema se emplea el Índice de Madurez del Software (IMS), que indica la estabilidad de un producto de software. El índice de madurez se calcula con la siguiente formula:

$$IMS = \frac{M_t - (F_a + F_b + F_c)}{M_t}$$

Donde:

$M_t$ : Número de módulos en la versión actual.

$F_a$ : Número de módulos en la versión actual que se han cambiado.

$F_b$ : Número de módulos en la versión actual que se han añadido.

$F_c$ : Número de módulos en la versión anterior que se han borrado en la versión actual.

Toda la recopilación de la información requerida por la fórmula se obtuvo en la siguiente tabla:

Tabla N.º 4. 6 Información requerida por el IMS

INFORMACIÓN	VALOR
$M_t$	13
$F_a$	0
$F_b$	0
$F_c$	0

Fuente: Elaboración propia

Se realiza el cálculo del IMS con los valores obtenidos:

$$IMS = \frac{13 - (0 + 0 + 0)}{13} = \frac{13}{13} = 1$$

$$IMS = 1 * 100 \% = 100 \%$$

Con ese resultado se concluye que la madurez del “Sistema Web de Gestión Administrativa y Seguimiento Contable” Caso: Empresa Construcción Innovación & Tecnología “C.I.T.”, la cual nos indica que tiene un índice de madurez de software del 100 %.

#### 4.1.5. Portabilidad

Es la capacidad del software para ser trasladado de un entorno de operación a otro. Para medir la portabilidad del sistema se empleará la siguiente fórmula que indica el grado de portabilidad del software.

$$GP = 1 - \frac{ET}{ER}$$

Donde:

ET: Es la medida de los recursos necesarios para llevar el sistema a otro entorno.

ER: es la medida de los recursos necesarios para crear el sistema en el entorno residente.

Si  $GP > 0$ : La portabilidad es más rentable que el re-desarrollo.

Si  $GP = 1$ : La portabilidad es perfecta.

Si  $GP < 0$ : El re-desarrollo es más rentable que la portabilidad.

Para llevar el sistema web a otro entorno se necesita una memoria USB 1 GB mínimo de capacidad, para crear el sistema en el entorno residente se necesita contar con un equipo de sistema operativo que pueda ser (Windows, Linux o MacOS) en las cuales se requiere tener instalado servidor Xampp, Composer, editor de código, lenguaje de programación PHP, gestor de base de datos MySQL instalado y navegador web Google Chrome, Mozilla Firefox, Safari, Opera y Brave.

Con esta información requiere por la formula, se procede a calcular el grado de portabilidad.

$$GP = 1 - \frac{1}{13} = 1 - 0.07 = 0.93 * 100 = 93 \%$$

Se concluye que el sistema web tiene un grado de portabilidad de un 93%.

#### **4.1.6. Calidad General**

Una vez calculado todos los atributos con sus respectivos porcentajes según el estándar de la ISO/IEC 25010, se realiza un resultado general de la calidad del software como se visualiza en la siguiente tabla:

Tabla N.º 4. 7 Información de Calidad Global

<b>ATRIBUTOS</b>	<b>VALOR EN %</b>
Adecuación funcional	88.15
Usabilidad	93
Fiabilidad	89.3
Mantenibilidad	100
Portabilidad	93
Calidad General	92.69

Fuente: Elaboración propia

La calidad general del “Sistema Web de Gestión Administrativa y Seguimiento Contable” Caso: Empresa Construcción Innovación & Tecnología “C.I.T.” es del 92.69 %.

## **4.2. Seguridad**

La seguridad lógica fue muy importante durante el desarrollo del sistema, se implementó la seguridad lógica en el sistema web, como ser la autenticación de usuario, asignación de roles y acciones realizadas por el usuario, las que pertenecen al módulo de usuario, para proteger la información que se traslada entre el servidor se emplea la encriptación, también se implementó validación de datos del Framework Laravel y finalmente la seguridad de la base de datos.

### **4.2.1. Autenticación**

El acceso al sistema es controlado por la autenticación, en la cual un usuario debe introducir los datos correctos como ser: nombre de Usuario, Contraseña y la transferencia de datos encriptadas, estos datos son validados en el lado del cliente y servidor.

En la siguiente figura se puede observar un fragmento de código de validación:

Figura N.º 4. 1 Código de validación de autenticación

```
class LoginController extends Controller
{
    public function showLoginForm(){
        return view( view: 'auth.login');
    }

    public function login(Request $request){

        $this->validateLogin($request);

        if (Auth::attempt(['usuario' => $request->usuario, 'password' => $request->password, 'condicion'=>1])){
            return redirect( to: '/home');
        }

        return back()->withErrors(['usuario' => trans( key: 'auth.failed')])
        ->withInput(request(['usuario']));
    }

    protected function validateLogin(Request $request){
        $this->validate($request,[
            'usuario' => 'required|string',
            'password' => 'required|string'
        ]);
    }

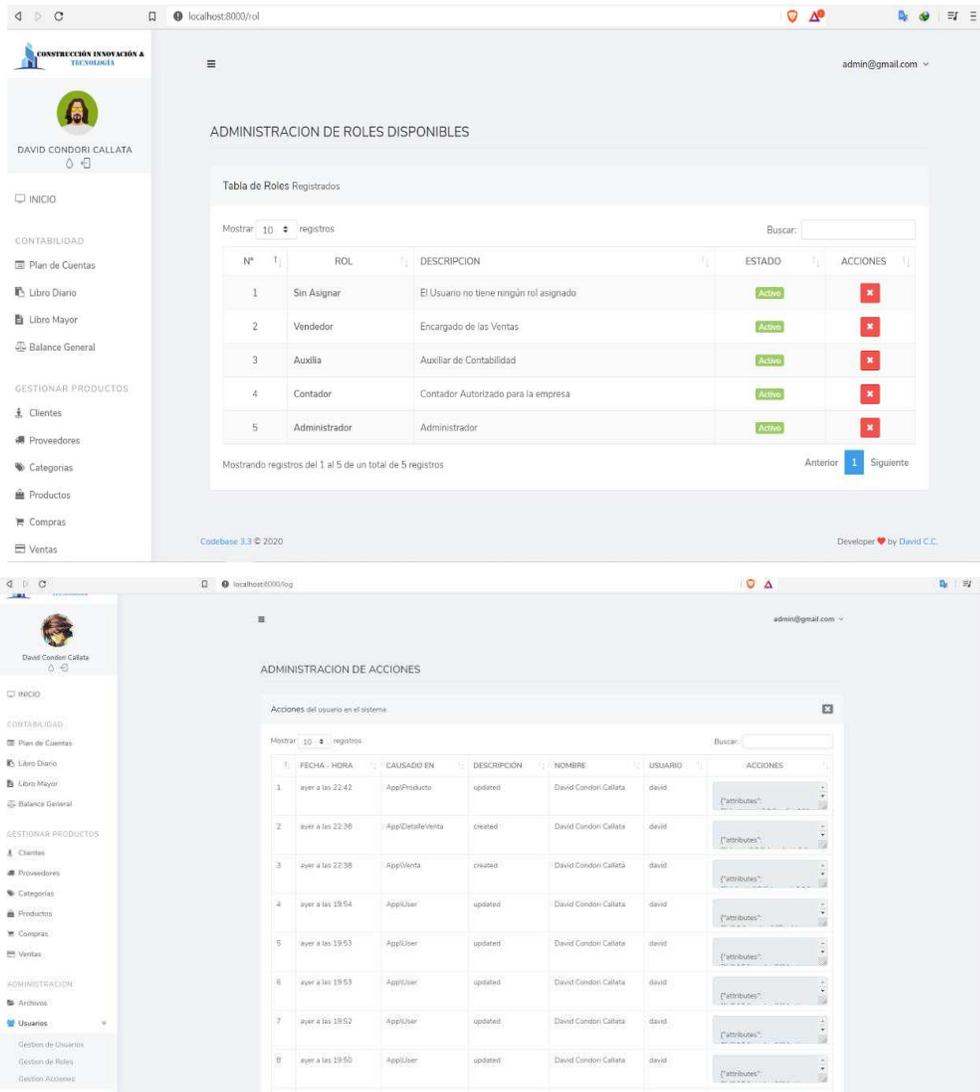
    public function logout(Request $request){
        Auth::logout();
        $request->session()->invalidate();
        return redirect( to: '/');
    }
}
```

Fuente: Elaboración propia

## 4.2.2. Asignación de roles y acciones

La asignación de roles permite trabajar con un grupo de usuarios y que este permita accesos y visualización de elementos independientes, permitiendo que cada grupo visualiza su información asignada, siendo invisible el contenido del otro grupo. Los roles permiten un nivel de permisos con una serie de privilegios. Esta es una medida de seguridad que se implementa para proteger que usuarios no autorizados tengan acceso a ciertos módulos del sistema y se registra todas las acciones realizadas por cada usuario.

Figura N.º 4. 2 Roles de asignación y Acciones



Fuente: Elaboración propia

### 4.2.3. Encriptación

La encriptación de datos es muy importante para proteger los paquetes de información que se transmiten entre servidores, en el presente sistema se utilizó algoritmos de encriptación que ofrece Laravel OpenSSL y cifrado AES -256-CBC y AES-128-CBC que los valores encriptados están firmados por un código de autenticación de mensajes que detecta si el mensaje encriptado fue alterado.

Figura N.º 4. 3 Encriptación Laravel

```
namespace App\Http\Controllers;

use App\Http\Controllers\Controller;
use App\User;
use Illuminate\Http\Request;

class UserController extends Controller
{
    /**
     * Store a secret message for the user.
     *
     * @param Request $request
     * @param int $id
     * @return Response
     */
    public function storeSecret(Request $request, $id)
    {
        $user = User::findOrFail($id);

        $user->fill([
            'secret' => encrypt($request->secret),
        ]->save());
    }
}
```

Fuente: (Laravel.com, 2020)

#### 4.2.4. Seguridad en la base de datos

El gestor base de datos MySQL, tiene sus propias medidas de seguridad como una básica autenticación. Uno de los ataques más frecuentes realizados a una base de datos son las inyecciones SQL.

Las inyecciones SQL son cadena de instrucciones SQL que un usuario puede introducir en cualquier campo de un formulario. El framework Laravel proporciona la protección contra ataques de inyección de CSRF, XSS y SQL todas las consultas se realizan pasando por los modelos y siendo enviados a los controladores para validar los datos y envían al servidor, las consultas se realizan como se observa en la siguiente figura:

Figura N.º 4. 4 Tipo de Consultas evitando las inyecciones SQL

```
public function create()
{
    $cuentas = DB::table('cuentas as cuent')
        ->select(DB::raw("CONCAT(cuent.cuenta, ' - ', cuent.tipo_cuenta) AS cuenta_tipo"), 'cuente.id')
        ->where('cuente.condicion', '=', '1')
        ->orderBy('cuente.cuenta')
        ->get();

    $contar = DB::table('asientos')
        ->select(DB::raw('count(*)+1 as numero'))
        ->get();
    $cont = 0;
    return view('asiento.create', ["cuentas" => $cuentas, "contar" => $contar, 'cont' => $cont]);
}
```

Fuente: Elaboración propia

### 4.3. Estimación de costos del software

#### 4.3.1. Cocomo II

Para el cálculo del costo total con COCOMO II se emplea el uso de la herramienta “COCOMO II – Constructive Cost Model” que pertenece al Centro de Sistemas e Ingeniería de Software de la Universidad de California del Sur.

En el presente proyecto se realiza la estimación de costo total por LDC (Líneas de Código), que se realiza con los datos obtenidos del software la cantidad de líneas de código que se empleó en el desarrollo. Además de la información de líneas de código el modelo requiere varias características como la flexibilidad del desarrollo, la arquitectura, el trabajo de equipo, tamaño de la base de datos, complejidad del producto, capacidad del personal, la limitación de tiempo, la restricción de almacenamiento, uso de herramientas de software entre otros.

Tabla N.º 4. 8 Líneas de código total

CODIGO	TOTAL, LDC
Código utilizado	2777
Códigos reutilizados	716
Código modificado	659

Fuente: Elaboración propia

Una vez calculado todas las Líneas de código que se emplearon en el presente proyecto se realiza la evaluación de estimación de costo, tomando en cuenta el salario mínimo que es de 2060.00 Bs la cual nos da 296.00 Dólares, en la siguiente Figura se observa el llenado de datos:

Figura N.º 4. 5 Inserción de datos para COCOMO II

COCOMO II - Modelo de costo constructivo

Monte Carlo Risk  
 Apagado ▼  
 Calcular automáticamente  
 Apagado ▼

Método de dimensionamiento del tamaño del software Líneas de código fuente ▼

	<a href="#">SLOC</a>	% De diseño modificado	% Código modificado	% De integración requerida	Evaluación y asimilación (0% - 8%)	Comprensión del software (0% - 50%)	Desconocimiento (0-1)
Nuevo	2777						
Reutilizado	716	0	0	80	6		
Modificado	659	23	8	83	7	47	1

**Controladores de escala de software**

Precedentedness	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Arquitectura / Resolución de riesgos	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>
Flexibilidad de desarrollo	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Cohesión del equipo	<span style="border: 1px solid #ccc; padding: 2px;">Alto ▼</span>

**Controladores de costos de software**

<b>Producto</b>		<b>Personal</b>		<b>Plataforma</b>	
Fiabilidad de software requerida	<span style="border: 1px solid #ccc; padding: 2px;">Alto ▼</span>	Capacidad de analista	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Limitación de tiempo	<span style="border: 1px solid #ccc; padding: 2px;">Alto ▼</span>
Tamaño de base de datos	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Capacidad de programador	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Restricción de almacenamiento	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>
Complejidad del producto	<span style="border: 1px solid #ccc; padding: 2px;">Bajo ▼</span>	Continuidad de personal	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Volatilidad de plataforma	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>
Desarrollado para la reutilización	<span style="border: 1px solid #ccc; padding: 2px;">Alto ▼</span>	Experiencia de aplicación	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>		
Documentación que coincida con las necesidades del ciclo de vida	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	Experiencia de plataforma	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>		
		Experiencia de lenguaje y conjunto de herramientas	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>	<b>Proyecto</b>	
				Uso de herramientas de software	<span style="border: 1px solid #ccc; padding: 2px;">Alto ▼</span>
				Desarrollo multistio	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>
				Programa de desarrollo requerido	<span style="border: 1px solid #ccc; padding: 2px;">Nominal ▼</span>

Mantenimiento Apagado ▼

Costo de las tarifas laborales del software por persona-mes (dólares) 296

Calcular

Fuente: (softwarecost.org/tools/COCOMO/,2020)

Después de llenar la información requerida herramienta nos proporciona los resultados lo cual se visualiza en la siguiente figura:

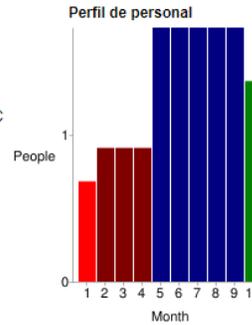
Figura # Resultado de estación de costo

**Resultados**

Esfuerzo de desarrollo de software (elaboración y construcción) = 11.8  
 Horario de persona-mes = 8.3  
 Costo de meses = \$ 3486

Tamaño equivalente total = 3504 Distribución de la fase de adquisición de SLOC

Fase	Esfuerzo (Persona-meses)	Horario (meses)	Personal promedio	Costo (dólares)
Comienzo	0.7	1.0	0.7	\$209
Elaboración	2.8	3.1	0.9	\$837
Construcción	9.0	5.2	1.7	\$2650
Transición	1.4	1.0	1.4	\$418



**Distribución de esfuerzo de software para RUP / MBASE (persona-meses)**

Fase / Actividad	Comienzo	Elaboración	Construcción	Transición
administración	0.1	0.3	0.9	0.2
Medio ambiente / CM	0.1	0.2	0.4	0.1
Requisitos	0.3	0.5	0.7	0.1
Diseño	0.1	1.0	1.4	0.1
Implementación	0.1	0.4	3.0	0.3
Evaluación	0.1	0.3	2.1	0.3
Despliegue	0.0	0.1	0.3	0.4

Su archivo de salida es [http://softwarecost.org/tools/COCOMO/data/COCOMO\\_June\\_12\\_2020\\_02\\_43\\_03\\_157087.txt](http://softwarecost.org/tools/COCOMO/data/COCOMO_June_12_2020_02_43_03_157087.txt) Para obtener más información, comuníquese con Ray Madachy en ray@softwarecost.org

Fuente: (softwarecost.org/tools/COCOMO/,2020)

Los resultados que nos proporciona la herramienta son datos importantes que se observan en la siguiente tabla:

Tabla N.º 4. 9 Resultados de COCOMO II

RESULTADOS	DETALLE
Esfuerzo de desarrollo de software (Elaboración y construcción).	11.8 Personas/Mes
Tiempo estimado de desarrollo.	8.3 Meses
Cantidad de personas para desarrollar el proyecto.	1 Persona
Costo total del proyecto final.	3.486,00 Dólares Total Bs. 24.297,42

Fuente: Elaboración propia

## **5. CONCLUSIONES Y RECOMENDACIONES**

## 5.1. Conclusiones

Se concluye con el desarrollo e implementación del Sistema Web de Gestión administrativa y Seguimiento Contable para la empresa Construcción Innovación & Tecnología “C.I.T.” la cual permite un seguimiento al flujo de caja para una mejor toma de decisiones y la administración de ventas, facturación electrónica, compras de productos y respaldo de documentos las cuales permiten a la institución así coadyuvando en una mejor información de sus actividades.

Se desarrolló todos los módulos según los requerimientos de la empresa después de realizar un análisis de toda la información proporcionada y las entrevistas que se realizaron para determinar los módulos que compondrá el sistema, los cuáles se mencionan a continuación:

- **Módulo Usuarios:** La cual nos permite administrar todos los usuarios existentes y una seguridad de asignación de roles y acciones realizadas dentro del sistema.
  - **Roles:** Agrupar a los usuarios limitando que módulos pueden administrar
  - **Acciones:** Tener un seguimiento de todo lo que realiza un usuario dentro del sistema.
- **Módulo Plan de cuentas:** Permitted registrar las diferentes cuentas que existen para el registro del libro diario.
- **Módulo Libro diario:** Es el libro contable donde se recogen, día a día los hechos económicos de la empresa.
- **Módulo Libro mayor:** Permitted realizar consultas los saldos actuales por cada cuenta existente que se registró.
- **Módulo Balance general:** Permitted realizar la consulta general de todas las cuentas entre un periodo de tiempo.

- **Módulo de Productos:** Permite un adecuado control del registro de los productos que se adquieren y están disponibles.
- **Módulo de Categorías:** Permitiendo que agrupar los diferentes productos existentes.
- **Módulo de Proveedor:** Registrar a los proveedores de los cuales se adquiere los productos.
- **Módulo de Clientes:** Permite un registro adecuado de todos los clientes a los cuales se realiza una determinada venta.
- **Módulo de Compras:** Permite administrar todas las compras que se realizan con todos los detalles necesarios así coadyuvar en la toma de decisiones de que productos se adquieren.
- **Módulo de Ventas:** Permitiendo administras las ventas realizadas de los productos que ofrecen y así tener un control de la información cotización de los montos de ventas que se realiza a los determinados clientes.
- **Generar Facturas o Recibos:** Permite genera las facturas de las ventas realizadas a terminados clientes con la factura electrónica y también generar recibos o facturas de las compras que la empresa realiza.
- **Módulo de Archivos:** Permite la creación de archivos donde se almacenan diferentes documentos de forma digital como las cuales pueden ser contratos, facturas, imágenes entre otros.

De esta manera se concluyó que el objetivo general como los específicos que se cumplió satisfactoriamente en el desarrollo del sistema, Así ayudando a la empresa tener una mejor gestión de la información contable y facilitando las tomas de decisiones.

## **5.2. Recomendaciones**

En el presente desarrollo del sistema se llevaron a cabo las diferentes pruebas de unitarias, pruebas de integración y pasando por las pruebas de caja blanca, caja negra, pero esto no significa que el sistema no tenga errores, por esta razón es recomendable que cuente con un personal que haga el mantenimiento al sistema para atender errores a largo plazo.

También se recomienda que se pueda implementar los pagos online para así tener mejores ventas y llegar a mucho más lugar que esto aumentaría los ingresos de la empresa.

Ultima recomendación, que se pueda implementar la automatización para pueda calcular directamente los formularios 200 IVA y el formulario 400 IT.

# BIBLIOGRAFÍA

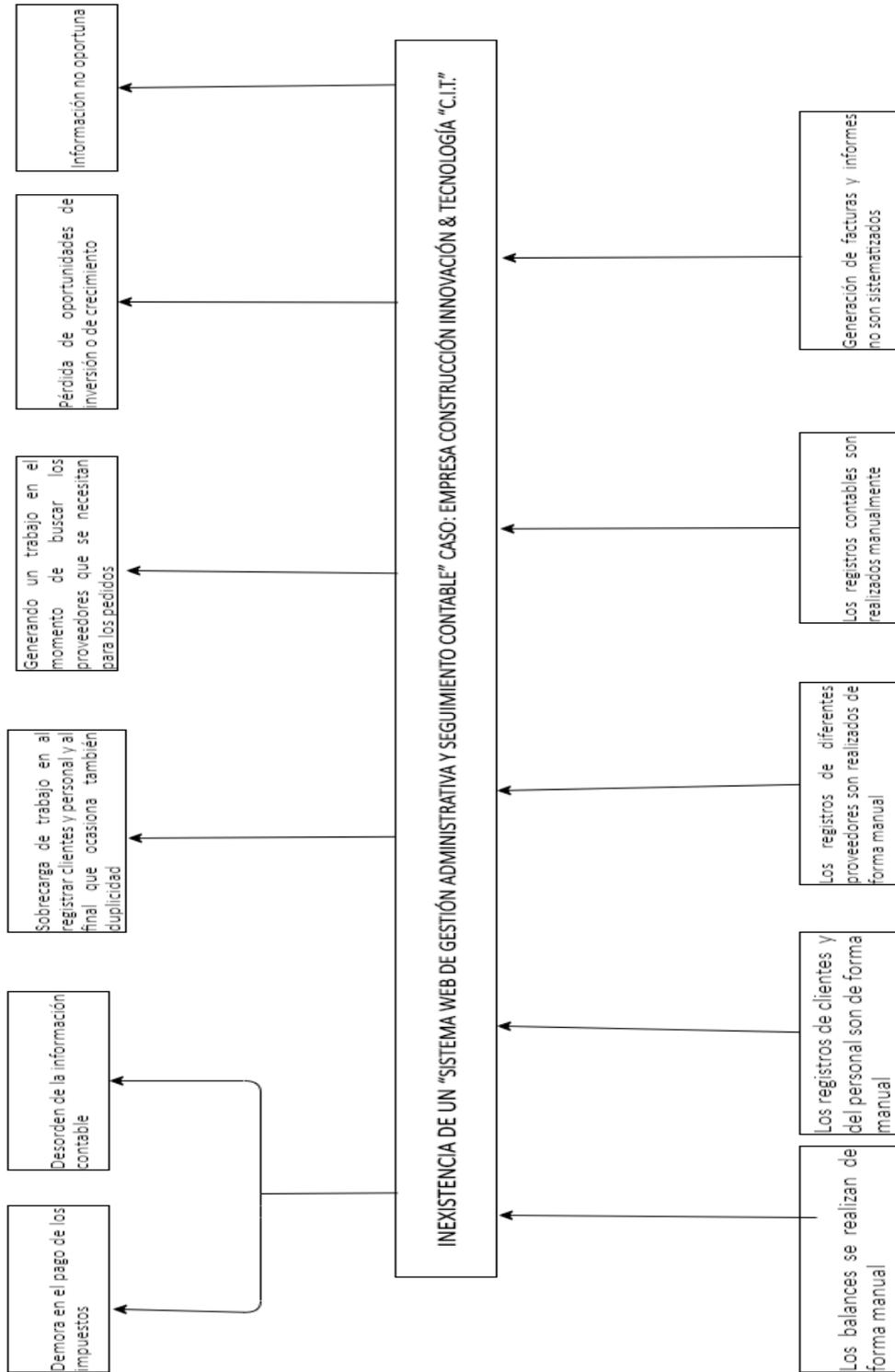
- Limited, S. (11 de 10 de 2014). *Qué es Composer y cómo usarlo*. Obtenido de Composer es un manejador de dependencias: <https://styde.net/que-es-composer-y-como-usarlo/>
- Beati, H. (2011). *PHP 7*.
- Beatriz Coronado, M. (s.f.). *Metodologías para el desarrollo de aplicaciones Web*. Obtenido de UML-Based Web: <https://programacionwebisc.wordpress.com/2-5-metodologias-para-el-desarrollo-de-aplicaciones-web/>
- Bootstrap. (2020). *Build fast, responsive sites with Bootstrap*. Obtenido de [getbootstrap.com](http://getbootstrap.com)
- Calero. (2005). *Introducción a la Calidad: Modelos de Calidad ISO 9126*. Obtenido de <http://alarcos.inf-cr.uclm.es/doc/cmsi/trans/S1.pdf>
- Consulting, T. (2016). *Tibel Consulting*. Obtenido de <http://tibel.com/sistema-de-gestion-administrativa>
- desarrolloweb.com. (2019). *Laravel*. Obtenido de [desarrolloweb.com](http://desarrolloweb.com)
- Diaz florian, J. M. (s.f.). *Modelo de Ciclo de Vida*. Obtenido de SystemColegas: <https://sites.google.com/site/systemcolegas/modelo-de-ciclo-de-vida>
- Fastly. (2020). *Apache friend*. Obtenido de <https://www.apachefriends.org/es/index.html>
- Galiano, L. (03 de 11 de 2012). *Planificación De Mi Proyecto II*. Obtenido de Definición de Metodología UWE.: <http://elproyectodeluisgaliano.blogspot.com/2012/11/metodologia-uwe-aplicada-mi-solucion.html>
- Gómez, A. (s.f.). *UN MODELO DE ESTIMACION DE PROYECTOS DE SOFTWARE*. Obtenido de COCOMO II: <https://blogadmi1.files.wordpress.com/2010/11/cocomoIIfull.pdf>
- Guzmán, Castañeda, Pérez, & pedroza. (2014). *Scrumban*.
- Hernanz, D. (s.f.). *Metodologías ágiles, KANBAN*. Obtenido de Conectart: <https://blog.conectart.com/metodologias-agiles/>
- Ibarra Guzman, D., Castañeda Islas, U., Perez Corona, C., & Pedroza Mendez, B. E. (2014). *Metodología ágil scrumban en el proceso de desarrollo y. Research in Computing Science*.

- its, O. C. (2019). *MySQL*. Obtenido de <https://www.mysql.com/>
- Jimenez, A. (s.f.). *Técnicas de recolección de datos*. Obtenido de Documentacion,Entrevista: <https://www.mindmeister.com/es/352485783/t-cnicas-de-recolecci-n-de-datos?fullscreen=1>
- Laravel. (2020). *Laravel Documentacion*. Obtenido de <https://laravel.com/docs/5.8>
- Lopez, J. (8 de diciembre de 2010). *INGENIERÍA WEB*. Obtenido de <https://upolijenny.blogspot.com/2010/12/ingenieria-web.html>
- Madachy, B. C. (2020). *COCOMO II - Constructive Cost Model*. Obtenido de Software Cost Estimation Metrics Manual for Defense Systems: <http://softwarecost.org/tools/COCOMO/>
- Merino, J. P. (2016). *Definición de sistema contable*. Obtenido de <https://definicion.de/sistema-contable/>
- Nils Adermann, J. B. (2019). *Composer*. Obtenido de <https://getcomposer.org/>
- Parra, C. (20 de 09 de 2015). *Metodología de desarrollo iterativo y creciente*. Obtenido de MODELADO Y GESTIÓN DE LA INFORMACION: <http://modelado-de-la-informacion.blogspot.com/2015/09/metodologia-de-desarrollo-iterativo-y.html>
- PRESSMAN, & R. P., R. (2010). *INGENIERIA DE SOFTWARE*. Mexico: Mexico Mc Graw Hill.
- Rodríguez, J. L. (s.f.). *Fases del modelo de desarrollo Scrumban*.
- Rodríguez, J. L. (s.f.). *Fases del modelo Scrumban*.
- Rumbaugh, J., Jacobson, I., & Booch, G. (2000). *EL LENGUAJE UNIFICADO DE MODELADO*. Madrid: PEARSON EDUCACION.
- Somerville . (2011). *SOFTWARE ENGINEERING*. Boston: Pearson Education.
- STANDAR, I. 2. (2019). *The ISO/IEC 25000 series of standards*. Obtenido de <https://iso25000.com/index.php/en/iso-25000-standards>
- Yarif, J. (24 de 05 de 2010). *ISO/IEC 9126 – Evaluación del producto Software*. Obtenido de Estándares del Software: <https://estandarsw.wordpress.com/category/iso/iso-9126/>

**ANEXOS**

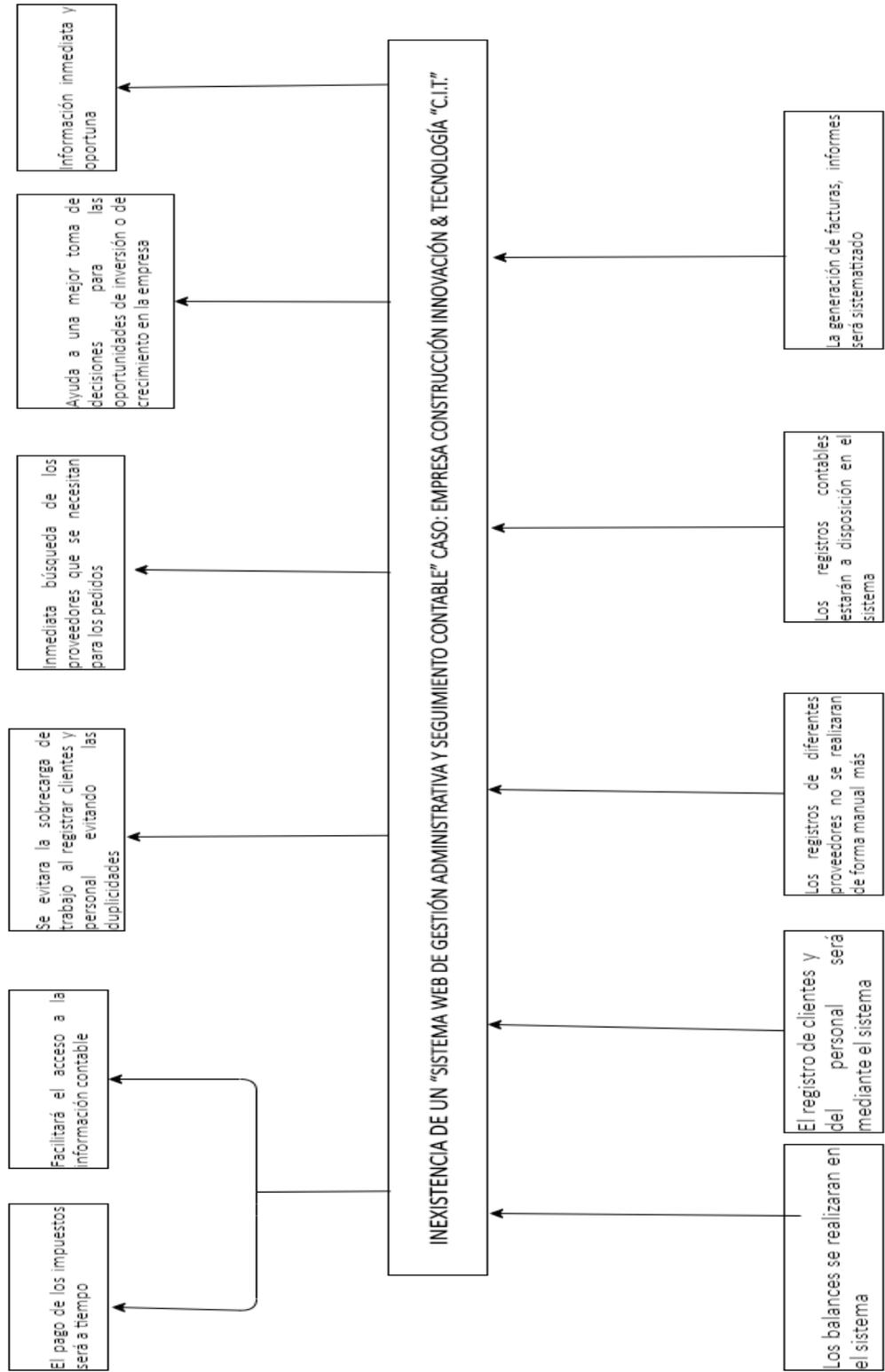
# Árbol de problemas

## ÁRBOL DE PROBLEMAS



# Árbol de objetivos

## ÁRBOL DE OBJETIVOS



## Herramientas de recolección de requerimientos

Cualitativo - Entrevista individual:

**ENTREVISTA PARA LA OBTENCIÓN DE REQUISITOS**

Entrevista para obtención de lista de requisitos, en entrevista personal con gerente general empresa Construcción Innovación & Tecnología, con preguntas abiertas.

1.- ¿Qué tipo de datos son registradas en la parte contable?

Se registran los importes, el tipo de cuenta, la fecha, la descripción.

2.- ¿Estos datos contables que se requiere registrar en cuantos grupos se divide o de que forma es establecido?

Son el plan de cuentas, los balances, el libro diario, el libro mayor

3.- ¿Qué es lo que se necesita para la toma de decisiones para impuestos nacionales?

Se necesita los registros del libro diario, donde se contemplan los registros de las actividades realizadas dentro de la empresa.

4.- ¿Qué tipo de documentos es necesario realizar un respaldo en formatos de PDF o imágenes entre otros?

Son: El contrato de obras y licitaciones, factura de pago de clientes.

5.- ¿Los registros de qué forma son realizadas?

Son realizadas manualmente, las mismas se registran en hojas simples.

6.- ¿Qué tipo de reportes son requeridas?

El libro diario, libro mayor y el balance.

7.- ¿Realiza ventas de diversos productos o solo una categoría de productos?

Se realizan la venta de diversos productos.

8.- ¿Tiene proveedores de los cuales compra los productos?

Se cuenta con diversos proveedores.

9.- ¿De qué forma de publicidad hace conocer sus productos para la venta?

Mediante volantes y folletos.

10.- ¿De qué forma emite las facturas o comprobantes de compras?

Se realizan de manera manual.

11.- ¿Los pagos realizados son al contado o a cuotas?

Los pagos son todo al contado.

12.- ¿Tiene alguna forma de ver los detalles de las compras?

Se cuenta con un registro de manera manual

13.- ¿Qué funciones cumple cada empleado?

- Contador, encargado de los registros contables.
- Auxiliar contable, encargado de ayudar al contador.
- Vendedor, encargado de las ventas.

14.- ¿Lleva alguna forma de respaldo de las ventas realizadas?

No se cuenta con un respaldo detallado para cada cliente de las ventas realizadas.

15.- ¿Lleva acabo alguna forma de registro de proveedores y de qué forma se registra?

Solo se tiene los números de contacto del proveedor, no se cuenta con un registro detallado.

16.- ¿Lleva acabo un registro de todos los clientes?

No se lleva ningún tipo de registro detallado de cada cliente.

## Aval de conformidad de tutor especialista

La Paz, junio 2020

Señor(a)

**Ing. Maricel Yarari Mamani**

**TUTOR METODOLOGICO TALLER DE LICENCIATURA II**

**CARRERA DE INGENIERÍA DE SISTEMAS - UPEA**

Presente.

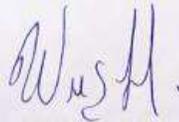
### REF. AVAL DE CONFORMIDAD

Distinguida Ingeniera,

Mediante la presente tengo a bien comunicarle mi conformidad con el Proyecto de Grado denominado **"SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO CONTABLE" CASO: EMPRESA CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA "C.I.T."**, que propone el postulante **Univ. David Condori Callata** con cedula de identidad **N.º 8279910** expedido en la ciudad de La Paz, para su defensa Pública, evaluación correspondiente a la materia Taller de Licenciatura II, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba mis saludos cordiales.

Atentamente,



.....  
Lic. Wendy Sarmiento Martínez

C.I.: 6146842 LP

**TUTOR ESPECIALISTA**

## Aval de conformidad de tutor revisor

La Paz, junio de 2020

Señor(a)

**Ing. Maricel Yarari Mamani**

**TUTOR METODOLOGICO TALLER DE LICENCIATURA II**

**CARRERA DE INGENIERÍA DE SISTEMAS - UPEA**

Presente.

**REF. AVAL DE CONFORMIDAD**

Distinguida Ingeniera,

Mediante la presente tengo a bien comunicarle mi conformidad con el Proyecto de Grado denominado **"SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO CONTABLE" CASO: EMPRESA CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA "C.I.T."**, que propone el postulante **Univ. David Condori Callata** con cedula de identidad **N.º 8279910** expedido en la ciudad de La Paz, para su defensa Pública, evaluación correspondiente a la materia Taller de Licenciatura II, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba mis saludos cordiales.

Atentamente,



.....  
**Lic. Beatriz Colque Condori**

**C.I.: 3431870 LP**

**TUTOR REVISOR**

## Aval de conformidad de tutor metodológico

La Paz, julio 2020

A: Ing. David Carlos Mamani Quispe  
DIRECTOR DE LA CARRERA DE INGENIERÍA DE SISTEMAS  
A: Honorable Consejo de Carrera  
INGENIERÍA DE SISTEMAS U.P.E.A.

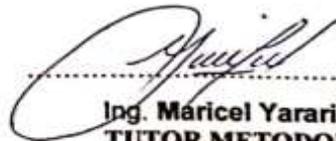
Presente. -

### REF. AVAL DE CONFORMIDAD

Mediante la presente tengo a bien comunicarle mi conformidad con el proyecto de Grado denominado "**SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO CONTABLE**" CASO: **EMPRESA CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA "C.I.T."**, que propone el postulante Univ. **David Condori Callata** con cedula de identidad N.º **8279910** expedido en la ciudad de La Paz, para su defensa pública, evaluación correspondiente a la materia Taller de Licenciatura II, de acuerdo a reglamento vigente de la carrera de Ingeniería de Sistemas de la Universidad Pública de El Alto.

Sin otro particular, reciba mis saludos cordiales.

Atentamente,

  
.....  
Ing. Márcel Yarari Mamani  
TUTOR METODOLÓGICO

## Aval de aceptación de la empresa.



### CONSTRUCCIÓN INNOVACIÓN Y TECNOLOGÍA

La Paz, 12 de diciembre de 2018

Señores:

Ing. Víctor Mamani Mamani

**DIRECTOR DE CARRERA INGENIERÍA DE SISTEMAS**

Presente. -

#### REF.: ACEPTACIÓN DE SOLICITUD

En primera instancia, hacer llegar un cordial saludo a su autoridad en la dirección de carrera Ingeniería de Sistemas y al correspondiente docente de la asignatura de Taller de Licenciatura I Ing. Dionicio Henry Pacheco Ríos.

El motivo de la presente, es para informarle la autorización de proporcionar la información de nuestra institución para cubrir los requerimientos y necesidades para el desarrollo del software y realizar el trabajo de investigación para el perfil de proyecto de grado la cual es denominado **"SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO CONTABLE" CASO: EMPRESA CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA "C.I.T."**. La cual está siendo elaborado por el Universitario David Condori Callata.

Además pedir que los resultados obtenidos sobre el presente perfil de proyecto de grado se nos puedan proporcionar, que será muy beneficioso para la institución que esto nos permitirá una mejor toma de decisiones de manera oportuna.

Sin otro particular me despido de usted con las consideraciones más distinguidas.

Atentamente:

Lic. Marycel Julieta Quisbert Torres  
GERENTE GENERAL EMPRESA CONSTRUCTORA:  
**CONSTRUCCIÓN INNOVACIÓN Y TECNOLOGÍA**



Av. San José Nro. 2217, esquina Av. Florida Zona Munaypata  
Telf.: (591-2)2396235 – 78910189. E-mail: construccioninnovaciontec@gmail.com

## Certificado de aval de culminación de la empresa



CONSTRUCCIÓN INNOVACIÓN Y TECNOLOGÍA

# CERTIFICADO

A quien corresponda:

Gerente general Lic. Marycel Julieta Quisbert Torrez con C.I. 4331911 LP de la empresa CONSTRUCCIÓN INNOVACIÓN Y TECNOLOGÍA, certifica que el Sr. David Condori Callata con C.I. 8279910 LP, estudiante de la carrera de Ingeniería de Sistemas de la UNIVERSIDAD PÚBLICA DE EL ALTO, ha realizado el desarrollo eh implementación del SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO CONTABLE como proyecto de grado en la gestión I/2020.

Asimismo, se informa que el Sr. de referencia demostró los siguientes aspectos puntuales en el desempeño del trabajo realizado.

- Disciplina y puntualidad.
- Presentación formal.
- Responsabilidad y esmero.
- Iniciativa propia y creatividad.
- Relaciones humanas y ética.

En cuanto certifico para fines consiguientes del interesado, en el marco del honor a la verdad y para fines legales.

La paz, 26 de junio de 2020



Av. San José nro. 2217, esquina Av. Florida Zona Munaypata  
Telf.: (591-2)2396235 – 78910189. E-mail: construccioninnovaiontec@gmail.com

“SISTEMA WEB DE GESTIÓN ADMINISTRATIVA Y SEGUIMIENTO  
CONTABLE”

CASO: EMPRESA CONSTRUCCIÓN  
INNOVACIÓN & TECNOLOGÍA “C.I.T.”



MANUAL DE USUARIO



**CONSTRUCCIÓN INNOVACIÓN &  
TECNOLOGÍA**

## **Introducción**

Es esencial revisar antes el Manual de Usuario la cual será la guía paso a paso en el manejo del sistema, para un mejor entendimiento.

El presente sistema se encarga de realizar un seguimiento de la gestión administrativa y seguimiento contable de la Empresa Construcción Innovación & Tecnología “C.I.T.”.

El Sistema Web de Gestión Administrativa y Seguimiento Contable de la Empresa Construcción Innovación & Tecnología “C.I.T.” fue creado por el Universitario David Condori Callata de la Universidad Pública de El Alto del 10mo semestre de la asignatura Taller II de la carrera Ingeniería de Sistemas.

## **Objetivos Del Manual**

Guiar al Usuario encargado sobre el manejo del Sistema Web de Gestión Administrativa y Seguimiento Contable “C.I.T.” obteniendo resultado eficiente.

- Dar a entender a los usuarios encargados las características y las formas del funcionamiento del sistema brindado
- Diagramación que corresponda a su necesidad
- Proporcionar asistencia al usuario encargado del sistema

## **Conocimientos Básicos del Usuario**

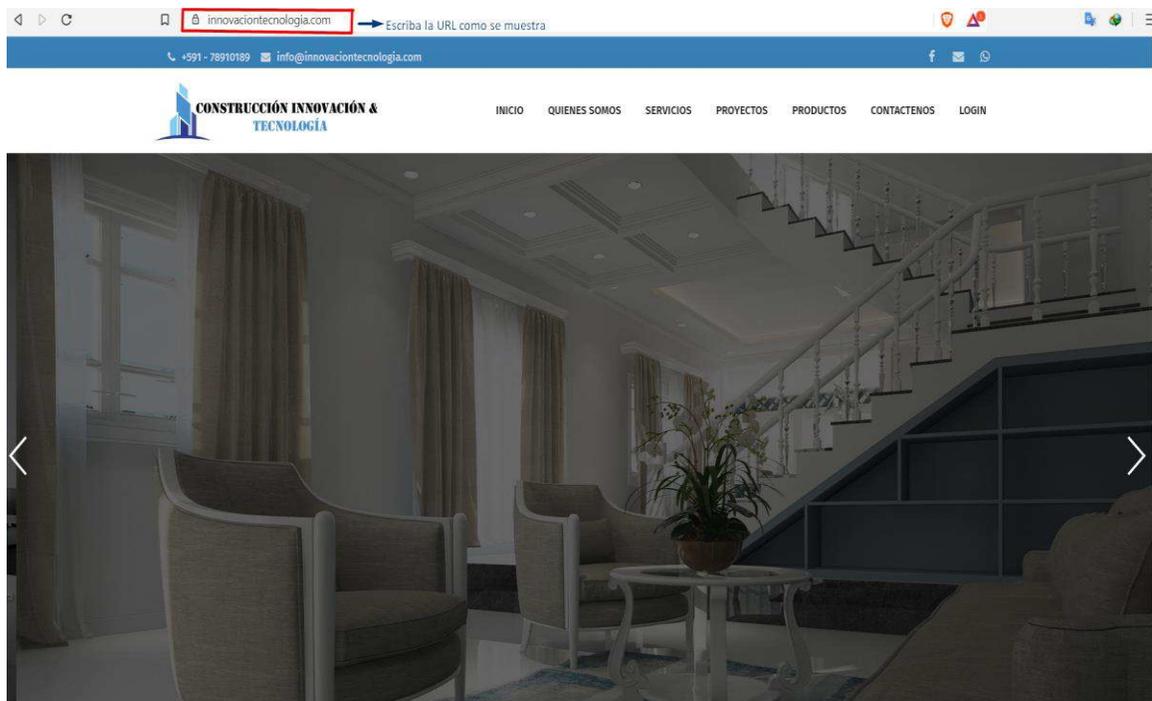
- Conocimiento básico en la Navegación Web
- Conocimiento básico en Manejo de Facebook
- Conocimiento básico en Correos Electrónicos
- Conocimientos básicos en WhatsApp

## **Requisitos Previos**

- Equipo de Computación, con una capacidad de RAM 2GB o superior, Disco duro Mínimo de 50 GB, Procesador Core i3 o Superior
- Sistema Operativo Windows, Linux o MacOS.
- Estar conectado a internet.
- Tener instalado un navegador ya sea Firefox, Chrome, entre otros navegadores.
- Tabletas, celular con conexión a internet y un navegador instalado.

## Interfaces de navegación para el Usuario

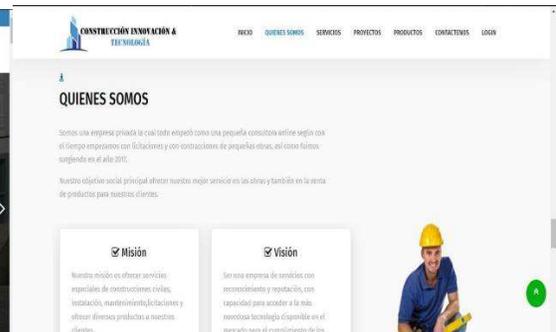
Abrir el Navegador y anotar la siguiente URL <https://innovaciontecnologia.com>



Ingresado a la URL, aparecerá la Página Principal.

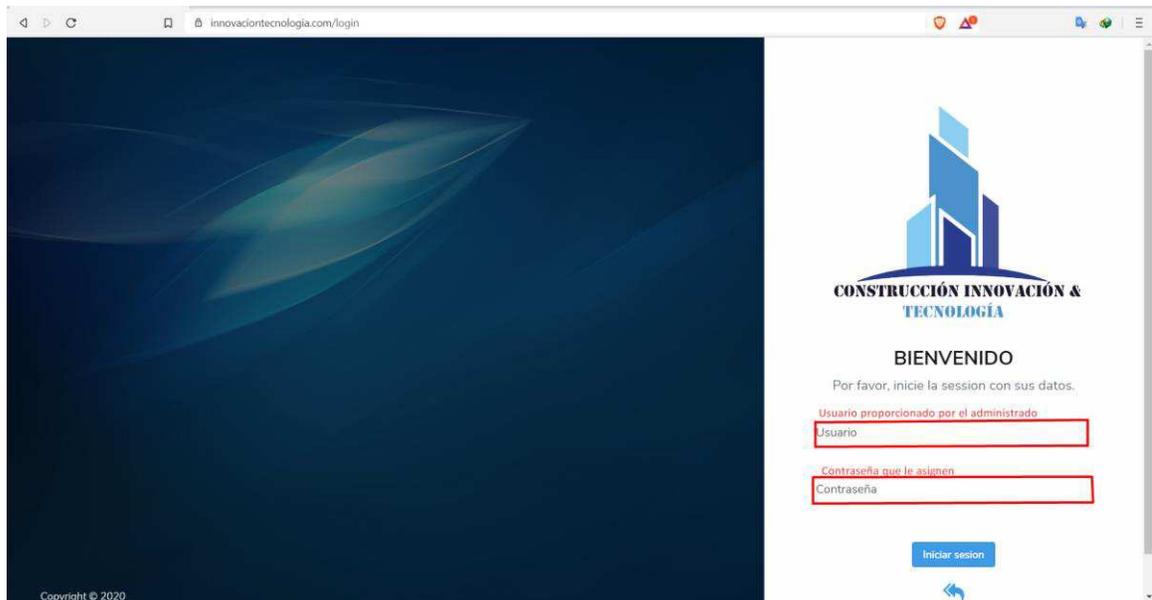
## Barra de menú principal disponibles:

- Seleccionar la pestaña INICIO la cual nos devuelve a la página principal.
- Seleccionar la pestaña QUIENES SOMOS, lleva a misión y visión.
- Seleccionar la pestaña SERVICIOS, la cual describe los servicios que se ofrece.
- Seleccionar la pestaña PROYECTOS, la lleva muestra los alguno de los proyectos realizados.
- Selecciona la pestaña PRODUCTOS, lista los productos disponibles para la venta.
- Selecciona la pestaña CONTÁCTENOS, la cual el cliente envía su consulta al correo corporativo.





Selecciona de la pestaña, el LOGIN: la cual lleva al formulario de inicio de sección del usuario que deberá introducir el nombre de Usuario y Contraseña, la cual es asignado por el Administrador, luego presionar el botón “Iniciar sesión”.



A continuación, ingresará como Administrador a la página principal, donde tenemos:



Seleccionar el botón Configuración del Administrador, donde aparece la opción de Configuración y el de Salir del modo Administrador.



A continuación, ingresa a Configuración: donde aparece la opción de cambiar la información del Administrador, en el botón Cambiar contraseña, además de editar la foto dando clic sobre la imagen.

Bienvenido, GUADALUPE PEREZ QUISPE

Gestiona tu información, privacidad y Seguridad. ✕

DATOS DEL USUARIO

Nombre Completo : GUADALUPE PEREZ QUISPE      CI : 2564123      Telefono : 78945978

Dirección : Z/ Bautista Av/ Siempre Viva      Correo : guada@gmail.com



[Cambiar contraseña](#)

Para cambiar contraseña primero debe ingresar la contraseña actual con el que inicio la sesión y luego establecer la nueva contraseña y confirmarlo según indica el formulario.

Bienvenido, GUADALUPE PEREZ QUISPE

[Cambiar contraseña](#) ✕

Nota: Al actualizar su contraseña con éxito será automáticamente redireccionado al login y deberá iniciar sesión con su nueva contraseña.

Contraseña actual:

Nueva contraseña:

Confirma contraseña:

[Cambiar contraseña](#)

Seleccionar la pestaña INICIO: donde muestra el resumen de las Ventas del Día, Productos Vendidos, Ventas Mensuales, Compras Mensuales y el Balance General.



## Seleccionar la pestaña Plan de Cuentas: Todas las cuentas disponibles

ADMINISTRACION DE PLAN DE CUENTAS

Tabla de Plan de Cuentas Registrados

+ Crear Nuevo Plan de Cuentas

Buscar Número de Cuenta

TIPO DE CUENTA	N° CUENTA	NOMBRE DE CUENTA	ESTADO	ACCIONES
EFFECTIVO Y EQUIVALENTES DE EFFECTIVOS				
ACTIVO	111001	CAJA	Activo	[Editar] [Eliminar]
ACTIVO	111002	FONDO FIJO	Activo	[Editar] [Eliminar]
ACTIVO	111003	BANCOS	Activo	[Editar] [Eliminar]
ACTIVO	111004	INVERSIONES TEMPORALES	Activo	[Editar] [Eliminar]
CUENTAS POR COBRAR				
ACTIVO	112001	CUENTAS POR COBRAR COMERCIALES	Activo	[Editar] [Eliminar]

Crear Nueva Cuenta

Tipo de Cuenta: ACTIVO → Seleccionar el tipo de cuenta

Número Cuenta: Ingrese el Número de Cuenta → Ingresar el numero de cuenta que corresponda

Nombre Cuenta: Ingrese el Tipo de Cuenta → Nombre de la cuenta

Calcular para no registrar ← Cerrar Guardar

Botn para crear nueva Cuenta

TIPO DE CUENTA	N° CUENTA	NOMBRE DE CUENTA	ESTADO	ACCIONES
EFFECTIVO Y EQUIVALENTES DE EFFECTIVOS				
ACTIVO	111001	CAJA	Activo	[Editar] [Eliminar]
ACTIVO	111002	FONDO FIJO	Activo	[Editar] [Eliminar]
ACTIVO	111003	BANCOS	Activo	[Editar] [Eliminar]
ACTIVO	111004	INVERSIONES TEMPORALES	Activo	[Editar] [Eliminar]

Editar la cuenta

Desactivar la cuenta

Seleccionar la pestaña Libro Diario: Registrar y ver los detalles del libro diario.

ADMINISTRACION DE COMPROBANTES DIARIOS

Tabla de Comprobantes Diarios Registrados Boton para crear nuevo registro de libro diario ← + Nuevo Libro Diario

Mostrar 10 registros Buscar:

N°	COMP. N°	FECHA	GLOSA	TOTAL DEBE	TOTAL HABER	REGISTRADO	ACCIONES
1	2	2020-06-24	pago recibido	568.00	0.00	David Condori Callata	
2	1	2020-06-22	Efectivo en caja	102000.00	102000.00	David Condori Callata	

Registros del 1 al 2 de un total de 2 registros

Anterior 1 Siguiente

Imprimir PDF de registro de libro diario

Tabla de Registro del Libro Diario

Nota: Todos los campos con (\*) son obligatorios

Número de Comprobante:  Fecha:

Forma de Pago (\*):  Banco:

Número de Cheque / Deposito:  Tipo de Cambio:  Cambio:

Razon Social (\*):  Concepto (\*):

Glosa (\*):

Nombre de Cuenta:  Debe:  Monto del Debe Haber:  Monto del Haber

Seleccionar Cuenta Agregar detalle de libro diario ← + Agregar

DETALLE DE CUENTAS

ELIMINAR	NOMBRE DE CUENTA	DEBE	HABER
	111001 - CAJA	<input type="text" value="23123"/>	<input type="text" value="0,00"/>
TOTAL:		23123.00	0.00

Guardar todo el detalle del libro diario

Seleccionar la pestaña Libro Mayor: Generar libro mayor y saldos disponibles.

CONSTRUCCIÓN INNOVACIÓN & TECNOLOGÍA

DAVID CONDORI CALLATA

INICIO

CONTABILIDAD

- Plan de Cuentas
- Libro Diario
- Libro Mayor
- Balance General

GESTIONAR PRODUCTOS

- Cientes
- Proveedores

LIBRO MAYOR

SELECCIONE

Fecha Inicio >> << Fecha Fin

Imprimir Buscar

Tabla de Libro Mayor

FECHA	N°COMPR.	NOMBRE DE CUENTA	DEBE	HABER
TOTAL:			0.00	0.00
SALDO:				0.00

Developer ❤️ by David C.C.

LIBRO MAYOR

Seleccionar Cuenta para calcular el mayor

Rango de fecha Inicio

Fecha Fin

SELECCIONE

Fecha Inicio >> << Fecha Fin

Imprimir reporte PDF Imprimir Buscar

Una vez seleccionada buscar por cuenta o por rango de fecha

Tabla de Libro Mayor

FECHA	N°COMPR.	NOMBRE DE CUENTA	DEBE	HABER
2020-06-22	1	CAJA	102,000.00	0.00
TOTAL:			102,000.00	0.00
Saldo disponible de la cuenta			SALDO:	102,000.00

Seleccionar la pestaña Balance General: Donde se imprime la lista de balance.

**BALANCE GENERAL** Una vez establecido las fechas presionar en buscar

Fecha inicio para calcular ← Fecha Inicio >> << Fecha Fin → Fecha final hasta dnde calcular

Imprimir <img alt="Imprimir icon" data-bbox="740 168 755 183"/> <img alt="Buscar icon" data-bbox="815 168 830 183"/> Buscar

Tabla de Balance General Imprimir generado del Balance

NOMBRE DE CUENTA	DEBE	HABER
<u>ACTIVO</u>		
<u>PASIVO</u>		
<u>PATRIMONIO</u>		
Total de los activos existentes de cada cuenta ←	TOTAL ACTIVO:	<input style="width: 100px;" type="text"/>
Total de los pasivos existente spor cuenta ←	TOTAL PASIVO:	<input style="width: 100px;" type="text"/>
Total patrimonio existente por cuenta ←	TOTAL PATRIMONIO:	<input style="width: 100px;" type="text"/>
Total del balance calculado entre las fechas seleccionadas ←	BALANCE GENERAL:	<input style="width: 100px;" type="text"/> .LL

Seleccionar la pestaña Clientes: Crear nuevos clientes y administrar.

**ADMINISTRACION DE CLIENTES**

Tabla de Clientes Registrados Abre el formulario para registrar un nuevo cliente ←  + Crear Nuevo Cliente

Mostrar  registros Buscar:

Nº	CLIENTE	DOCUMENTO	NºDOCUMENTO	EXPEDITO	TELEFONO	ESTADO	ACCIONES
1	DEYSI LOPEZ MIRANDA	NIT	7228579	CHUQUISACA	2285798 <small>ver detalle</small>	Activo	  
2	CAMILA QUISPE QUISPE	NIT	8245725013	COCHABAMBA	71985468 <small>editar</small>	Activo	  
3	MARIO JUARES PEREZ	NIT	7121212013	LA PAZ	2545556 <small>desactivar al cliente o activar</small>	Activo	  

Llenar los Datos del Cliente Llenar el Formulario segun lo solicitado ✕

Nombre

Dirección

Documento

Número documento  seleccionar NIT o CI tipo de documnto

Expedito

Telefono  Seleccionar lugar de expedicion

Correo  Permite Guardar el registro

Boton para calcelar y salir del formulario
✕ Cerrar
Guardar

## Seleccionar la pestaña Proveedores: Administrar proveedores

CONTABILIDAD

- Plan de Cuentas
- Libro Diario
- Libro Mayor
- Balance General

GESTIONAR PRODUCTOS

- Cientes
- Proveedores**
- Categorías
- Productos
- Compras
- Ventas

ADMINISTRACION

- Archivos
- Usuarios

admin@gmail.com

### ADMINISTRACION DE PROVEEDORES

Tabla de Proveedores Registrados

Mostrar 10 registros

+ Crear Nuevo Proveedor

Buscar:

N°	CLIENTE	DOCUMENTO	N°DOCUMENTO	EXPEDITO	TELEFONO	ESTADO	ACCIONES
1	Martin Ortega Perez	NIT	3659944	LA PAZ	77754865	Active	<span style="color: #007bff; font-size: 1.2em;">↶</span> <span style="color: #007bff; font-size: 1.2em;">↷</span> <span style="color: red; font-size: 1.2em;">✕</span>

Registros del 1 al 1 de un total de 1 registros

Anterior
1
Siguiente

Codebase 3.3 © 2020 Developer by David C.C.

Seleccionar la pestaña Categorías: Registrar categorías para nuevos productos.

ADMINISTRACION DE CATEGORIAS

Tabla de Categorías Registrados + Crear Nueva Categoría

Mostrar  registros Buscar:

N°	NOMBRE	DESCRIPCION	FECHA CREACION	FECHA ACTUALIACION	ESTADO	ACCIONES
1	MAQUINARIA	Todo tipo de maquinaria de construcción	2020-07-01 16:48:57	2020-07-01 16:48:57	Activo	 
2	HERRAMIENTAS	Todo tipo de herramienta de construcción	2020-06-29 18:45:05	2020-06-29 18:45:05	Activo	 

Registros del 1 al 2 de un total de 2 registros Anterior **1** Siguiente

Seleccionar la pestaña Productos: Registrar e imprimir detalles.

ADMINISTRACION DE PRODUCTOS

Tabla de Productos Registrados Abrir formulario de registro de nuevo producto

Generar reporte genera de todos los productos Reporte PDF + Crear Nuevo Producto

Mostrar  registros Buscar:

N°	CATEGORIA	PRODUCTO	CODIGO	DESCUENTO %	P. V. C/U	STOCK	IMAGEN	ESTADO	ACCIONES
1	HERRAMIENTAS	Soldadora 200	HSI4567	0	Bs. 9000.00	0		Desactivado	  
2	HERRAMIENTAS	Sierra Circular	HSC	0	Bs. 800.00	0		Desactivado	  
3	HERRAMIENTAS	Pulidora	HPHM1234	0	Bs. 700.00	0		Desactivado	  

Seleccionar la pestaña Compras: Para realizar compras.

ADMINISTRACION DE COMPRAS

Tabla de Compras Registrados Abrir formulario para registrar nuevas compras [+ Nueva Compra](#)

Mostrar  registros Buscar comprar por comprador por fecha o proveedor

N°	FECHA	PROVEEDOR	N°COMPRA	IVA	TOTAL	COMPRADOR	ESTADO	ACCIONES
1	02 de julio 2020	JUAN GABRIEL LOPEZ MAMANI	3	0.13	Bs. 11,017.50	David Condori Callata	Registrado	<a href="#">Ver detalle de la compra</a>
2	29 de junio 2020	BORIS LIMACHI NACHO	2	0.13	Bs. 1,808.00	David Condori Callata	Anulado	<a href="#">Anular compra</a>
3	29 de junio 2020	ANDRES LIMA GONZALES	1	0.13	Bs. 14,520.50	David Condori Callata	Registrado	<a href="#">Imprimir detalles de la compra</a>

Tabla de Registro de Compras Salir del formulario de registro de compras

Nota: Todos los campos con (\*) son obligatorios

Número de Compra (\*)  Documento (\*)  [Seleccionar Factura o recibo](#)

Nombre del Proveedor (\*)  Producto (\*)  [Seleccionar el producto a comprar](#)

Cantidad (\*)  Precio de Compra por Unidad (\*)  [Establecer precio de compra unitaria del producto](#)

[+ Agregar Detalle Compra](#)

DETALLE DE COMPRAS DE LOS DIVERSOS PRODUCTOS

ELIMINAR	PRODUCTO	PRECIO (BS)	CANTIDAD	SUBTOTAL (BS)
	MAPI3698 Atornillador	<input type="text" value="480"/>	<input type="text" value="6"/>	Bs. 2880
TOTAL:				Bs. 2880.00
TOTAL IVA (13%):				Bs. 374.40
TOTAL PAGAR:				Bs. 3254.40

[Realizar Compra](#) Registrar la compra

Seleccionar la pestaña Ventas: para ver las ventas y realizar la venta.

ADMINISTRACION DE VENTAS

Abrir registro de nueva venta

Tabla de Ventas Registrados Lista de Facturas Anulados

Facturas Anulados Nueva Venta

Mostrar 10 registros Buscar:

Imprimir factura

Nº	FECHA	CLIENTE	NºVENTA	IVA	TOTAL	VENDEDOR	ESTADO	ACCIONES
1	02 de julio 2020	JUAN PEREZ LOPEZ	6	0.16	Bs. 638.00	LIZETH FLORES MIRANDA	Registrado	
2	02 de julio 2020	DEYSI LOPEZ MIRANDA	5	0.16	Bs. 6,728.00	David Condori Callata	Registrado	

Anular una venta

Detalles de la venta

Tabla de Registro de Ventas

Nota: Todos los campos con (\*) son obligatorios

Número de Venta: 7 Documento (\*): FACTURA

Nombre del Cliente (\*): JUAN PEREZ LOPEZ Producto (\*): MAPI3698 Atornillador

Cantidad (\*): Ingrese cantidad Stock: 14

Precio Venta: Ingrese precio de venta Descuento en %: 0

Establecer descuento o por defecto el descuento que es 0

cantidad de productos a vender agregar productos para la venta

+ Agregar Detalle Venta

DETALLE DE VENTA DE LOS DIVERSOS PRODUCTOS					
ELIMINAR	PRODUCTO	PRECIO VENTA (BS.)	DESCUENTO	CANTIDAD	SUBTOTAL (BS.)
	MAPI3698 Atornillador	580,00	0,00	2	Bs. 1160.00
TOTAL:					Bs. 1160.00
IVA + IT ( 16% ):					Bs. 185.60
TOTAL PAGAR:					Bs. 1345.60

Detalle de los productos para vender y sus totales y mas impuestos

Realizar Venta Boton para Efectuar la Venta

FACTURAS ANULADOS

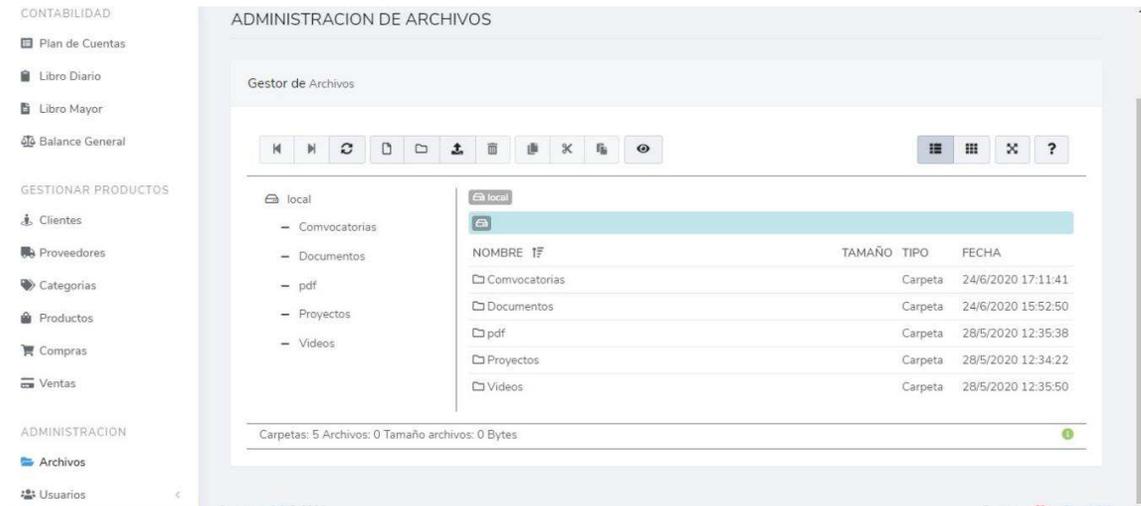
Lista de Facturas Anulados

Mostrar 10 registros Buscar:

Nº	FECHA	DOCUMENTO	NºFACTURA	TOTAL	ESTADO
1	2020-06-29 19:05:28	FACTURA	1	1084.60	Anulado

Registros del 1 al 1 de un total de 1 registros Anterior 1 Siguiente

Seleccionar la pestaña Archivos: En la cual puede guardar diferentes documentos.



Seleccionar la pestaña Usuarios: se desplazará en la parte inferior los usuarios existentes.



La cual contempla Usuarios, roles, Acciones realizadas por el usuario.

## Seleccionar Usuarios. Todos los usuarios registrados.

ADMINISTRACION DE USUARIOS

Tabla de Usuarios Registrados Abrir Formulario para registrar nuevo usuario [+ Crear Nuevo Usuario](#)

Mostrar 10 registros Buscar un usuario segun sus datos registrados

N°	IMAGEN	NOMBRE	CORREO	USUARIO	ROL	ESTADO	ACCIONES
1		Martin Ortega Lopez	martin@gmail.com	martinol	Sin Asignar <small>Ver detalles de usuario</small>	Inactivo	  
2		NATALY GUERRA APURI	naty@gmail.com	natalyga	Sin Asignar <small>Editar un usuario</small>	Activo	  
3		SANTIAGO NACHO CORNEJO	santi@gmail.com	santiagonc	Vendedor <small>Eliminacion logica de un usuario, sin acceso al sistema</small>	Activo	  

Crear Nuevo Usuario Llenar el formulario segun lo indicado ✕

Nombre

Dirección

Documento Tipo de documento

Número documento

Expedito Lugar de expedición del documento

Telefono

Correo

Rol

Usuario  establecer un nombre de usuario para que ingrese al sistema

Contraseña  Establecer una clave para permitir el ingreso al sistema a los usuarios

Imagen  Ningún archivo seleccionado

Registrar al usuario

seleccionar una fotografia o dejar la imagen por defecto Salir del formulario

## Roles Disponibles para el sistema según solicitado.

ADMINISTRACION DE ROLES DISPONIBLES

Tabla de Roles Registrados

Mostrar 10 registros

N°	ROL	DESCRIPCION	ESTADO	ACCIONES
1	Sin Asignar	El Usuario no tiene ningún rol asignado	Activo	
2	Vendedor	Encargado de las Ventas	Activo	
3	Auxiliar	Auxiliar de Contabilidad	Activo	
4	Contador	Contador Autorizado para la empresa	Activo	
5	Administrador	Administrador	Activo	

## Acciones del usuario que son realizados dentro del sistema.

ADMINISTRACION DE ACCIONES

Acciones del usuario en el sistema

Mostrar 10 registros

Salir del panel de acciones

Buscar por fecha o nombre

En la URL interno que altero

El usuario que altero los datos

Lista de los datos alterados nuevo o antiguos

T	FECHA - HORA	CAUSADO EN	DESCRIPCIÓN	NOMBRE	USUARIO	ACCIONES
1	hoy a las 9:05	AppProducto	updated	GUADALUPE PEREZ QUISPE	guadalupepq	
2	hoy a las 8:24 Hora o fecha en el cual se realizo la accion	AppDetalleVenta	created	LIZETH FLORES MIRANDA	lizethfm	
3	hoy a las 8:24	AppVenta	created	LIZETH FLORES MIRANDA	lizethfm	

Nombre completo del usuario causante