



UNIVERSIDAD TÉCNICA DEL NORTE



INSTITUTO DE POSTGRADO

MAESTRÍA EN INGENIERÍA DE SOFTWARE

**“GUÍA METODOLÓGICA PARA LA GESTIÓN DE REQUERIMIENTOS DE
DESARROLLO DE SOFTWARE DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO
MUNICIPAL DE SAN MIGUEL DE IBARRA”**

Trabajo de Grado previo a la obtención del Título de Magíster en Ingeniería de Software

TUTOR:

Ing. MSc. Fernando Garrido Sánchez

AUTORA:

Ing. Rosa Andrea Rea Lozada

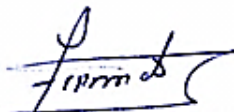
IBARRA - ECUADOR

2017

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de grado: “GUÍA METODOLÓGICA PARA LA GESTIÓN DE REQUERIMIENTOS DE DESARROLLO DE SOFTWARE DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPAL DE SAN MIGUEL DE IBARRA” presentado por la Ing. Rosa Andrea Rea Lozada para optar por el grado de Magíster en Ingeniería de Software, doy fe que dicho trabajo reúne los requisitos y méritos suficientes para ser sometido a presentación (pública o privada) y evaluación por parte del Jurado Examinador que se designe.

En la ciudad de Ibarra, a los 15 días del mes de Septiembre del 2017.



.....
Ing. MSC. Fernando Garrido Sánchez
C.C. 1707852081

CARTA DE ACEPTACION MIEMBROS TRIBUNAL



UNIVERSIDAD TECNICA DEL NORTE
INSTITUTO DE POSGRADO



CARTA DE ACEPTACION TUTORES

13 de Noviembre de 2017

Magíster
Jorge Caraguay
DIRECTOR POSGRADO UTN




De nuestras consideraciones:

Nos permitimos informar a usted que revisado el Trabajo de Grado de la maestrante: Rosa Andrea Rea Lozada, del Programa de Maestría en: Ingeniería de Software, con el tema: "Guía metodológica para la gestión de requerimientos de desarrollo de software del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra" tenemos a bien certificar que han sido acogidas y satisfechas todas las observaciones realizadas en la defensa privada.

En tal virtud, facultamos empastar el mencionado trabajo y que su tutor solicite fecha para defensa pública.

Agradecemos su atención.

Atentamente,

	Apellidos y Nombres	Firma
Miembro Tribunal 1:	Magister Fausto Salazar	
Miembro Tribunal 2:	Magister Antonio Quiña	
Miembro Tribunal 3:	Magister Vicente Guevara	

AUTORÍA

Yo, Rea Lozada Rosa Andrea con cédula Nro. 1002961454, declaro bajo juramento que el trabajo aquí descrito es de mi autoría, que no ha sido previamente presentado para ningún grado ni calificación profesional, que he consultado referencias bibliográficas que se incluyen en este documento y que todos los datos presentados son resultado de mi trabajo.



Ing. Rosa Andrea Rea Lozada
C.C. 1002961454



UNIVERSIDAD TÉCNICA DEL NORTE
BIBLIOTECA UNIVERSITARIA

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

1. Identificación de la Obra

La Universidad Técnica del Norte dentro del proyecto de Repositorio Digital Institucional, determinó la necesidad de disponer de textos completos en formato digital con la finalidad de apoyar los procesos de investigación, docencia y extensión de la Universidad. Por medio del presente documento dejo sentada mi voluntad de participar en este proyecto, para lo cual pongo a disposición la siguiente información:

DATOS DE CONTACTO	
CÉDULA DE IDENTIDAD:	1002961454
APELLIDOS Y NOMBRES:	Rea Lozada Rosa Andrea
DIRECCIÓN:	Ibarra
EMAIL:	andrearea87@gmail.com
TELÉFONO:	0989746332

DATOS DE LA OBRA	
TÍTULO:	“Guía metodológica para la gestión de requerimientos de desarrollo de software del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra”
AUTOR:	Ing. Rosa Andrea Rea Lozada
FECHA:	16 de noviembre del 2017
PROGRAMA:	Posgrado
TÍTULO POR EL QUE OPTA:	Magíster en Ingeniería de Software
DIRECTOR:	Ing. MSc. Fernando Garrido Sánchez

2. Autorización a favor de la Universidad


Yo, Rosa Andrea Rea Lozada, con cédula de identidad Nro. 1002961454, en calidad de autor y titular de los derechos patrimoniales del trabajo de la obra o trabajo de grado descrito anteriormente, hago entrega del ejemplar respectivo en formato digital y autorizo a la Universidad Técnica del Norte, la publicación de la obra en el Repositorio Digital Institucional y uso del archivo digital en la Biblioteca de la Universidad con fines académicos, para ampliar la disponibilidad del material y como apoyo a la educación, investigación y extensión; en concordancia con la Ley de Educación Superior Artículo 144.

3. Constancias

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

Ibarra, a los 16 días del mes de noviembre del 2017

Autor


Ing. Rosa Andrea Rea Lozada
C.C: 1002961454



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

Yo, Rosa Andrea Rea Lozada, con cédula de identidad Nro.1002961454, manifiesto mi voluntad de ceder a la Universidad Técnica del Norte los derechos patrimoniales consagrados en la Ley de Propiedad Intelectual del Ecuador, artículos 4, 5 y 6, en calidad de autor de la obra o trabajo de grado denominado “GUÍA METODOLÓGICA PARA LA GESTIÓN DE REQUERIMIENTOS DE DESARROLLO DE SOFTWARE DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPAL DE SAN MIGUEL DE IBARRA”, que ha sido desarrollado para optar por el título de Magister en Ingeniería de Software, en la Universidad Técnica del Norte, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente. En mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia suscribo este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Técnica del Norte.

Ibarra, a los 16 días del mes de noviembre del 2017.

A handwritten signature in blue ink, appearing to read 'Rosa Andrea Rea Lozada', is written over a faint, illegible stamp.

Ing. Rosa Andrea Rea Lozada

C.C: 1002961454

DEDICATORIA

*A mi madre, quien con su lucha diaria me ha demostrado
el significado del amor incondicional.*

Andrea

AGRADECIMIENTO

Al Instituto de Posgrado de la Universidad Técnica del Norte, por la acertada decisión de ofrecer programas de estudio afines a la carrera de Ingeniería en Sistemas.

A la Dirección de Tecnologías de la Información del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra, por la apertura y el esfuerzo entregado para el desarrollo de este trabajo.

A mi tutor Ing. Fernando Garrido Sánchez, por compartir su valioso tiempo y conocimientos conmigo.

A los miembros del tribunal calificador, docentes de la Universidad Técnica del Norte, por los aportes realizados a este trabajo.

Andrea

ÍNDICE DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	ii
CARTA DE ACEPTACION MIEMBROS TRIBUNAL	iii
AUTORÍA.....	iv
AUTORIZACIÓN DE USO Y PUBLICACIÓN A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	v
CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE GRADO A FAVOR DE LA UNIVERSIDAD TÉCNICA DEL NORTE	vii
DEDICATORIA	viii
AGRADECIMIENTO.....	ix
RESUMEN.....	x
SUMMARY	xii
1. CAPÍTULO I. EL PROBLEMA	1
1.1 Antecedentes	1
1.2 Planteamiento del Problema.....	3
1.3 Formulación del Problema	4
1.4 Justificación de la Investigación	4
1.5 Objetivos de la Investigación	5
1.5.1 Objetivo General	5
1.5.2 Objetivos Específicos.....	5
2. CAPÍTULO II. MARCO REFERENCIAL	6
2.1 La Ingeniería de Software	6
2.1.1 Antecedentes	6
2.1.2 Definición de Software.....	6
2.1.3 Definición de Ingeniería de Software.....	7
2.1.4 El Proceso del Software	7
2.1.5 Modelos del Ciclo de Vida de Desarrollo del Software	8
2.1.6 Estándares Internacionales de Ingeniería de Software	9
2.2 La Ingeniería de Requerimientos de Software	11
2.2.1 Definición de Requerimiento de Software	11
2.2.2 Definición de Ingeniería de Requerimientos	11
2.2.3 Tipos de requerimientos de software.....	13
2.2.4 Características de un buen requerimiento de software	15
2.2.5 Atributos de requerimientos de software.....	16
2.2.6 El Analista de Requerimientos de software	16

2.2.7	Definición de Stakeholder	17
2.2.8	Especificación de Requerimientos de Software	18
2.2.9	Taxonomía de Ingeniería de Requerimientos.....	19
2.2.10	Actividades de la Ingeniería de Requerimientos	19
2.3	Desarrollo de Requerimientos	20
2.3.1	Elicitación de requerimientos	20
2.3.2	Análisis de requerimientos	21
2.3.3	Especificación de requerimientos.....	23
2.3.4	Validación y Verificación de requerimientos.....	25
2.4	Administración de Requerimientos	26
2.4.1	Control de versiones.....	27
2.4.2	Control de cambios.....	27
2.4.3	Trazabilidad de requerimientos	30
2.4.4	Seguimiento a requerimientos	31
2.5	Herramientas de Gestión de Requerimientos de Software	32
2.5.1	Herramientas de Administración del Ciclo de Vida del Software.....	32
2.5.2	Limitaciones del uso de Especificaciones de Requerimientos	32
2.5.3	Definición de herramienta de gestión de requerimientos	33
2.5.4	Características de una herramienta de gestión de requerimientos	33
2.5.5	Herramientas de gestión de requerimientos existentes en el mercado	35
3.	CAPÍTULO III. MATERIALES Y MÉTODOS	36
3.1	Lugar de desarrollo de la investigación.....	36
3.1.1	Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra	36
3.1.2	Dirección de Tecnologías de la Información y Comunicación	36
3.1.3	Unidad de Desarrollo de Software	37
3.2	Tipo de Investigación	38
3.3	Diseño de la Investigación	38
3.4	Población de Estudio	40
3.5	Metodología de trabajo para el desarrollo del Objetivo Específico 1	42
3.6	Metodología de trabajo para el desarrollo del Objetivo Específico 2	47
3.7	Metodología de trabajo para el desarrollo del Objetivo Específico 3	49
3.7.1	Levantamiento y Análisis de requerimientos	50
3.7.2	Diseño	50
3.7.3	Construcción.....	50
3.7.4	Pruebas	50
3.7.5	Implantación.....	50

4. CAPÍTULO IV. DESARROLLO DE LA PROPUESTA	51
4.1. Desarrollo del Objetivo Específico 1	51
4.1.1. Definir una taxonomía de tipos de requerimientos.....	51
4.1.2. Determinar atributos de requerimientos	54
4.1.3. Determinar tipos de stakeholders	56
4.1.4. Definir el proceso de gestión de requerimientos de software.....	58
4.1.5. Definir artefactos y proceso para el control de versiones.....	71
4.1.6. Definir artefactos y proceso para mantener trazabilidad	73
4.1.7. Definir artefactos y proceso para el control de cambios	74
4.1.8. Definir el proceso para seguimiento de requerimientos	76
4.1.9. Definir artefactos y proceso para el mantenimiento de software	79
4.1.10. Elaborar formatos para los artefactos definidos	80
4.1.11. Definir políticas del proceso de gestión de requerimientos de software	88
4.1.12. Documentación de la guía metodológica	89
4.2. Desarrollo del Objetivo Específico 2	90
4.2.1. Seleccionar un conjunto de herramientas de gestión de requerimientos de software existentes en el mercado para el estudio.....	90
4.2.2. Determinar criterios de agrupamiento de las características funcionales de una herramienta de gestión de requerimientos de software	93
4.2.3. Verificar las funcionalidades de cada herramienta mediante la instalación, uso, y revisión de documentación técnica, dependiendo del caso.	94
4.2.4. Determinar las características funcionales de un sistema de gestión de requerimientos de software, que se considerarán para el presente estudio.	101
4.3. Desarrollo del Objetivo Específico 3	103
4.3.1. Módulos de la aplicación informática	104
4.3.2. Descripción del Módulo de Configuración del Sistema.....	104
4.3.3. Descripción del Módulo de Inventario de Sistemas	110
4.3.4. Descripción del Módulo de Mantenimiento de Sistemas	111
4.3.5. Descripción del Módulo de Gestión de Proyectos de Desarrollo	117
5. CAPÍTULO V. RESULTADOS Y DISCUSIÓN.....	124
5.1. Introducción	124
5.2. Proceso de gestión de requerimientos de software.....	124
5.3. Características funcionales seleccionadas para la aplicación informática.....	124
5.4. Aplicación informática para la gestión de requerimientos de software.....	126
5.5. Guía Metodológica para la gestión de requerimientos de software	127
5.5.1. Planeación de la medición del proceso.....	127
5.5.2. Medición del proceso	129

5.5.3. Análisis de los Resultados.....	135
6. CAPÍTULO VI. CONCLUSIONES Y RECOMENDACIONES.....	137
6.1. CONCLUSIONES	137
6.2. RECOMENDACIONES	138
BIBLIOGRAFÍA.....	139
ANEXOS	142
Anexo 1: Árbol de problemas detectados en la Unidad de Desarrollo de Software	143
<u>Anexo 2: Árbol de Objetivos para la Unidad de Desarrollo de Software</u>	<u>144</u>
Anexo 3: Modelo Entidad Relación de la aplicación informática desarrollada	145
Anexo 4: Formato de Acta de Reunión.....	146
Anexo 5: Formato de Proyecto de Software	147
Anexo 6: Datos levantados para la medición de atributos de la entidad Esfuerzo.....	149
Anexo 7: Datos levantados para la medición de atributos de la entidad Unidad de Trabajo	150
Anexo 8: Guía metodológica para la gestión de requerimientos de software del GAD-I.....	152

ÍNDICE DE TABLAS

Tabla 1: Problemas de la gestión de proyectos de desarrollo de software – Ecuador	2
Tabla 2: Fases del ciclo de vida del software	7
Tabla 3: Características de modelos de ciclo de vida de desarrollo de software	8
Tabla 4: Estándares Internacionales de Ingeniería de Software	9
Tabla 5: Tipos de información de requerimientos de software	13
Tabla 6: Características de un buen requerimiento de software	15
Tabla 7: Descripción de actividades de la Ingeniería de Requerimientos	19
Tabla 8: Técnicas de elicitación de requerimientos	22
Tabla 9: Características de una ERU y una ERS	24
Tabla 10: Conceptos y ejemplos de validación y verificación de requerimientos	25
Tabla 11: Proceso de inspección de requerimientos de software	26
Tabla 12: Artefactos impactados y estrategia de implementación de una petición de cambio	28
Tabla 13: Beneficios de implementar trazabilidad en el desarrollo de software	30
Tabla 14: Relaciones de trazabilidad entre artefactos del proyecto	31
Tabla 15: Tipos comunes de problemas en requerimientos	32
Tabla 16: Herramientas comerciales de Ingeniería de Requerimientos e Ingeniería de Software	35
Tabla 17: Sistemas informáticos existentes en la Dirección de TIC – 2017	37
Tabla 18: Técnicas usadas en la investigación	39
Tabla 19: Población de estudio	40
Tabla 20: Características de la Unidad de Desarrollo de Software del GAD-I	41
Tabla 21: Metodología de trabajo para el desarrollo del primer objetivo específico	44
Tabla 22: Metodología de trabajo para el desarrollo del Objetivo Específico 2	48
Tabla 23: Definiciones de tipos de requerimientos a ser usados en la Dirección de TIC	53
Tabla 24: Atributos de requerimientos a ser usados en la Dirección de TIC	55
Tabla 25: Descripciones de atributos de requerimientos a ser usados en la Dirección de TIC	56
Tabla 26: Tipos de stakeholders a ser usados en la Dirección de TIC	57
Tabla 27: Actividades definidas para el proceso de desarrollo de software	58
Tabla 28: Actividades definidas para el proceso de modificación de software	59
Tabla 29: Conocimiento y experiencia en técnicas de elicitación de requerimientos	60
Tabla 30: Entradas, tareas, responsables y artefactos para el despegue del proyecto	62
Tabla 31: Entradas, tareas, responsables y artefactos para la elicitación de requerimientos	64
Tabla 32: Entradas, tareas, responsables y artefactos para el análisis de requerimientos	67
Tabla 33: Entradas, responsables y artefactos para la especificación de requerimientos	69
Tabla 34: Entradas, tareas, responsables y artefactos para la validación de requerimientos	70
Tabla 35: Análisis de la importancia de control de versiones en artefactos del proceso	71
Tabla 36: Tareas, responsables y artefactos del control de versiones	73
Tabla 37: Tareas, responsables y artefactos para mantener trazabilidad	74
Tabla 38: Tareas, responsables y artefactos del control de cambios	75
Tabla 39: Estados de requerimientos funcionales	77
Tabla 40: Tareas y responsables del seguimiento a requerimientos	78
Tabla 41: Tareas, responsables y artefactos del proceso de mantenimiento de software	79
Tabla 42: Definiciones de artefactos del proceso	81
Tabla 43: Formato de Registro de Stakeholders	83
Tabla 44: Formato de registro de información relacionada	83
Tabla 45: Formato de Registro de sistemas relacionados	84
Tabla 46: Formato de Glosario de términos y abreviaturas	84
Tabla 47: Formato de observación para levantar información	84
Tabla 48: Formato de entrevista para levantar información	84

Tabla 49: Formato de Historial de cambios	85
Tabla 50: Formato de Matriz de Trazabilidad	85
Tabla 51: Formato Caso de Prueba	85
Tabla 52: Registro de Requerimientos – Análisis	86
Tabla 53: Registro de Requerimientos – Especificación	86
Tabla 54: Registro de Requerimientos – Validación	87
Tabla 55: Formato de Petición de cambios	87
Tabla 56: Formato Orden de Mantenimiento	88
Tabla 57: Listado de herramientas de gestión de requerimientos encontradas	90
Tabla 58: Herramientas de gestión de requerimientos con sus características generales	92
Tabla 59: Listado de herramientas seleccionadas para el estudio	93
Tabla 60: Paquetes e información obtenida de las herramientas seleccionadas	94
Tabla 61: Características funcionales de la herramienta Case Complete 2016	95
Tabla 62: Características funcionales de la herramienta Accompa	96
Tabla 63: Características funcionales de la herramienta Caliber 11.5	97
Tabla 64: Características funcionales herramienta CodeBeamerRequirements Management 8	99
Tabla 65: Características funcionales de la herramienta Gatherspace 2.2	100
Tabla 66: Características funcionales seleccionadas para la aplicación informática	101
Tabla 67: Funcionalidades módulo Proceso de Desarrollo	105
Tabla 68: Funcionalidades módulo Artefactos del Proceso	107
Tabla 69: Funcionalidades módulo de Inventario de Sistemas	110
Tabla 70: Funcionalidades Registro de Proyectos	119
Tabla 71: Organización del trabajo para medición del proceso	127
Tabla 72: Entidades y atributos seleccionados para la medición del proceso	128
Tabla 73: Incremento de valores de atributos de la entidad Esfuerzo	129
Tabla 74: Incremento de valores de atributos de la entidad Unidad de Trabajo	130
Tabla 75: Datos de medición de la entidad Tiempo para el grupo G02	131
Tabla 76: Datos de medición de la entidad Tiempo para el grupo G01	132
Tabla 77: Incremento de valores de atributos de la entidad "Tiempo"	132
Tabla 78: Resultados encuesta usuarios GAD-I	134
Tabla 79: Análisis de Resultados	135

ÍNDICE DE FIGURAS

<i>Figura 1: Gestión de requerimientos de software</i>	13
<i>Figura 2: Relaciones entre los diferentes tipos de información de requerimientos</i>	15
<i>Figura 3: Diagrama de cebolla de stakeholders</i>	17
<i>Figura 4: Vista general del estándar ISO/IEC/IEEE 29148-2011</i>	18
<i>Figura 5: Actividades del desarrollo de requerimientos</i>	20
<i>Figura 6: Actividades de la administración de requerimientos</i>	27
<i>Figura 7: Taxonomía de fuentes de cambios en requerimientos de software</i>	29
<i>Figura 8: Diagrama de transición de estados para el control de cambios</i>	30
<i>Figura 9: Características de una herramienta de gestión de requerimientos</i>	34
<i>Figura 10: Estructura Orgánica de la Dirección de TIC del GAD-I</i>	37
<i>Figura 11: Metodología de trabajo para el objetivo específico 1 - Primera Parte</i>	42
<i>Figura 12: Metodología de trabajo para el objetivo específico 1 - Segunda Parte</i>	43
<i>Figura 13: Metodología de trabajo para el cumplimiento del segundo objetivo específico</i>	47
<i>Figura 14: Metodología de desarrollo de la aplicación informática</i>	49
<i>Figura 15: Taxonomía de tipos de requerimientos de software para la Dirección de TIC</i>	52
<i>Figura 16: Diagrama - Realizar despegue del proyecto</i>	63
<i>Figura 17: Diagrama - Elicitar y Recolectar Requerimientos</i>	66
<i>Figura 18: Diagrama - Analizar Requerimientos</i>	68
<i>Figura 19: Diagrama - Especificar Requerimientos</i>	70
<i>Figura 20: Diagrama - Validar y Verificar Requerimientos</i>	71
<i>Figura 21: Diagrama de transición de estados de la petición de cambio</i>	75
<i>Figura 22: Diagrama - Control de Cambios en Requerimientos</i>	77
<i>Figura 23: Diagrama de transición de estados de requerimientos funcionales</i>	78
<i>Figura 24: Estados de órdenes de mantenimiento</i>	80
<i>Figura 25: Diagrama - Mantenimiento de sistemas</i>	81
<i>Figura 26: Diagrama de arquitectura de la aplicación informática</i>	103
<i>Figura 27: Diagrama de módulos de la aplicación informática</i>	104
<i>Figura 28: Pantalla - Módulos de la aplicación</i>	104
<i>Figura 29: Pantalla - Sub módulos Configuración del Sistema</i>	105
<i>Figura 30: Pantalla - Proceso de Desarrollo de Software</i>	106
<i>Figura 31: Pantalla - Agregar y Editar artefactos al proceso</i>	106
<i>Figura 32: Pantalla - Catálogo de Artefactos</i>	107
<i>Figura 33: Pantalla - Registro de Artefactos</i>	108
<i>Figura 34: Pantalla - Catálogo de Tipo de Documentación Técnica</i>	108
<i>Figura 35: Pantalla - Catálogo de Tipo de Sistemas</i>	109
<i>Figura 36: Pantalla - Catálogo Tipo de Stakeholders</i>	109
<i>Figura 37: Pantalla - Inventario de Sistemas</i>	110
<i>Figura 38: Pantalla - Datos del Sistema</i>	111
<i>Figura 39: Pantalla - Módulos Mantenimiento de Sistemas</i>	111
<i>Figura 40: Pantalla - Listado de Órdenes de Mantenimiento (usuario)</i>	112
<i>Figura 41: Pantalla - Detalle Orden de Mantenimiento</i>	113
<i>Figura 42: Pantalla - Órdenes de Mantenimiento (administrador)</i>	113
<i>Figura 43: Pantalla - Detalle de Orden de Mantenimiento</i>	114
<i>Figura 44: Pantalla - Registro de Analista de Sistemas en Orden</i>	114
<i>Figura 45: Pantalla - Mis Órdenes de Mantenimiento</i>	115
<i>Figura 46: Pantalla - Registro de análisis y atención de orden</i>	115
<i>Figura 47: Pantalla - Registro de atención orden</i>	116
<i>Figura 48: Pantalla - Impresión de Orden de Mantenimiento</i>	116

Figura 49: Pantalla - Módulo de Gestión de Proyectos de Desarrollo _____	117
Figura 50: Pantalla - Glosario de Términos y Abreviaturas _____	117
Figura 51: Pantalla - Registro de Información Relacionada _____	118
Figura 52: Pantalla - Registro de Stakeholders _____	118
Figura 53: Pantalla - Registro de Proyectos _____	119
Figura 54: Pantalla - Administración del Proyecto _____	120
Figura 55: Pantalla - Glosario de Términos y Abreviaturas del Proyecto _____	120
Figura 56: Pantalla - Registro de Stakeholders del Proyecto _____	121
Figura 57: Pantalla - Registro de Sistemas Relacionados con el Proyecto _____	121
Figura 58: Pantalla - Registro de Documentación Relacionada del Proyecto _____	122
Figura 59: Pantalla - Artefactos del Proyecto _____	122
Figura 60: Pantalla - Matriz de Trazabilidad del proyecto _____	123
Figura 61: Vista general del proceso de gestión de requerimientos de software _____	124
Figura 62: Características aplicación informática - Desarrollo de Requerimientos _____	125
Figura 63: Características aplicación informática - Administración de Requerimientos _____	125
Figura 64: Arquitectura de la aplicación informática _____	126
Figura 65: Medición de atributos de la entidad "Esfuerzo" para G01 y G02 _____	130
Figura 66: Medición de atributos de la entidad "Unidad de Trabajo" para G01 y G02 _____	131
Figura 67: Medición de atributos de la entidad "Tiempo" para G01 y G02 _____	133
Figura 68: Medición de atributos de la entidad "Personas - Usuarios" en el año 2016 y 2017 _____	134
Figura 69: Resultados de la implantación de la guía metodológica _____	135

**UNIVERSIDAD TÉCNICA DEL NORTE
INSTITUTO DE POSGRADO**

Maestría en Ingeniería de Software

“GUÍA METODOLÓGICA PARA LA GESTIÓN DE REQUERIMIENTOS DE
DESARROLLO DE SOFTWARE DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO
MUNICIPAL DE SAN MIGUEL DE IBARRA”

Autora: Ing. Rosa Andrea Rea Lozada
Tutor: Ing. MSc. Fernando Garrido Sánchez
Año: 2017

RESUMEN

El desarrollo de software abarca actividades de análisis, diseño, desarrollo, pruebas e implantación; y como parte del análisis se debe descubrir, desarrollar, validar y gestionar los requerimientos que especifican un sistema. Pero a medida que se van definiendo más requerimientos, los miembros del equipo de trabajo tienen dificultad para controlar cambios y versiones, definir relaciones entre requerimientos, conocer el estado y el avance de cada requerimiento, identificar requerimientos perdidos, duplicados e innecesarios.

Este es el caso de la Unidad de Desarrollo de Software del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra, donde los problemas existentes en la gestión de requerimientos de software han ocasionado trabajo imprevisto, inexactitud en la estimación de esfuerzo y tiempo, requerimientos omitidos o mal identificados, documentación de requerimientos incompleta, no estandarizada o inexistente, desconocimiento del estado y avance de cada requerimiento, dificultades para control de cambios, retrasos en la entrega del producto software, entre otros.

Al ser una institución regulada por disposiciones gubernamentales, la importancia de atender adecuadamente los requerimientos de software atraviesa el ámbito funcional y legal, debido a que los requerimientos deben ser cuidadosamente documentados, rastreados, revisados y probados. En ese sentido, el presente trabajo tiene como objetivo implementar una guía metodológica de Ingeniería de Requerimientos mediante la definición de un proceso y elaboración de una aplicación informática, que permitan mejorar la gestión de requerimientos de la Unidad de Desarrollo de Software.

Para ello se utilizó un enfoque de investigación cualitativo con la aplicación de técnicas de recolección de datos como la observación, entrevistas abiertas, revisión de documentos y talleres de trabajo que permitieron definir el proceso de gestión de requerimientos y la aplicación informática que lo soporte.

Como resultado se obtuvieron los siguientes productos adaptados a las necesidades de la Unidad de Desarrollo de Software: taxonomía y atributos de requerimientos, técnicas para elicitación de requerimientos, plantillas de artefactos, proceso para la elicitación, análisis, especificación y validación de requerimientos, proceso de control de cambios y control de

versiones, trazabilidad de requerimientos hacia casos de prueba, así como una aplicación informática adaptada al proceso definido.

Durante el desarrollo del trabajo, se constató que los Analistas de Sistemas ya estaban acostumbrados a trabajar informalmente en el desarrollo de software y les resultó dificultoso el uso de un procedimiento documentado con artefactos. Sin embargo, se han obtenido varios beneficios en la Unidad, entre ellos: centralización de información, estandarización de formatos y actividades, mejoras para la identificación y clasificación de requerimientos, disponibilidad de información, control de avances del proyecto, acceso a versiones vigentes de artefactos, sistematización de órdenes de trabajo, entre otros.

Palabras Clave: Ingeniería de Software; Ingeniería de Requerimientos; Requerimiento de software, Elicitación, Análisis, Especificación y Validación de requerimientos, Herramienta de gestión de requerimientos.

**UNIVERSIDAD TÉCNICA DEL NORTE
INSTITUTO DE POSGRADO**

Maestría en Ingeniería de Software

“METHODODOLOGICAL GUIDE FOR SOFTWARE REQUIREMENTS MANAGEMENT OF GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPAL DE SAN MIGUEL DE IBARRA”

Author: Ing. Rosa Andrea Rea Lozada
Tutor: Ing. MSc. Fernando Garrido Sánchez
Year: 2017

SUMMARY

Software development encompasses analysis, design, development, testing and implementation activities; and as a part of the analysis must discover, develop, validate and manage the requirements that specify a system. But as more requirements are defined, team members have difficulty controlling changes and versions, defining relationships between requirements, knowing the status and progress of each requirement, identifying lost, duplicate, and unnecessary requirements.

This is the case of the Software Development Unit of the Municipal Decentralized Autonomous Government of San Miguel de Ibarra, where problems in the management of software requirements have led to unforeseen work, inaccuracies in the estimation of effort and time, poorly identified or omitted requirements, incomplete, non-standard or non-existent documentation of requirements, lack of knowledge of the status and progress of each requirement, difficulties in controlling changes, delays in software product delivery, among others.

As an institution governed by government regulations, the importance of adequately addressing software requirements goes through the functional and legal ambit, because requirements must be carefully documented, tracked, reviewed and tested. In this sense, the present work has as objective to implement a methodological guide of Requirements Engineering through of the definition of a process and elaboration of a computer application, that allow to improve the management of requirements of the Unit of Software Development.

For this purpose, a qualitative research approach was used with the application of data collection techniques such as observation, open interviews, review of documents and workshops that allowed the definition of the requirements management process and the computer application that supports it.

As a result, are obtained the following products adapted to the needs of the Software Development Unit: taxonomy and requirements attributes, requirements elicitation techniques, artifact templates, elicitation, analysis, specification and validation processes, change control and version control processes, traceability of requirements to test cases, as well as a computer application adapted to the defined process.

During the development of the work, it was found that System Analysts were already accustomed to working informally in the development of software and they had difficulty in using a documented procedure with artifacts. However, several benefits have been obtained in the Unit, including: centralization of information, standardization of formats and activities, improvements for identification and classification of requirements, availability of information, control of project progress, access to current versions of artifacts, Systematization of work orders, among others.

Keywords: Software Engineering; Requirements Engineering; Software requirements, Elicitation, Analysis, Specification and Validation of requirements, Requirements management tool.

1. CAPÍTULO I. EL PROBLEMA

1.1 Antecedentes

Durante los últimos años, la Ingeniería de Requerimientos ha captado especial atención por parte de investigadores, profesores, estudiantes y profesionales del área de la Ingeniería de Software, debido a que la identificación y gestión correcta de requerimientos se ha convertido en una pieza clave para el éxito de proyectos de software a nivel mundial.

La Ingeniería de Requerimientos integra actividades de identificación, análisis, documentación, control y validación, con el fin de lograr los resultados deseados en términos de calidad, costo y plazo predefinidos. Sin embargo, muchos proyectos de desarrollo de software han fracasado por la omisión o ejecución inadecuada de estas actividades.

En el informe Chaos Report publicado por el Standish Group en el 2014, se muestran los resultados de una encuesta aplicada a los gerentes de TI a nivel internacional en relación a los factores considerados como causa de fracaso o cancelación de un proyecto de software. Los gerentes de TI indicaron que los factores más importantes para la cancelación de un proyecto de este tipo, eran: requerimientos incompletos, falta de involucramiento del usuario, falta de recursos, expectativas no realistas, requerimientos cambiantes y ausencia de planeación.

En un estudio realizado en la Universidad de Sevilla en España, en el que se buscaba establecer técnicas para la gestión de proyectos de desarrollo de 'software a medida' en empresas medianas, Navascués Fernández (2013) determinó que los principales factores de fracaso en estos proyectos eran la falta de experiencia o capacitación inadecuada de los directores de proyectos, fallos en la identificación, documentación y seguimiento de requerimientos, así como procesos de planificación deficientes.

Por otro lado, en la Universidad Técnica Federico Santa María de Chile, Guerra y Bedini González (2013) reconocen que el éxito de un proyecto de desarrollo de software está fuertemente ligado con la fase de Planificación y establecen que las principales causas de fallas se deben a la inadecuada definición del proyecto, comprensión errónea del problema, definición incompleta de los requerimientos, fallas en la estimación, así como organización errónea del grupo de trabajo.

En nuestro país, estudios de la Escuela Politécnica del Ejército ESPE, han identificado algunos de los problemas derivados de la gestión de proyectos en empresas desarrolladoras de software de las ciudades de Quito y Guayaquil. En la **Tabla 1** se pueden observar los problemas encontrados en este estudio, entre los que se pueden citar: definiciones incompletas de requerimientos, uso de herramientas inadecuadas para gestión de proyectos, falta de definición de entregables, entre otros.

En todos los estudios analizados, uno de los factores comunes para el fracaso de los proyectos de desarrollo de software está relacionado con la gestión de requerimientos; esto, debido a que la Ingeniería de Requerimientos es frecuentemente olvidada por la gestión de proyectos. Por ello es importante comprender que los requerimientos son la base de todo desarrollo, ya que definen lo que los usuarios necesitan y lo que el sistema debe hacer para cumplir sus necesidades.

Tabla 1: Problemas de la gestión de proyectos de desarrollo de software – Ecuador

Fuente: “Guía de fundamentos para la dirección de proyectos de desarrollo de software” por A. Palacios, 2014.

PROBLEMA	CONSECUENCIA
Definiciones incompletas de requerimientos.	Margen de error en la estimación de tiempos. Impacto en los costos y recursos.
La participación y responsabilidades de los miembros del equipo de desarrollo no están formalizadas.	Trabajo imprevisto. Falta de compromiso
Para el seguimiento de proyectos se usan herramientas como hojas de cálculo, documentos, correo electrónico.	Documentación dispersa y sin formatos. Omisión del registro de información importante. Indisponibilidad de la información.
No existe una definición formal de los entregables que deben generarse en cada etapa.	Documentación no estandarizada. Documentación inexistente.
No se ha establecido el grupo de pruebas que deben ejecutarse.	No se garantiza la calidad de los productos de software.
No se tiene una bitácora de lecciones aprendidas.	Falta de información para la toma de decisiones en proyectos futuros

De acuerdo con Berenbach, Paulish y Kazmeie (2009) la Ingeniería de Requerimientos está dedicada a entender y trabajar en la declaración del problema y no en la solución. Sin embargo, los administradores de proyectos suelen pensar que sólo cuando los desarrolladores empiezan a trabajar en la solución, el proyecto empieza a mostrar su progreso real. Por lo tanto, para los administradores como para los desarrolladores, hay una tendencia a precipitarse muy pronto en el desarrollo de la solución.

En el Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra - GAD-I, la Dirección de Tecnologías de la Información y Comunicación - TIC tiene la responsabilidad de aportar con soluciones tecnológicas de software, hardware y comunicaciones que soporten las actividades y procesos institucionales. Para ello, en la Dirección de TIC se establecieron las unidades administrativas de Desarrollo de Software, Infraestructura y Comunicaciones, Innovación y Gobierno Electrónico.

Todos los requerimientos de desarrollo y mantenimiento de software de la institución son reasignados a la Unidad de Desarrollo de Software, donde se tienen sistemas informáticos implementados internamente, implementados por empresas externas, e implantaciones de software libre.

Con el objetivo de hacer frente a la constante evolución de los sistemas informáticos, en la Unidad de Desarrollo de Software se establecieron procedimientos para el desarrollo y mantenimiento de software. Además, en el año 2012 se realizó un estudio con el fin de

implementar las Normas de Control Interno emitidas por la Contraloría General del Estado en la Dirección de TIC, como resultado de un interés por cumplir con la normativa legal y reducir la aparición de problemas encontrados durante la gestión de esta dependencia.

El estudio abarcó varios puntos establecidos en la Norma, tales como: Administración de proyectos tecnológicos, Desarrollo y adquisición de software aplicativo, Seguridad de tecnología de la información, Mantenimiento y control de la infraestructura tecnológica. Sin embargo, por la amplitud del tema no se trabajó a detalle con los procesos de software, ni se elaboró un sistema informático que soporte su administración.

En ese sentido, los problemas generados durante la gestión de requerimientos de software en esta Unidad son similares a los encontrados en los estudios citados anteriormente, tales como: inadecuada definición de objetivos y alcances; inexactitud en la estimación de esfuerzo, tiempo y complejidad; requerimientos omitidos, duplicados y/o mal identificados; documentación de requerimientos incompleta, no estandarizada o inexistente; desconocimiento del estado y avance del requerimiento; falta de control de cambios; retrasos en la entrega del producto software, entre otros.

A nivel mundial se han realizado grandes esfuerzos por desarrollar herramientas y metodologías que apoyen el trabajo de los administradores, analistas, desarrolladores y testers durante el ciclo de vida de desarrollo del software. Sin embargo, de la revisión bibliográfica realizada no se encontró un estudio enfocado en el análisis y adaptación de las buenas prácticas de la Ingeniería de Requerimientos a un Gobierno Autónomo Descentralizado Municipal, por ello el interés de centrarse en esta disciplina de estudio y desarrollar una aplicación informática que permita gestionar los requerimientos de software en esta Unidad.

1.2 Planteamiento del Problema

El presente trabajo investigativo parte de un análisis realizado a la Unidad de Desarrollo de Software de la Dirección de TIC del GAD-I, identificando como problema central a la inadecuada gestión de requerimientos de desarrollo de software. Las causas de este problema radican en fallos en la elicitación y análisis de requerimientos, fallos en la especificación y documentación de requerimientos, fallos en el control y seguimiento de requerimientos, fallos en la verificación y validación de requerimientos, tal como se describe en el **Anexo 1**.

Los fallos en la elicitación y análisis de requerimientos se deben a la ausencia y/o inadecuado involucramiento del usuario, requerimientos inestables y cambiantes (volatilidad), escasa negociación de compromisos al inicio del proyecto, intangibilidad del software y asignación de tiempo insuficiente a la etapa de análisis. Esto genera una comprensión errónea del problema; una incorrecta definición de objetivos y alcances; requerimientos omitidos, duplicados y/o mal identificados; así como conflictos entre requerimientos.

Lo expuesto en el punto anterior, implica que exista un grado de inexactitud para la estimación de tiempo, esfuerzo y complejidad. Por lo general, estos valores no son re estimados si los requerimientos cambian, lo que desencadena en un trabajo imprevisto. Los usuarios no entienden el impacto potencial en el software, causado por un aparente cambio superficial. Los fallos en la especificación y documentación de requerimientos inciden en la existencia de documentos de requerimientos no legibles, no procesables, incorrectos, incompletos, dispersos, no estandarizados, o no existentes.

Además, se ha podido identificar el uso de herramientas no adecuadas para la gestión de requerimientos, tales como: hojas de cálculo, correos electrónicos y documentos de texto, que sumados a la carencia de control de cambios, ubicación no centralizada y crecimiento incontrolado de requerimientos, implican fallos en el control y seguimiento de los mismos. Por consiguiente, esto genera el desconocimiento del estado y avance del proyecto, difícil control y seguimiento de las actividades, indisponibilidad de información para la toma de decisiones, retraso en la entrega del pedido de software, y en consecuencia, un aumento en el uso de recursos durante su desarrollo.

Finalmente, la falta de requerimientos aprobados por los interesados y la falta de procedimientos para realizar casos de prueba, hacen que sea muy complicado verificar y validar los requerimientos, detectar o prevenir defectos y demostrar que el software cumple con lo solicitado por el usuario.

Por todo lo expuesto, resulta necesario realizar una investigación para dar una solución a los problemas relacionados con la gestión de requerimientos de desarrollo y mantenimiento de software en esta Unidad.

- **Delimitación Espacial:** Unidad de Desarrollo de Software de la Dirección de Tecnologías de la Información y Comunicación del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra.
- **Delimitación Temporal:** El trabajo se realizará durante los años 2016 y 2017.

1.3 Formulación del Problema

¿La implementación de una guía metodológica de Ingeniería de Requerimientos incide en la mejora de la gestión de requerimientos de software de la Dirección de TIC del GAD-I?

Preguntas de investigación

- ¿El establecimiento de una guía metodológica de Ingeniería de Requerimientos permitirá mejorar la gestión del desarrollo y mantenimiento de software en la Dirección de TIC del GAD-I?
- ¿Cuál es el proceso de gestión de requerimientos de software adecuado para la Dirección de TIC del GAD-I?
- ¿Cuáles son las características funcionales que debe poseer una herramienta de gestión de requerimientos de software para la Dirección de TIC del GAD-I?
- ¿Cómo debe ser una aplicación informática que permita realizar la gestión de requerimientos de desarrollo y mantenimiento de software en la Dirección de TIC del GAD-I?

1.4 Justificación de la Investigación

Las múltiples competencias designadas a los Gobiernos Autónomos Descentralizados (GAD), por el Código Orgánico de Organización Territorial Autonomía y Descentralización (COOTAD), implican que dentro de la estructura orgánica de los GAD exista una Dirección encargada de la entrega de servicios de Tecnologías de la Información y Comunicación; con el

fin de dar soporte a todas las áreas administrativas que, de forma directa o indirecta, prestan servicios públicos a la ciudadanía.

La Dirección de Tecnologías de la Información y Comunicación TIC del GAD-I, a través de la Unidad de Desarrollo de Software, implementa sistemas informáticos a ser usados por toda la institución, como es el caso de los sistemas de Recaudación, Avalúos y Catastros, Actividades Económicas, Información Geográfica, Participación Ciudadana, entre otros. La administración de las peticiones relacionadas a estos sistemas y muchos otros más, se ha vuelto complicada, debido a los problemas identificados en esta área para la gestión de requerimientos de software.

La importancia de poder atender adecuadamente los requerimientos de software atraviesa el ámbito funcional y legal. Esto conlleva a que los profesionales de TI que trabajan en la institución tengan interés en estudiar el proceso de desarrollo y mantenimiento de software, con el objetivo no sólo de mejorarlo, sino también de generar productos y servicios de calidad. La aplicación de la Ingeniería de Requerimientos, según Berenbach et al. (2009), es muy esencial en las instituciones que deben cumplir con regulaciones gubernamentales, donde los requerimientos deben ser cuidadosamente documentados, rastreados, revisados y probados.

Por esta razón, se plantea la elaboración de una guía metodológica y una aplicación informática que ayuden a la Dirección de TIC del GAD-I a mejorar su nivel de gestión de requerimientos de software; para que funcionarios públicos y ciudadanos se vean beneficiados con los resultados.

1.5 Objetivos de la Investigación

1.5.1 Objetivo General

Implementar una guía metodológica de Ingeniería de Requerimientos, mediante la definición de un proceso y elaboración de una aplicación informática, para mejorar la gestión de requerimientos de software de la Dirección de TIC del GAD-I.

1.5.2 Objetivos Específicos

- Determinar un proceso de gestión de requerimientos de software para la Dirección de TIC, mediante el análisis y aplicación de las buenas prácticas de la Ingeniería de Requerimientos.
- Determinar las características funcionales de una aplicación de gestión de requerimientos de software para la Dirección de TIC, mediante el análisis de herramientas existentes en el mercado.
- Elaborar una aplicación informática para la gestión de requerimientos de software de la Dirección de TIC, mediante la sistematización del proceso definido y la implementación de características funcionales seleccionadas de las herramientas analizadas.

2. CAPÍTULO II. MARCO REFERENCIAL

2.1 La Ingeniería de Software

2.1.1 Antecedentes

El término de Ingeniería de Software se propone por primera vez en el año 1968 durante una conferencia patrocinada por el Comité de Ciencia de la OTAN - Organización del Tratado del Atlántico Norte realizada en Alemania, en la que se intentaba dar soluciones a la denominada “Crisis del Software”, problemática que hacía referencia a la dificultad de escribir código que esté libre de defectos, que se pueda entender, mantener y verificar.

En la conferencia se discutieron los problemas que se tenían para lograr fiabilidad en los programas de computador, las dificultades para cumplir con los tiempos y con las especificaciones grandes de requerimientos, así como la formación académica que deberían tener los ingenieros que realicen estas actividades de acuerdo a lo indicado por McClure (2001). Todo esto con el objetivo de incrementar la calidad del software, evitando costos altos, tiempos desmedidos de implementación y defectos que puedan ocasionar daños a la vida humana o a la propiedad por falta de seguridad y confiabilidad.

Desde ese entonces y hasta la actualidad, se han desarrollado métodos y técnicas formales que permiten afrontar la implementación de diferentes tipos de software, como los de transporte, médicos, de telecomunicaciones, de entretenimiento, industriales, de oficina, entre otros; ya que el software sigue y seguirá siendo uno de los pilares fundamentales del crecimiento de la sociedad moderna.

2.1.2 Definición de Software

En el estándar de terminología de Ingeniería de Software del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, 2002), se define al software como el conjunto de programas, procedimientos, documentación y datos asociados que forman parte de las operaciones de un sistema de computación.

El software es un sistema lógico y no físico, y por ende tiene características que lo diferencian del hardware de computador; según Pressman (2010) dichas características se pueden resumir en las siguientes:

- El software se desarrolla o modifica con intelecto, no se manufactura.
- El software no se desgasta, pero si se deteriora debido a que sufre cambios y cuando esto ocurre es posible que se introduzcan errores.
- La mayor parte del software se construye para uso individualizado, aunque actualmente exista una tendencia a la construcción de componentes genéricos reutilizables.

En ese sentido, en la actualidad existen diferentes categorías de software, como las descritas por Foster (2014): sistemas operativos, compiladores, intérpretes, sistemas de administración de base de datos, aplicaciones de escritorio, sistemas de información, aplicaciones de negocio, sistemas de inteligencia artificial, sistemas hipermedia, sistemas para desarrollo asistido por computador, paquetes de desarrollo de software, entre otros.

2.1.3 Definición de Ingeniería de Software

La Ingeniería de Software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software; es decir, es la aplicación de la ingeniería al software (IEEE, 2002). En la actualidad, la Ingeniería de Software es una sub disciplina de la Computación, al mismo nivel de la Ingeniería en Computación, Ciencias de la Computación, Sistemas de Información y Tecnología de la Información, de acuerdo a lo establecido por la ACM - Association for Computing Machinery (ACM Inc, 2017).

Este enfoque sistemático establece el uso de procesos, métodos y técnicas para analizar, modelar, desarrollar, administrar y mantener software, con el objetivo de crear nuevos sistemas, mejorar su funcionamiento, corregir defectos existentes, adaptarlo a nuevas necesidades del entorno, así como ampliar sus funcionalidades y características.

2.1.4 El Proceso del Software

Un proceso es el conjunto de actividades que transforman entradas en salidas para cumplir un objetivo específico. En el campo de la Ingeniería de Software, las entradas son las necesidades del usuario, que a través de un grupo de actividades de análisis, diseño, desarrollo, pruebas e implantación, son transformadas en productos de software que cumplen metas y objetivos determinados.

Según Pressman (2010), un proceso de software debe ser ágil y adaptable al problema, al proyecto, al equipo y a la cultura organizacional de una empresa; por ello un proceso adoptado para un proyecto puede ser distinto al adoptado para otro proyecto. Sin embargo existen fases comunes para el ciclo de vida del software, tales como las descritas en (Bourque & Fairley, 2014) que se pueden observar en la **Tabla 2**.

Tabla 2: Fases del ciclo de vida del software

Fuente: Elaborado con contenido de "Guide to the Software Engineering Body of Knowledge – SWEBOK" por Bourque & Fairley, 2014.

FASE	DESCRIPCIÓN
Especificación de Requerimientos	Actividades de elicitación, análisis, especificación, y validación de los requerimientos de software, así como también la administración de dichos requerimientos.
Diseño de Software	Actividades de definición de la arquitectura, componentes, interfaces y otras características del sistema, en base a los requerimientos de software.
Construcción de Software	Actividades de creación de software funcional mediante la codificación, verificación, pruebas unitarias y pruebas de integración.
Pruebas de Software	Actividades de verificación dinámica de que un programa provee el comportamiento esperado, mediante la aplicación de casos de prueba.
Mantenimiento de Software	Actividades que se realizan luego de la implementación de un sistema; y que tienen como objetivo modificarlo para corregir defectos, adaptarlo a nuevos requerimientos de usuario o a cambios en los ambientes operativos.

2.1.5 Modelos del Ciclo de Vida de Desarrollo del Software

Un modelo de ciclo de vida establece el flujo para el proceso de desarrollo del software, es decir el orden y la secuencia de las fases a ejecutar; el flujo del proceso puede ser lineal, evolutivo o iterativo. Existen varios modelos de ciclos de vida de desarrollo de software, tales como: Cascada, Desarrollo evolutivo, Iterativo, Prototipo rápido, Transformación formal, Basado en componentes y Desarrollo ágil, tal como se observa en la **Tabla 3**.

Tabla 3: Características de modelos de ciclo de vida de desarrollo de software

Fuente: Elaborado con contenido de “Software Engineering: A Methodical Approach” por E. Foster, 2014.

MODELO	CARACTERÍSTICAS
Cascada	<ul style="list-style-type: none"> - El conocimiento total de los requerimientos de un sistema debe ser obtenido antes de su desarrollo. - Cada fase es aceptada por los usuarios antes de avanzar con la siguiente fase. No es posible regresar a fases anteriores. - El desarrollo del sistema puede tomar mucho tiempo. - Permite obtener un sistema comprensible, documentado, funcional y bien integrado. - Sin embargo, los requerimientos pueden cambiar antes de que el sistema se finalice.
Desarrollo Evolutivo	<ul style="list-style-type: none"> - Permite el desarrollo de una implementación inicial del sistema, revisarla con el usuario y refinarla las veces que sean necesarias hasta su finalización. - Permite obtener retroalimentación del usuario en cada refinamiento. - La especificación de requerimientos puede desarrollarse en forma creciente. - El usuario obtiene una idea del sistema antes de su finalización.
Iterativo	<ul style="list-style-type: none"> - El ciclo de vida se compone de varias iteraciones. - Cada iteración es un mini proyecto que tiene todas las fases de un ciclo de vida. - Las iteraciones pueden ser en serie o en paralelo, pero todas son integradas en la versión final del producto software.
Desarrollo Ágil	<ul style="list-style-type: none"> - Pone mayor énfasis en los resultados de la Ingeniería de Software (construcción y entrega), más que en el proceso de obtener esos resultados (análisis, diseño, desarrollo, implementación y administración). - Los requerimientos del software pueden ir evolucionando en el tiempo de acuerdo a la necesidad. - Se planifica la entrega del software por partes funcionales, de acuerdo a la priorización del usuario. - Según el (Manifiesto Ágil, 2001), se prefiere el software funcionando sobre la documentación extensiva, los individuos e interacciones sobre procesos y herramientas, la colaboración con el cliente sobre negociación contractual y la respuesta ante el cambio sobre seguir un plan.

2.1.6 Estándares Internacionales de Ingeniería de Software

Un estándar internacional es una especificación técnica de un proceso en un ámbito determinado, que ha sido aprobada por una organización o comité de estándares; constituye el resultado de la investigación y documentación de buenas prácticas de diferentes profesionales a nivel mundial.

En la actualidad, existen organizaciones como ISO (Organización Internacional de Normalización), IEC (Comisión Electrotécnica Internacional) e IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) que trabajan conjuntamente en la publicación de estándares internacionales para la normalización de procesos relacionados con la Ingeniería de Software.

En la **Tabla 4** se detallan algunos estándares específicos para la gestión de requerimientos, diseño, construcción, pruebas y mantenimiento de software.

Tabla 4: Estándares Internacionales de Ingeniería de Software

Fuente: Elaborado por la autora

ACTIVIDAD	ESTÁNDAR	DESCRIPCIÓN
Gestión de Software	IEEE 828-2012	Establece los requisitos mínimos para los procesos de Gestión de Configuración en sistemas e ingeniería de software.
	ISO/IEC/IEEE 16326-2009	Provee normativas para especificar planes de administración de proyectos, cubriendo proyectos de software y proyectos de sistemas de software intensivo.
	ISO/IEC/IEEE 12207-2017	Establece un proceso común para describir el ciclo de vida completo de los productos de software desde su concepción hasta su retiro. Se aplica a la adquisición o desarrollo de productos y servicios de software, ya sea que se realicen interna o externamente a la organización.
Requerimientos de Software	ISO/IEC/IEEE 29148-2011	Estándar para la especificación de requerimientos de software.
	IEEE 1012-2016	Establece la normativa para los procesos de verificación y validación de software y hardware, con el objetivo de determinar si los productos cumplen con los requisitos de la actividad de desarrollo y si el producto satisface las necesidades del usuario.

Continuación de la **Tabla 4.**

ACTIVIDAD	ESTÁNDAR	DESCRIPCIÓN
Diseño de Software	IEEE 1016-2009	Establece el contenido y la organización del documento de descripción de diseño de software.
	ISO/IEC/IEEE 42010:2011	Contiene estándares para la creación, el análisis y el mantenimiento de arquitecturas de sistemas a través del uso de descripciones de arquitectura. Para ello establece un modelo conceptual de arquitectura, así como los contenidos requeridos para su descripción.
Construcción de Software	IEEE 1008-1987	Establece un enfoque integrado para la ejecución sistemática y documentada de pruebas unitarias de software.
	IEEE 1012-2016	Establece la normativa para los procesos de verificación y validación de software y hardware, con el objetivo de determinar si los productos cumplen con los requisitos de la actividad de desarrollo y si el producto satisface las necesidades del usuario.
	IEEE 1016-2009	Establece el contenido y la organización del documento de descripción de diseño de software.
Pruebas de Software	IEEE 829-2008	Establece una normativa para la documentación de pruebas de software, así como la amplitud y profundidad de cada una de ellas.
	IEEE 1012-2016	Establece la normativa para los procesos de verificación y validación de software y hardware, con el objetivo de determinar si los productos cumplen con los requisitos de la actividad de desarrollo y si el producto satisface las necesidades del usuario.
Mantenimiento de Software	ISO/IEC/IEEE 14764	Proporciona una guía para la planificación, ejecución, control, revisión, evaluación y cierre del Proceso de Mantenimiento de software.
	IEEE 12207-2017	Establece un proceso común para describir el ciclo de vida completo de los productos de software desde su concepción hasta su retiro. Se aplica a la adquisición o desarrollo de productos y servicios de software, ya sea que se realicen interna o externamente a la organización.

Continuación de la **Tabla 4**.

ACTIVIDAD	ESTÁNDAR	DESCRIPCIÓN
Aseguramiento de la Calidad	IEEE 730-2014	Establece un conjunto de requerimientos para iniciar, planear, controlar y ejecutar el proceso de aseguramiento de la calidad en proyectos de desarrollo y mantenimiento de software.
	IEEE 1012-2016	Establece la normativa para los procesos de verificación y validación de software y hardware, con el objetivo de determinar si los productos cumplen con los requisitos de la actividad de desarrollo y si el producto satisface las necesidades del usuario.
	IEEE 1028-2008	Define tipos de revisiones y auditorías del software, así como los procedimientos para cada uno de ellos.
	IEEE 1061-1998	Define una metodología para establecer requerimientos de calidad, así como para identificar, implementar, analizar y validar métricas de calidad del proceso y del producto de software.

2.2 La Ingeniería de Requerimientos de Software

2.2.1 Definición de Requerimiento de Software

El estándar de terminología de Ingeniería de Software del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE, 2002) provee las siguientes tres definiciones para la palabra “requerimiento”:

- (1) Condición o capacidad requerida por un usuario para resolver un problema o para lograr un objetivo.
- (2) Condición o capacidad que debe cumplir o poseer un sistema o un componente de un sistema para satisfacer un contrato, estándar, especificación u otro documento formal.
- (3) Una representación documentada de una condición o capacidad como en (1) o (2).

En ese sentido, Aurum y Wohlin (2005) establecen que los requerimientos no son solamente necesidades de usuario, si no también necesidades de la organización, del gobierno y del entorno. Por lo tanto, un requerimiento de software es un atributo, capacidad o característica requerida en un sistema, con el fin de dar utilidad a todos los interesados del mismo.

2.2.2 Definición de Ingeniería de Requerimientos

La Ingeniería de Requerimientos está dedicada a entender y trabajar en la declaración del problema y no en la solución. Sin embargo, los administradores de proyectos suelen pensar que tienen un progreso real sólo cuando el equipo de desarrollo empieza a trabajar en la solución del problema, es decir en la construcción del sistema.

Comúnmente, la fase de levantamiento de requerimientos es mal entendida como el conjunto de actividades que permiten documentar y especificar requerimientos, olvidando la importancia de la toma de decisiones; como resultado se tiene soluciones con requerimientos omitidos, incompletos e inconsistentes. A continuación se listan algunas causas que pueden ocasionar problemas durante la identificación de requerimientos, citados por Young (2004):

- El proyecto se está empezando a organizar y las cosas son confusas.
- El equipo de trabajo no tiene dominio del negocio del sistema.
- Existe un acceso limitado a los expertos en la materia.
- Los cambios en los requerimientos son aceptados y realizados sin tomar en cuenta impactos de costo, tiempo y calidad.
- No se tiene un proceso definido para gestionar requerimientos de software.
- No existe un mecanismo para mantener la comunicación entre las partes involucradas de un proyecto.
- El usuario no es incluido como socio a lo largo del ciclo de vida del proyecto.
- No se utilizan métodos, técnicas y herramientas conocidas, familiares o probadas.
- No existe un repositorio unificado de requerimientos; los requerimientos se encuentran en numerosas fuentes, con antigüedad y validez variable.

Hull, Jackson y Dick (2011) definen a la Ingeniería de Requerimientos como una rama de la Ingeniería de Software que se encarga de descubrir, analizar, desarrollar, rastrear, validar y gestionar los requerimientos que definen un sistema. En ese sentido, esta rama tiene un enfoque formal y su aplicación es muy importante en los dominios que son regulados por normativas o estándares, donde cada requerimiento debe ser cuidadosamente documentado, rastreado, revisado y probado.

Sin embargo, a medida que se van definiendo más y más requerimientos, los miembros del equipo de trabajo suelen tener dificultad para manejarlos, debido a que los requerimientos son añadidos y cambiados durante todo el ciclo de vida de desarrollo del sistema. Berenbach et al. (2009), definen algunos factores clave de éxito para la aplicación de la Ingeniería de Requerimientos, tales como:

- Definición de un proceso para gestionar requerimientos de software.
- Definición de un proceso para el control de cambios en requerimientos.
- Identificación e involucramiento de todos los stakeholders del proyecto, especialmente de los expertos en el dominio del negocio.
- Entrenamiento y capacitación adecuada para los analistas de requerimientos.
- Disponibilidad a tiempo completo de todos los miembros del equipo de trabajo.
- Utilización de una herramienta de Ingeniería de Requerimientos para aumentar la calidad y productividad.

Chemuturi (2013) establece que la gestión de requerimientos es el proceso de analizar, acordar, documentar, rastrear, priorizar requerimientos para controlarlos y comunicarlos a los stakeholders. Según el autor, en el contexto de la Ingeniería de Requerimientos, la gestión viene dada por lo mostrado en la **Figura 1**.

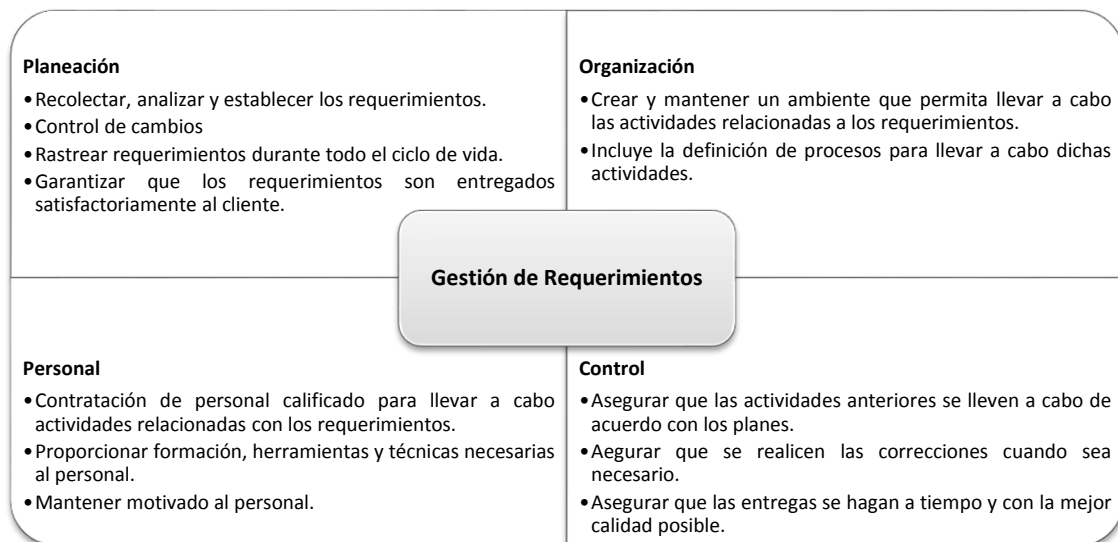


Figura 1: Gestión de requerimientos de software

Fuente: Elaborado con contenido de “Requirements Engineering and Management for Software Development” por M. Chemuturi, 2013.

2.2.3 Tipos de requerimientos de software

Los requerimientos de software pueden describir el comportamiento, un atributo o propiedad del sistema, e incluso alguna restricción que se debe considerar durante el proceso de desarrollo. Wiegers y Beatty (2013) señalan que es importante distinguir todos los tipos de información relacionada a los requerimientos de software, considerando la vista del usuario (comportamiento externo del sistema) y la vista del desarrollador (características internas), tal como se observa en la **Tabla 5**.

Tabla 5: Tipos de información de requerimientos de software

Fuente: Elaborado con contenido de “Software Requirements” por Wiegers & Beatty, 2013.

TÉRMINO	DESCRIPCIÓN
Requerimientos del Negocio	Objetivos de alto nivel del negocio, que la organización espera lograr con la implementación del sistema.
Reglas del Negocio	Política, guía, estándar, o regulación gubernamental que define o restringe algún aspecto del negocio. No son un tipo de requerimiento ya que existen fuera de la existencia de cualquier software.
Restricciones	Imposiciones sobre el diseño y construcción de un sistema, que deben ser respetadas por el desarrollador.
Requerimientos de Interface Externa	Descripción de una conexión entre un sistema de software y un usuario, otro sistema, o un dispositivo de hardware.
Características	Una o más capacidades del sistema relacionadas lógicamente, que proporcionan valor a un usuario y que se describen mediante un conjunto de requerimientos funcionales.

Continuación de la **Tabla 5**.

TÉRMINO	DESCRIPCIÓN
Requerimientos Funcionales	Descripción del comportamiento observable que un sistema debe tener bajo condiciones específicas. Describen lo que los desarrolladores deben implementar para cumplir con los requerimientos de usuario.
Requerimientos No Funcionales	Descripción de una propiedad o característica que un sistema debe exhibir o una restricción que debe respetar.
Atributos de Calidad	Un tipo de requerimiento no funcional que describe una característica o propiedad de un sistema, tales como la disponibilidad, usabilidad, seguridad, rendimiento, portabilidad.
Requerimientos del Sistema	Un requerimiento de alto nivel de un producto que contiene múltiples subsistemas, que pueden estar formados de software y hardware.
Requerimientos de Usuario	Describen metas o tareas que el usuario debe ser capaz de realizar con el sistema, incluyen la descripción de atributos o características del sistema que son importantes para la satisfacción del usuario.

Sin embargo, esta información no presenta de forma clara la clasificación y jerarquía de los requerimientos de software existentes. En ese sentido, podemos tomar la clasificación y definición realizada por el Instituto Internacional para Análisis del Negocio (IIBA, 2009) en el que se distinguen los siguientes tipos de requerimientos:

- **De Negocio:** Declaraciones de alto nivel de las metas, objetivos o necesidades que la organización espera lograr con la implementación del sistema. Permiten capturar la visión de la organización para la definición del alcance del sistema, estimación de costo y tiempo requeridos para su implementación.
- **De Stakeholders:** Declaraciones de las necesidades de un stakeholder o tipo de stakeholder. Describen la manera en que un stakeholder va a interactuar con la solución.
- **De Solución:** Declaraciones de las características que una solución debe poseer para cumplir los requerimientos de negocio y de los stakeholders. Se dividen en requerimientos Funcionales y No Funcionales. De acuerdo con Sommerville (2005), los requerimientos funcionales son declaraciones de los servicios que el sistema debe ser capaz de proveer; y los requerimientos no funcionales son restricciones en los servicios ofrecidos por el sistema, tales como restricciones de tiempo, de procesos o de estándares.

En la **Figura 2** se puede observar la relación entre los diferentes tipos de información de requerimientos. Las líneas sólidas indican que ese tipo de información es almacenada en un documento específico, por ejemplo en el documento de visión y alcance; y las líneas punteadas indican que ese tipo de información influye o es el origen de un cierto tipo de requerimiento, por ejemplo requerimientos funcionales.

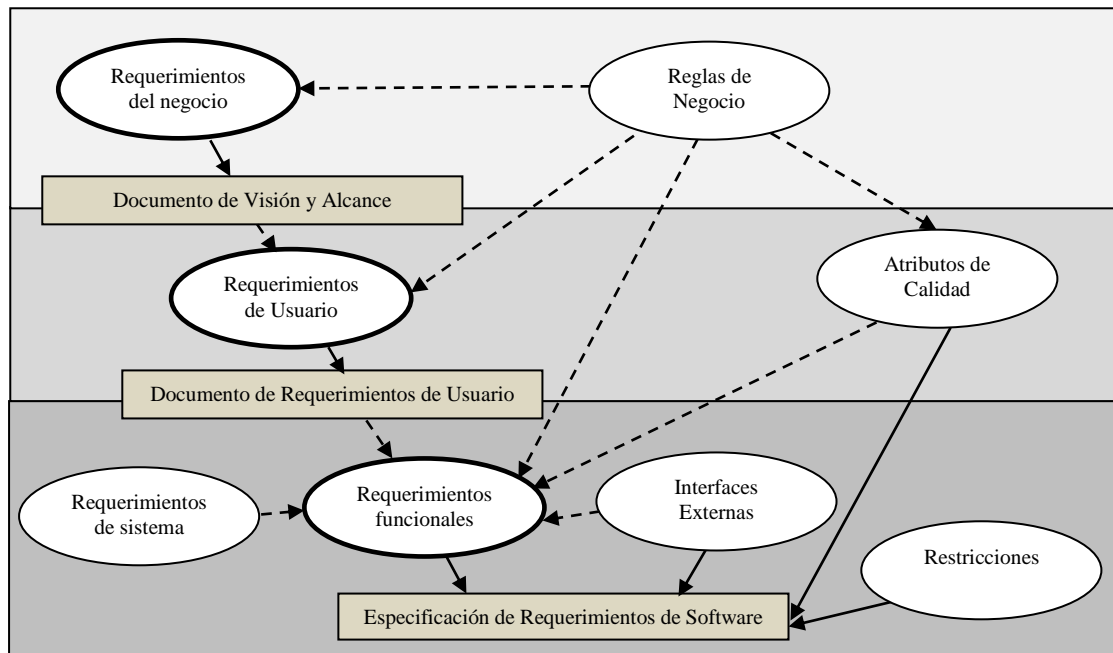


Figura 2: Relaciones entre los diferentes tipos de información de requerimientos
Fuente: “Software Requirements” por Wieggers & Beatty, 2013.

2.2.4 Características de un buen requerimiento de software

Según Berenbach et al. (2009), durante el desarrollo de un software, los profesionales se apresuran a iniciar con la etapa de codificación, dejando a un lado la validación de las características que debe cumplir un requerimiento. Pevio a su implementación, es importante determinar si el requerimiento cumple con las características mostradas en la **Tabla 6** a fin de garantizar una mayor calidad en el producto y en el proceso de desarrollo.

Tabla 6: Características de un buen requerimiento de software

Fuente: Elaborado con contenido de “Software & Systems Requirements Engineering: In practice” por B. Berenbach, 2009.

CARACTERÍSTICA	DESCRIPCIÓN
Factible	Su implementación es factible dentro de las restricciones del proyecto (recursos, presupuesto, habilidades, cronograma, tecnología)
Necesario	Si el sistema puede satisfacer las necesidades reales sin el requerimiento, entonces no es necesario.
Válido	Es un requerimiento que el cliente necesita y que el sistema debe cumplir.
No ambiguo	El requerimiento tiene una única interpretación.
Verificable	La implementación del requerimiento en el sistema, es decir el código que provee esa capacidad, puede ser probada.

Continuación de la **Tabla 6**

CARACTERÍSTICA	DESCRIPCIÓN
Consistente	Si el requerimiento no se contradice o no entra en conflicto con alguna ley, normativa u otro requerimiento.
Completo	Si contiene toda la información necesaria para ser implementado. Todas las condiciones bajo las cuales se aplica el requerimiento deben ser declaradas.
Conciso	El requerimiento debe ser declarado de forma clara y simple.
Rastreadable	La trazabilidad es la habilidad de poder describir y seguir la vida de un requerimiento desde su origen, especificación, desarrollo y uso. Un requerimiento es trazable si pueden ser identificados su fuente, los componentes que lo implementan, y los casos de prueba que verifiquen que en verdad se haya implementado.

2.2.5 Atributos de requerimientos de software

Un atributo de un requerimiento de software es una propiedad o característica, que sirve para dar soporte a la gestión del requerimiento durante todo su ciclo de vida. De acuerdo con el Instituto Internacional para Análisis de Negocio (IIBA, 2009) los atributos permiten asociar requerimientos entre sí; facilitando el seguimiento, la trazabilidad, el análisis de riesgos y el análisis de impactos causados por los cambios propuestos.

La Junta de Andalucía de España definió un Marco de Desarrollo de Software (2016), en el que se diferencian los atributos generales de los requerimientos y aquellos relacionados con su gestión durante el ciclo de vida, tal como se describe a continuación:

- **Atributos generales:** Son atributos propios del requerimiento y que se pueden identificar durante el trabajo de elicitación. Por ejemplo: identificador único, nombre descriptivo, autor, versión, complejidad, importancia.
- **Atributos relacionados con la gestión:** Son atributos que el requerimiento va adquiriendo a medida que va pasando por las etapas del ciclo de vida de desarrollo. No son propiedades del requerimiento y sólo tienen sentido en el contexto del proyecto. Por ejemplo: estado, prioridad, costo estimado, componente del sistema al que es asignado, trazabilidad.

2.2.6 El Analista de Requerimientos de software

El Analista de Requerimientos de software tiene la responsabilidad de elicitar, analizar, especificar, verificar y administrar las necesidades de todos los stakeholders. De acuerdo con Young (2004) los Analistas de Requerimientos deben tener habilidades de análisis, comunicación, escritura y organización, tal como se describe a continuación:

- Habilidades para dirigir talleres de elicitación de requerimientos.
- Habilidades para entrevistar e interactuar con los stakeholders acerca de sus necesidades.

- Habilidades para escuchar y entender lo que las personas dicen o intentan decir.
- Habilidades analíticas para evaluar críticamente la información obtenida de diferentes fuentes.
- Habilidades para ayudar a negociar prioridades y resolver conflictos entre los stakeholders.
- Habilidades de organización para trabajar con la gran cantidad de información obtenida durante la elicitación y análisis.
- Habilidades de modelado para representar la información de requerimientos en formas gráficas y representaciones textuales en lenguaje natural.
- Habilidades para escribir especificaciones de requerimientos completas y no ambiguas.

2.2.7 Definición de Stakeholder

Un stakeholder es cualquier persona u organización que tenga un interés en el proyecto y que se vea afectado por el resultado positivo o negativo del mismo, se le conoce también como interesado o actor. Los stakeholders pueden ser:

- **Cientes** que pagan por el sistema.
- **Usuarios** que utilizan el sistema.
- **Consultores** que prestan un bien o servicio a la entidad.
- **Patrocinadores** que financian todo o parte del proyecto.
- **Miembros del equipo de desarrollo:** líder de proyecto, analistas, programadores, testers.

En la **Figura 3** se muestra un diagrama de cebolla con los stakeholders externos o internos a la organización, que pueden ser parte de un proyecto de desarrollo de software y que se ven afectados por el mismo.

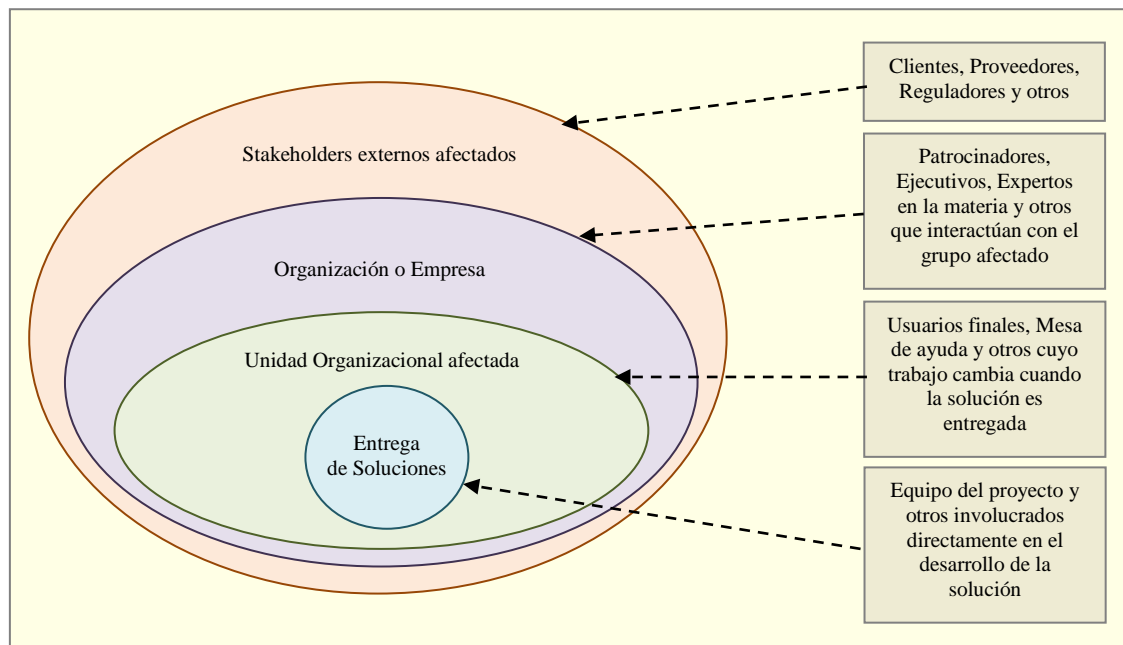


Figura 3: Diagrama de cebolla de stakeholders

Fuente: "A Guide to the Business Analysis Body of Knowledge" por Instituto Internacional para Análisis del Negocio, 2009.

2.2.8 Especificación de Requerimientos de Software

Una Especificación de Requerimientos de Software (SRS - Software Requirements Specification) es una descripción completa del sistema que se va a desarrollar, y constituye una base de los acuerdos entre stakeholders y miembros del equipo de desarrollo para la implementación del sistema. Por lo general, una SRS contiene: objetivos y alcances; definiciones y abreviaturas; requerimientos funcionales; requerimientos no funcionales como atributos de calidad, interfaces externas y restricciones.

El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE - Institute of Electrical and Electronics Engineers) desarrolló un estándar para la especificación de requerimientos de software denominado IEEE 830-1998 – Recommended Practice for Software Requirements Specifications. Posteriormente, el estándar fue reemplazado por ISO/IEC/IEEE 29148-2011 - Systems and software engineering - Life cycle processes - Requirements Engineering, tal como se muestra en la **Figura 4**.

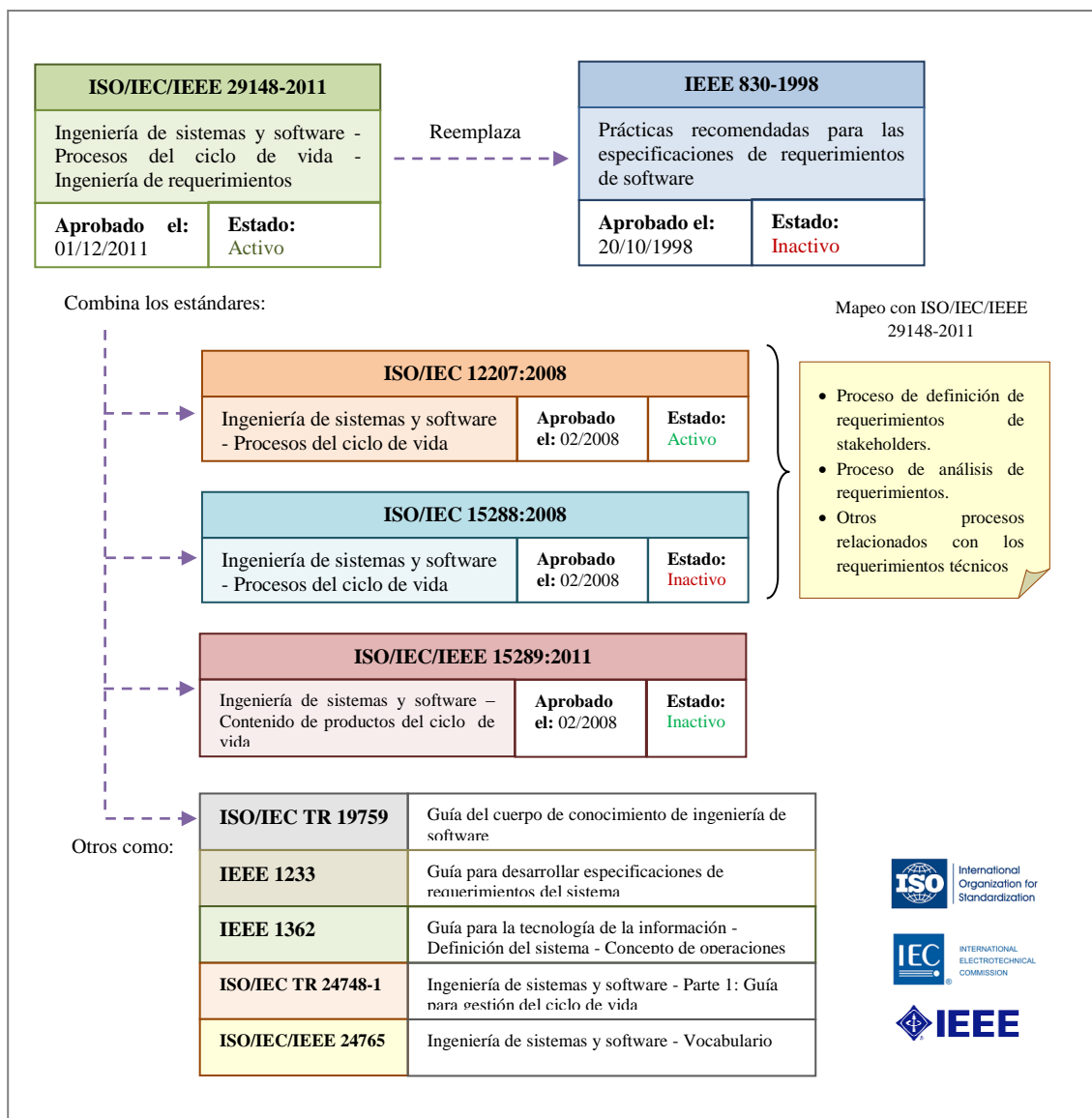


Figura 4: Vista general del estándar ISO/IEC/IEEE 29148-2011

Fuente: Elaborado por la autora con contenido de ISO/IEC/IEEE 29148-2011

Para que una especificación de requerimientos sea factible, válida, no ambigua, verificable, consistente, completa y trazable cada uno de sus requerimientos debe cumplir con dichas características. Además, una especificación de requerimientos debe ser modificable, es decir su estructura debe tener una adecuada organización sin redundancia y permitir que cualquier cambio a un requerimiento pueda realizarse fácilmente.

2.2.9 Taxonomía de Ingeniería de Requerimientos

Una taxonomía de Ingeniería de Requerimientos es una colección de tipos de requerimientos organizados en una estructura de clasificación jerárquica, y para que esté bien formada cada uno de los términos debe tener un solo padre. Según Berenbach et al. (2009), una taxonomía de Ingeniería de Requerimientos debe tener los siguientes atributos:

- **Completa:** Se deben incluir todos los tipos de requerimientos que serán utilizados por la organización o proyecto.
- **Navegable:** Debe ser fácil de navegar desde un elemento a otro.
- **Válida:** La taxonomía debe ser validada contra fuentes de libros y expertos.

2.2.10 Actividades de la Ingeniería de Requerimientos

Anteriormente se había dicho que la Ingeniería de Requerimientos involucra todas las actividades dedicadas a la identificación, análisis, documentación y validación de requerimientos frente a las necesidades de usuario, así como el proceso que las soporta. En ese sentido, se tomará la clasificación de actividades de Wiegers y Beatty (2013), en la que se divide a la Ingeniería de Requerimientos en dos sub disciplinas: Desarrollo de Requerimientos y Administración de Requerimientos.

Dentro del Desarrollo de Requerimientos, los autores establecen la ejecución de cuatro actividades: Elicitación, Análisis, Especificación y Validación. Por el contrario, en la Administración de Requerimientos se establecen actividades que se deben ejecutar durante todo el proyecto, tales como: Control de cambios, Control de versiones, Seguimiento a requerimientos y Trazabilidad de requerimientos.

Independientemente del ciclo de vida de desarrollo que se utilice (cascada, iterativo, incremental o híbrido), las actividades enunciadas anteriormente se deben realizar con diferentes niveles de detalle a lo largo del proyecto. En la **Tabla 7** se exponen de forma general las actividades y tareas a efectuar dentro de las sub disciplinas de la Ingeniería de Requerimientos establecidas por Wiegers y Beatty (2013).

Tabla 7: Descripción de actividades de la Ingeniería de Requerimientos

Fuente: Elaborado por la autora

Sub Disciplina:	Desarrollo de Requerimientos
Actividad	Descripción
Elicitación	Conocimiento del entorno y descubrimiento de los requerimientos.
Análisis	Obtener un entendimiento más claro y preciso de los requerimientos.
Especificación	Representar y almacenar los requerimientos en una forma organizada y entendible.
Validación	Confirmar que se han obtenido los requerimientos correctos y que están declarados correctamente.

Continuación de la **Tabla 7**

Sub Disciplina:	Administración de Requerimientos
Actividad	Descripción
Control de Cambios	Controlar los cambios en requerimientos y evaluar el impacto de dichos cambios en el proyecto.
Control de versiones	Establecer líneas base de requerimientos y controlar sus versiones.
Trazabilidad de requerimientos	Mantener relaciones entre requerimientos, que permitan rastrearlos desde su origen hasta su implementación.
Seguimiento a requerimientos	Realizar el seguimiento del estado y avance de los requerimientos.

2.3 Desarrollo de Requerimientos

Antes de iniciar con las actividades de elicitación, análisis, especificación y validación de requerimientos de software, es necesario determinar el alcance del proyecto. Robertson y Robertson (2013) recomiendan iniciar con reuniones de despegue, que tienen como propósito principal la identificación de la funcionalidad que va a ser incluida y la que va a ser excluida del software por desarrollar.

En las reuniones de despegue deben participar los principales interesados del proyecto, con el objetivo de aclarar las metas, beneficios, riesgos, así como también llegar a un consenso sobre el alcance del trabajo a ser realizado. Este análisis permite elaborar una estimación preliminar de costos, tiempo y factibilidad, con el fin de tomar la decisión de continuar o no con el proyecto.

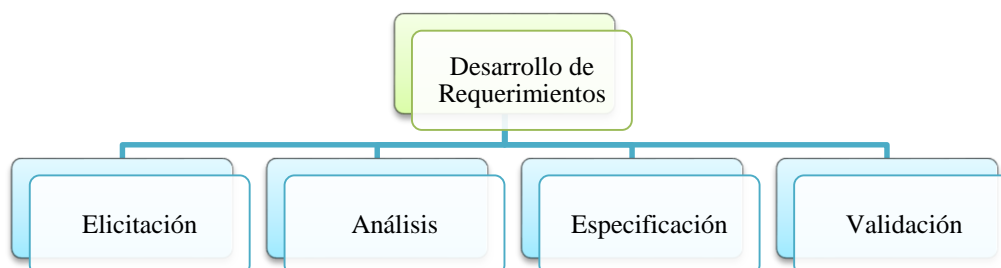


Figura 5: Actividades del desarrollo de requerimientos

Fuente: Elaborado por la autora

2.3.1 Elicitación de requerimientos

Esta actividad se compone de dos sub actividades que son la Elicitación y la Recolección de requerimientos, cuyo objetivo es obtener el conocimiento del entorno y descubrir los requerimientos que servirán como base a las demás actividades del ciclo de vida de desarrollo.

Según Chemuturi (2013), la Elicitación es la recopilación de información de individuos (fuentes primarias) que están directamente relacionados con el proyecto y que ejecutan los procesos de negocio, tales como usuarios finales y expertos; para ello se pueden usar técnicas de entrevista, cuestionarios, encuestas, observación y lluvia de ideas. Así mismo indica que la Recolección es una recopilación de información procedente de fuentes distintas a los seres humanos (fuentes secundarias) tales como documentos, aplicaciones existentes, estándares y guías.

El Analista de Requerimientos de software debe conocer todas las técnicas para la elicitación y recolección de requerimientos y ser capaz de seleccionar la más adecuada en una determinada situación. Independientemente de la técnica seleccionada, el Instituto Internacional para Análisis del Negocio (IIBA, 2009) establece que en esta actividad se deben realizar las siguientes tareas:

- **Preparar la elicitación:** Tiene como objetivo asegurar que todos los recursos necesarios estén disponibles, organizados y programados.
- **Dirigir la elicitación:** Tiene como objetivo ejecutar la elicitación y recolección de requerimientos, haciendo uso de las técnicas seleccionadas.
- **Documentar los resultados de la elicitación:** Tiene como objetivo registrar la información obtenida durante la elicitación. Los entregables de la elicitación dependen de la técnica usada, como notas de entrevistas, respuestas de encuestas, entre otros.
- **Confirmar los resultados de la elicitación:** Revisar los resultados documentados con los stakeholders para asegurar que el entendimiento del analista se ajusta a sus necesidades.

Para tener una especificación completa de requerimientos, Chemuturi (2013) indica que durante la elicitación y recolección es necesario identificar todos los procesos que están dentro del alcance del software y obtener los siguientes detalles acerca de dichos procesos:

- **Entradas:** de fuentes externas, internas o de otros procesos.
- **Salidas:** cada proceso entrega una salida.
- **Pasos del proceso:** cada proceso realiza alguna transformación de las entradas y las convierte en salidas. Cada proceso consiste en un conjunto de pasos con un evento de inicio y uno de finalización. El proceso debe incluir la verificación de las entradas para asegurar que sean completas, apropiadas y libres de errores.
- **Disparadores:** cada proceso necesita un disparador que inicie su ejecución. Puede ser un evento iniciado por una persona, otra aplicación, o por la misma aplicación.

Para ejecutar la elicitación de requerimientos, es necesario aplicar un conjunto de técnicas para la obtención adecuada de la información. Los resultados de las técnicas usadas, dependen de las habilidades, experiencia y conocimiento en el negocio del analista de requerimientos. En la **Tabla 8** se describen algunas técnicas para elicitar requerimientos, descritas por Aurum y Wohlin (2005).

2.3.2 Análisis de requerimientos

Esta actividad permite identificar si los requerimientos obtenidos durante la Elicitación y Recolección están: completos, son factibles, no ambiguos y no contradictorios entre sí. Para ello, resulta necesario volver a los stakeholders con el objetivo de aclarar y resolver todas las inconsistencias que se presenten. Algunas de las tareas que se deben realizar durante el Análisis de requerimientos son las indicadas por Chemuturi (2013) y listadas a continuación:

- Enumerar los requerimientos y verificar que cada uno esté completo.
- Evaluar la factibilidad técnica, económica y de tiempo de cada requerimiento.
- Clasificar los requerimientos y distribuirlos en componentes lógicos del sistema.
- Identificar requerimientos duplicados y contradictorios.
- Identificar interfaces del sistema.

Tabla 8: Técnicas de elicitación de requerimientos
Fuente: Elaborado por la autora

TÉCNICA	DESCRIPCIÓN
Entrevista	<p>Conversación en lenguaje natural entre el analista y los stakeholders para obtener información. Existen tres tipos:</p> <ul style="list-style-type: none"> • No estructurada: El entrevistador ejerce poco control sobre la dirección de la discusión. Se usa como técnica inicial cuando se tiene un limitado entendimiento del dominio, o como antecesora a las entrevistas formales. • Estructurada: Es conducida usando un conjunto predeterminado de preguntas que permiten obtener información específica. Su éxito depende de saber cuáles son las preguntas correctas, cuando hacerlas y a quien hacerlas. • Semi estructuradas: Son una combinación de las no estructuradas y estructuradas.
Encuesta	<p>Se aplica un conjunto de preguntas abiertas y cerradas en forma de cuestionario a múltiples stakeholders. Para que sea efectiva, los términos y conceptos deben ser entendidos por el diseñador de la encuesta y por los participantes. Sin embargo son limitadas ya que no permiten profundizar o expandir un tema.</p>
Análisis de tareas	<p>Tiene como objetivo construir una jerarquía de las tareas realizadas por los usuarios, y determinar el conocimiento usado o requerido para llevarlas a cabo.</p>
Análisis del dominio	<p>Tiene como objetivo examinar la información y los sistemas relacionados para capturar conocimiento. Va acompañada de técnicas de observación y entrevista.</p>
Lluvia de ideas	<p>Es un proceso en donde los participantes de diferentes grupos tienen una discusión informal para generar rápidamente ideas sin enfocarse en ninguna en particular.</p> <p>Se usa para desarrollar las declaraciones iniciales del sistema. Promueve el libre pensamiento y el descubrimiento de nuevas ideas para resolver problemas.</p>
Observación	<p>El analista debe observar la ejecución real de los procesos existentes realizados por los usuarios, sin intervención. Requiere que el analista tenga habilidades para interpretar y entender las acciones realizadas.</p>

- Identificar stakeholders para cada requerimiento.
- Priorizar requerimientos.
- Estimar esfuerzo y tiempo de implementación de cada requerimiento.

La priorización es una actividad en la que deben participar todos los stakeholders del proyecto, con el objetivo de tomar decisiones sobre la importancia de cada requerimiento y asegurar la implementación de los requerimientos más críticos. Según el Instituto Internacional para Análisis del Negocio (IIBA, 2009) algunos de los factores a considerar durante la priorización de requerimientos son:

- **Valor del negocio:** Implementar primero los requerimientos que den mayor valor a la organización, en base a un análisis costo-beneficio.
- **Dificultad de implementación:** Implementar primero los requerimientos que son más fáciles, con el objetivo de ganar experiencia y disminuir la introducción de riesgos en el proyecto.
- **Cumplimiento de regulaciones o políticas:** Seleccionar requerimientos que ayuden a cumplir demandas regulatorias o políticas impuestas en la organización, que tienen precedencia sobre los intereses de los stakeholders.
- **Relaciones con otros requerimientos:** Un requerimiento puede no tener un alto valor por sí solo, pero puede estar relacionado con requerimientos de alto valor.
- **Acuerdo con el stakeholder:** el stakeholder debe decir si el requerimiento tiene o no valor.
- **Urgencia:** Implementar requerimientos en base a la necesidad por alguna restricción de tiempo.

2.3.3 Especificación de requerimientos

Especificar requerimientos significa declarar y establecer, de una manera organizada, todas las características funcionales y no funcionales que debe poseer el software por desarrollar. Al elaborar una especificación de requerimientos, se cuenta con una base para continuar con las actividades del ciclo de vida de desarrollo del software: diseño, construcción y pruebas; así como también para las actividades de gestión del proyecto: análisis de riesgos, estimación de costo y tiempo.

Al redactar una especificación de requerimientos, lo más común es utilizar lenguaje natural debido a que es el lenguaje de comunicación compartido por todos los stakeholders dentro del proyecto. A pesar de ello, uno de los problemas más conocidos del lenguaje natural es la ambigüedad. Según Aurum y Wohlin (2005), una de las soluciones a la ambigüedad es el uso de un lenguaje de especificación de requerimientos formal como Software Cost Reduction – SCR, o semiformal como Unified Modeling Language - UML.

Sin embargo, Bourque y Fairley (2014) recalca que la selección de una notación formal está basada en el entrenamiento, habilidades y preferencias de los autores y lectores del documento. Por lo general, los lenguajes de especificación formal son usados en sistemas críticos o de alto riesgo. De acuerdo con Wieggers y Beatty (2013), los requerimientos de software se pueden representar de las siguientes formas:

- Lenguaje natural bien estructurado y cuidadosamente escrito.

- Modelos visuales que ilustran los procesos de transformación, estados de sistema y cambios entre ellos, relaciones de datos, flujos lógicos.
- Lenguajes formales de especificación precisa.

Independientemente de la forma seleccionada para la representación de los requerimientos, en esta actividad generalmente se elaboran dos tipos de documentos: Especificación de Requerimientos de Usuario - ERU y Especificación de Requerimientos de Software - ERS. En la **Tabla 9** se pueden observar las principales características de una ERU y de una ERS.

En el estándar internacional ISO/IEC/IEEE - 29148-2011 (ISO/IEC/IEEE, 2011) se especifican los procesos requeridos para implementar la Ingeniería de Requerimientos en el ciclo de vida de un producto software, así como los artefactos que deben ser producidos durante esos procesos. Este estándar reemplaza, entre otros, a IEEE 830-1998 e IEEE 1233-1998, que eran estándares con prácticas recomendadas para la elaboración de Especificaciones de Requerimientos de Software y Especificaciones de Requerimientos de Usuario respectivamente.

Tabla 9: Características de una ERU y una ERS
Fuente: Elaborado por la autora

Nº	ESPECIFICACIÓN DE REQUERIMIENTOS DE USUARIO	ESPECIFICACIÓN DE REQUERIMIENTOS DE SOFTWARE
1	Sirve para registrar los requerimientos de usuario.	Sirve para registrar los requerimientos funcionales, no funcionales (atributos de calidad, restricciones, interfaces externas)
2	Se redacta desde una perspectiva del usuario final.	Se redacta desde una perspectiva técnica del producto software.
3	Debe ser aprobada por el usuario final.	No se necesita aprobación del usuario, es un documento para uso interno del equipo de desarrollo.
4	Los usuarios la usan como referencia para todas las interacciones con el equipo de desarrollo y para la aceptación del producto.	El equipo de desarrollo la usa para guiar el diseño y la construcción del producto software.
5	Se usa lenguaje natural y casos de uso para la documentación de requerimientos.	Se usa lenguaje natural y otros modelos para representación de requerimientos de software.

La elaboración de las especificaciones de requerimientos de usuario y de software depende de las políticas y necesidades de la organización y/o del proyecto; en algunos casos las dos especificaciones se combinan en un solo documento, o incluso se omite la elaboración de la ERS. De acuerdo con Chemuturi (2013), en la actividad de especificación de requerimientos se deben realizar las siguientes tareas:

- **Documentación:** Registrar cada requerimiento con todos sus detalles, de una manera estructurada, en base a guías y formatos de documentación establecidos por la organización.
- **Aseguramiento de la calidad:** Verificar que el artefacto generado en la tarea anterior, cumpla con los criterios de calidad establecidos en la organización. Generalmente esto se realiza mediante las actividades de validación y verificación de requerimientos.
- **Aprobación:** Gestionar la aprobación física o digital del artefacto, por parte de los stakeholders que tienen autoridad para hacerlo.
- **Gestión de la Configuración:** Cuando un artefacto es aprobado, los cambios deben ser controlados bajo el proceso de gestión de cambio y control de versiones, para poder mantener una única versión vigente del mismo.

2.3.4 Validación y Verificación de requerimientos

Las actividades de validación y verificación de requerimientos tienen como objetivo determinar que se hayan obtenido los requerimientos correctos y además que se hayan declarado correctamente; son parte del aseguramiento de la calidad del software. En la **Tabla 10** se han tomado conceptos y ejemplos del libro “Guide to Advanced Software Testing” de Mette y Hass (2008), con el objetivo de aclarar y diferenciar las actividades de validación y verificación de requerimientos.

Tabla 10: Conceptos y ejemplos de validación y verificación de requerimientos

Fuente: Elaborado con contenido de “Guide to Advanced Software Testing” por Mette y Hass, 2008.

ACTIVIDAD	DESCRIPCIÓN	EJEMPLO
Validación	Evaluación de la exactitud del software en relación a los requerimientos y necesidades del usuario. Determinar si se está construyendo el software correcto.	Evaluar si un cálculo de descuento requerido por el usuario, ha sido diseñado y codificado en el software.
Verificación	Evaluación de si el software cumple los requerimientos especificados. Determinar si se está construyendo el software correctamente.	Evaluar si el algoritmo de descuento que ha sido implementado, realiza los cálculos correctamente.

En la **Tabla 10** se hace referencia a la evaluación del software, sin embargo durante la etapa de desarrollo de requerimientos no es posible realizar pruebas al software debido a que aún no se ha llegado al diseño y codificación.

En ese sentido, las actividades de validación y verificación de la Ingeniería de Requerimientos están orientadas a: 1) evaluar que las declaraciones de requerimientos satisfagan las necesidades de los usuarios y 2) evaluar que las declaraciones de requerimientos se han escrito correctamente cumpliendo las propiedades deseables de calidad.

En el estándar IEEE 1012-1998¹ (IEEE, 1998) se establece que las actividades de validación y verificación se deben realizar para los procesos de gestión, adquisición, provisión, desarrollo, operación y mantenimiento de software. Para el proceso de desarrollo, el estándar establece actividades de validación y verificación de: concepto, requerimientos, diseño, implementación, pruebas e instalación; en el caso de los requerimientos, el objetivo es asegurar la exactitud, precisión, consistencia, completitud y capacidad de prueba.

Para cumplir con la validación y verificación de requerimientos, se utilizan técnicas de revisión informal e inspecciones. En las revisiones informales se invita a uno o varios compañeros de trabajo para la revisión de la especificación de requerimientos, solicitando sus comentarios al respecto. En las inspecciones se realizan reuniones formales con los stakeholders del proyecto, con el objetivo de producir un reporte de los problemas detectados en los requerimientos.

En la **Tabla 11** se muestran las tareas necesarias para cumplir una inspección de requerimientos de software, según Wieggers y Beatty (2013). Una vez finalizado el proceso de validación y verificación, la especificación de requerimientos de software debe ser manejada bajo la gestión de la configuración.

Tabla 11: *Proceso de inspección de requerimientos de software*

Fuente: *Elaborado con contenido de “Software Requirements” por Wieggers y Beatty, 2013.*

TAREA	DESCRIPCIÓN
Planeación	Determinar cuándo y dónde va a ser la reunión, quién va a participar, el material que se va a revisar y el tiempo requerido.
Preparación	Previo a la reunión, entregar el material a los participantes para su revisión.
Reunión de inspección	Realizar la reunión de inspección en la que se deberá: 1) leer cada requerimiento, 2) solicitar a los participantes que indiquen los problemas y errores encontrados y 3) registrarlos.
Corrección	Resolver los problemas y corregir los errores encontrados en los requerimientos.
Verificación	Verificar que todos los problemas identificados han sido resueltos y los errores han sido corregidos.

2.4 Administración de Requerimientos

Una vez que los requerimientos han sido identificados, documentados y validados, es necesario controlar su evolución a lo largo del ciclo de vida del desarrollo de software. En ese sentido, es necesario establecer un conjunto de actividades que permitan gestionar los cambios en los requerimientos, controlar sus versiones, darles seguimiento y rastrearlos desde su origen hasta su implementación.

¹ Para el estudio se tomará como referencia a IEEE 1012-1998 debido a no tener acceso al estándar vigente IEEE 1012-2016.

De acuerdo con Wiegers y Beatty (2013), la administración de requerimientos incluye todas las actividades que mantienen la integridad, exactitud y actualidad de los requerimientos a lo largo del proyecto; estas actividades son: Control de cambios, Control de versiones, Trazabilidad de requerimientos y Seguimiento a requerimientos.

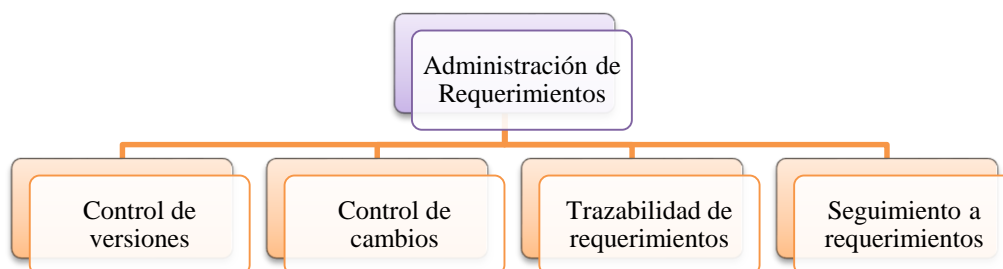


Figura 6: Actividades de la administración de requerimientos

Fuente: Elaborado por la autora

2.4.1 Control de versiones

Luego de que los requerimientos han atravesado las actividades de validación, verificación y aprobación, pasan a formar parte de una línea base. La línea base es un conjunto de requerimientos aprobados, a partir de los cuales, cualquier cambio debe ser sometido al procedimiento de control de cambios antes de ser implementado. Se puede comparar a la línea base con una instantánea de los requerimientos en un momento dado.

Cada cambio aprobado modificará la versión de un requerimiento, de un conjunto de requerimientos o de otros artefactos del proyecto. Por ello, es requerido definir un mecanismo para la identificación única de las diferentes versiones de un artefacto, lo que es conocido como control de versiones.

Existen varias formas de realizar el control de versiones, una de ellas es de manera automática mediante una herramienta informática; la otra es manual en la que se etiqueta cada versión y revisión de un artefacto, como por ejemplo: Versión 1, Revisión1, Versión 2, Revisión2.

2.4.2 Control de cambios

A lo largo del ciclo de vida de desarrollo del software y una vez obtenida la línea base, los requerimientos pueden modificarse, eliminarse o añadirse debido a diferentes factores. De acuerdo con Aurum y Wohlin (2005) los cambios a los requerimientos deben ser capturados, administrados y controlados cuidadosamente para asegurar la supervivencia del sistema.

Algunos de los factores por los cuales los requerimientos pueden cambiar son debido a cambios en el entorno de la organización, reorganización de los procesos, cambios en la legislación y mayor entendimiento del sistema por parte de los stakeholders. En la **Figura 7** se muestran diferentes fuentes de cambios en requerimientos, basados en los dominios del mercado, organización del cliente, visión del proyecto, especificación de requerimientos y solución.

Todas las peticiones de cambio deben ser analizadas para evaluar el impacto de su implementación en el sistema. Para ello es necesario definir un proceso de control de cambios que permita gestionar todas las peticiones que se realicen. Sommerville (2011) propone un proceso de control de cambios, que se resume a continuación:

- **Análisis del problema y especificación del cambio:** El proceso inicia con un problema identificado o con una propuesta de cambio. El problema o la propuesta de cambio son analizados para verificar si son válidas.
- **Análisis del cambio:** Evaluar el efecto del cambio y estimar el costo de su implementación. Al finalizar este análisis, se debe tomar una decisión sobre proceder o no con la implementación del cambio.
- **Implementación del cambio:** Modificar el documento de requerimientos, los elementos del diseño y el código.

A pesar de tener definido un proceso, el impacto que ocasiona la implementación de un cambio en el proyecto es inevitable. En realidad, el impacto se relaciona directamente con la fase de desarrollo de software en la cual se recibe la petición de cambio. En la **Tabla 12** se muestran los artefactos del proyecto que se ven impactados por un cambio, dependiendo de la fase en la cual se recibe la petición.

Tabla 12: Artefactos impactados y estrategia de implementación de una petición de cambio

Fuente: “Requirements Engineering and Management for Software Development Projects” por M. Chemuturi, 2013.

FASE		ARTEFACTOS IMPACTADOS	ESTRATEGIAS DISPONIBLES PARA IMPLEMENTACIÓN
Fase de requerimientos de usuario y de software	de	Documento de requerimientos de usuario y de software	Cuando se recibe. Antes de iniciar el diseño
Fase de diseño		Documento de requerimientos de usuario y de software. Documentos de diseño	Cuando se recibe. Antes de completar el diseño
Fase de construcción		Documento de requerimientos de usuario y de software. Documentos de diseño. Código fuente	Cuando se recibe

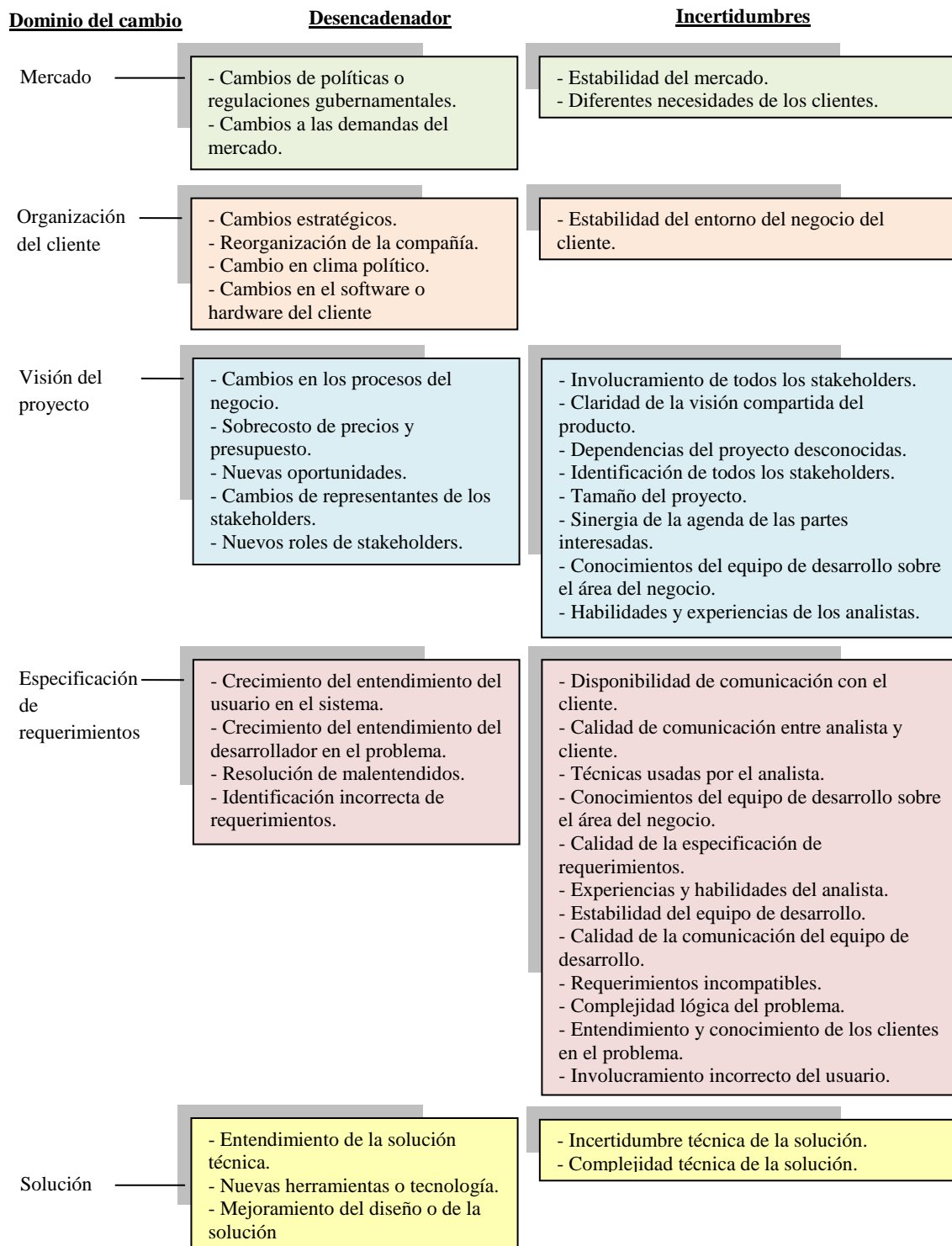


Figura 7: Taxonomía de fuentes de cambios en requerimientos de software

Fuente: "A Software Requirements Change Source Taxonomy" por McGee y Greer, 2009.

Con la definición de un proceso de control de cambios, se debe definir también un conjunto de estados para dar seguimiento a la petición y representarlos en un diagrama de transición de estados. En la **Figura 8** se muestra un diagrama de transición de estados para el cambio, que va del estado "Propuesto", "Aceptado" o "Rechazado", "Pospuesto", "Planeado" a "Incorporado".

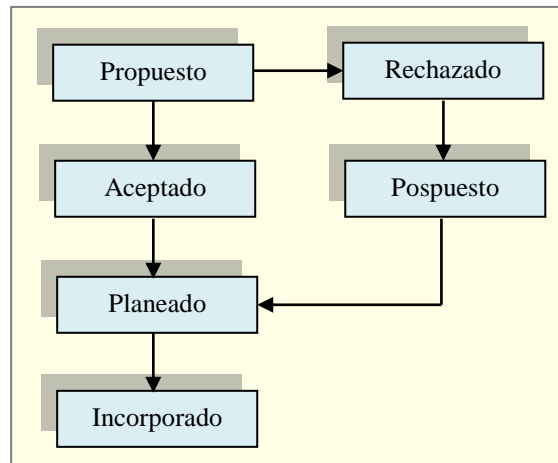


Figura 8: Diagrama de transición de estados para el control de cambios
Fuente: “Requirements Engineering” por Hull, Jackson, & Dick, 2011.

2.4.3 Trazabilidad de requerimientos

Trazabilidad es el proceso de documentar las relaciones y dependencias que un requerimiento tiene con otros artefactos del proyecto en todo su ciclo de vida (Accompa, Inc, 2017). Dentro del desarrollo de software, es esencial registrar trazabilidad y dar seguimiento a cada requerimiento, con el objetivo de garantizar que todas las necesidades del usuario sean atendidas.

En ese sentido, la trazabilidad permite rastrear un requerimiento hacia su origen: requerimiento de usuario o de negocio; y hacia su implementación: otros requerimientos, documentos de diseño, código, pruebas, características del producto final. En la **Tabla 13** se muestran algunos de los beneficios de implementar trazabilidad en los proyectos de desarrollo de software.

Tabla 13: Beneficios de implementar trazabilidad en el desarrollo de software
Fuente: Elaborado por la autora

BENEFICIO	DESCRIPCIÓN
En el proceso de desarrollo de software:	Ayuda a encontrar requerimientos perdidos, innecesarios o no implementados; y a realizar pruebas derivadas de requerimientos funcionales.
En la contratación de desarrollo de software:	Ayuda a demostrar el cumplimiento de especificaciones técnicas.
En la administración del proyecto:	Ayuda a realizar el seguimiento del proyecto al proporcionar información sobre el estado de la implementación.
En el mantenimiento del software:	Ayuda a identificar los componentes dependientes que deben modificarse debido a un mantenimiento (diseño y código)
En el proceso de control de cambios:	Ayuda a realizar el análisis de impacto, identificando los componentes que se verán afectados por un cambio.

La forma más usada para representar las relaciones entre requerimientos y otros artefactos del proyecto es la matriz de trazabilidad, la misma que puede implementarse en documentos de texto, hojas de cálculo o en herramientas de software especializadas. En la **Tabla 14**, se muestran ejemplos de relaciones de trazabilidad que se pueden implementar con el objetivo de dar seguimiento a los requerimientos.

Tabla 14: Relaciones de trazabilidad entre artefactos del proyecto

Fuente: “The Definitive Guide to Requirements Traceability” por Accompa Inc, 2014.

TRAZABILIDAD HACIA ARRIBA	TRAZABILIDAD HACIA ABAJO
Requerimiento de negocio	Requerimiento funcional
Requerimiento de negocio	Requerimiento no funcional
Característica	Requerimiento funcional
Requerimiento funcional	Caso de prueba
Caso de Uso	Caso de prueba
Requerimiento funcional	Especificación de diseño
Requerimiento funcional	Documentación de ayuda
Necesidad del cliente	Requerimiento funcional
Especificación de diseño	Código fuente

Wieggers y Beatty (2013) indican que a diferencia de los requerimientos funcionales, los requerimientos no funcionales no se trazan directamente al código; este es el caso de un requerimiento de tiempo de respuesta que puede implicar el uso de un hardware específico, diferentes estructuras de datos y enfoques de diseño.

La implementación de trazabilidad en los proyectos de desarrollo depende del tipo de industria y la criticidad del software, tal es el caso de los sistemas médicos, de tiempo real o militares. Sin embargo, no se puede olvidar los beneficios de la trazabilidad para garantizar la atención y seguimiento de todas las necesidades del usuario.

2.4.4 Seguimiento a requerimientos

El seguimiento a los requerimientos forma parte de la gestión del proyecto e intenta asegurar que los requerimientos han sido implementados de forma completa, es decir que han atravesado todas las etapas de desarrollo. Para ello, a más de la información de trazabilidad, se utilizan atributos que permiten rastrear el estado de cada uno de los requerimientos.

Proporcionar información sobre el estado actual de los requerimientos tiene mayor significado que intentar monitorear porcentajes de avance de cada uno de ellos. Por ejemplo, un reporte de seguimiento debería indicar que de los 100 requerimientos funcionales, 80 han sido implementados (código fuente) pero no probados (casos de prueba), y 20 han sido implementados y probados.

Además de realizar el seguimiento del estado de los requerimientos, es importante poder resolver problemas que aparecen durante el proceso de desarrollo tales como: información incompleta, conflictos, toma de decisiones, dudas e inquietudes. En la **Tabla 15** se muestran los tipos comunes de problemas en requerimientos de software que se deben atender durante su seguimiento.

Tabla 15: *Tipos comunes de problemas en requerimientos*

Fuente: “*Software Requirements*” por *Wiegiers y Beatty, 2013.*

TIPO DE PROBLEMA	DESCRIPCIÓN
Pregunta de requerimiento	Algo no ha sido entendido o decidido sobre un requerimiento.
Requerimiento perdido	Los desarrolladores han descubierto un requerimiento perdido durante el diseño y la implementación.
Requerimiento incorrecto	Debe ser corregido o eliminado.
Pregunta de implementación	Mientras los desarrolladores implementan un requerimiento, tienen preguntas sobre cómo algo debe trabajar o acerca de alternativas de diseño.
Requerimiento duplicado	Se han encontrado requerimientos similares o equivalentes.
Requerimiento no necesario	El requerimiento ya no se necesita.

El seguimiento es una actividad de gestión que se alimenta de la información de otras actividades de la Ingeniería de Requerimientos, tales como: elicitación, análisis, especificación, documentación, validación, control de versiones, control de cambios y trazabilidad. Sin un adecuado seguimiento, los problemas en los requerimientos se pueden volver incontrolables, lo que puede llevar al fracaso del proyecto de software.

2.5 Herramientas de Gestión de Requerimientos de Software

2.5.1 Herramientas de Administración del Ciclo de Vida del Software

Las herramientas para Administración del Ciclo de Vida (Application Lifecycle Management – ALM) sirven para dar soporte al proceso de desarrollo y mantenimiento de software; así como también para tener colaboración entre los diferentes roles que intervienen en un proyecto. Existen diferentes tipos de herramientas ALM, entre las que se pueden mencionar:

- Herramientas para administración de requerimientos.
- Herramientas para administración de tareas.
- Control de código fuente.
- Control de defectos.
- Gestión de Proyectos.
- Herramientas para administración de casos de prueba.
- Herramientas para control de versiones.
- Herramientas para ejecutar pruebas automatizadas.

2.5.2 Limitaciones del uso de Especificaciones de Requerimientos

Una Especificación de Requerimientos de Software SRS, contiene una descripción de los requerimientos de negocio, funcionales y no funcionales del sistema que se va a desarrollar. Esta especificación tiene una estructura definida, y tradicionalmente es elaborada con un editor de texto, que la almacena como un documento o archivo electrónico.

Sin embargo, de acuerdo con Wieggers & Beatty (2013) este enfoque tiene muchas limitaciones a la hora de administrar los requerimientos, tales como:

- Dificultad de mantener los documentos actualizados y sincronizados con los últimos cambios.
- Dificultad para leer la especificación y para buscar requerimientos.
- La comunicación de los cambios a los miembros del equipo del proyecto, es un proceso manual.
- Dificultad para controlar cambios y versiones de los documentos de especificación.
- No es fácil almacenar información adicional de los requerimientos, como los atributos, documentación relacionada, modelos, diagramas.
- Dificultad para definir relaciones entre requerimientos y componentes del sistema.
- Dificultad para conocer el estado y el avance de cada requerimiento.
- Dificultad para agregar, modificar, eliminar requerimientos.
- Dificultad para mantener un repositorio central de requerimientos, ya que por lo general cada miembro del equipo de trabajo tiene una copia de la especificación.
- Dificultad para identificar requerimientos perdidos, duplicados e innecesarios, debido a que la especificación no es dinámica.

2.5.3 Definición de herramienta de gestión de requerimientos

En el mercado se han implementado algunas herramientas de gestión de requerimientos de software, que son sistemas o aplicaciones informáticas que permiten almacenar, administrar y dar seguimiento a los requerimientos de un proyecto de desarrollo. El uso de una herramienta de gestión de requerimientos permite una gestión más eficiente del proyecto, así como también ayuda a solventar las limitaciones mostradas en el punto anterior. En la **Figura 9** se pueden observar algunas de las características comunes que tienen este tipo de herramientas.

2.5.4 Características de una herramienta de gestión de requerimientos

El uso de una Especificación de Requerimientos de Software en un documento de texto u hoja de cálculo, es útil para proyectos de desarrollo de software pequeños. Por el contrario, para equipos que trabajan con proyectos grandes y requerimientos que están en constante cambio, es recomendable utilizar una herramienta que permita gestionar toda la información de manera dinámica y automática. Young (2004) establece que una herramienta de gestión de requerimientos puede tener las siguientes características:

- Detalle de la visión, alcance y limitaciones del sistema.
- Un esquema predefinido para soportar el almacenamiento de requerimientos.
- Manejar atributos para los requerimientos, tales como: Prioridad, Estado, Autor, Categoría, Palabras clave, Tipo; así como la capacidad para crear atributos personalizados.
- Definición de excepción de flujos normales y condiciones de error de los requerimientos.
- Propagación automática de atributos hacia los niveles inferiores de requerimientos.
- Capacidad de almacenar y relacionar los siguientes documentos con los requerimientos: bases legales, notas de reuniones y entrevistas con clientes y usuarios, notas de talleres de requerimientos.

- Soporte de trazabilidad de requerimientos hacia adelante y hacia atrás.
- Capacidad de tener relaciones padre e hijo entre requerimientos.
- Mantener un glosario de términos y abreviaturas para el proyecto.
- Mantener las descripciones de sistema relacionados.
- Control de versiones de un requerimiento y consultas del historial de un requerimiento.
- Capacidad para realizar control de cambios sobre requerimientos.
- Generación automática y personalizada de especificaciones de requerimientos desde la base de datos.
- Capacidad de generar reportes personalizados a partir de los atributos de los requerimientos y de exportar esos reportes en diferentes formatos.
- Capacidad de restringir el ingreso o edición de requerimientos, dependiendo del rol de cada usuario.

Algunas herramientas de gestión de requerimientos existentes en el mercado, soportan el estándar ReqIF – Requirements Interchange Format, que es un formato abierto y no propietario de intercambio de información de requerimientos de software a través del uso de XML. Este estándar permite el intercambio de requerimientos entre equipos de desarrollo que utilizan herramientas diferentes para su gestión, evitando la tarea de reescribirlos y facilitando su administración.

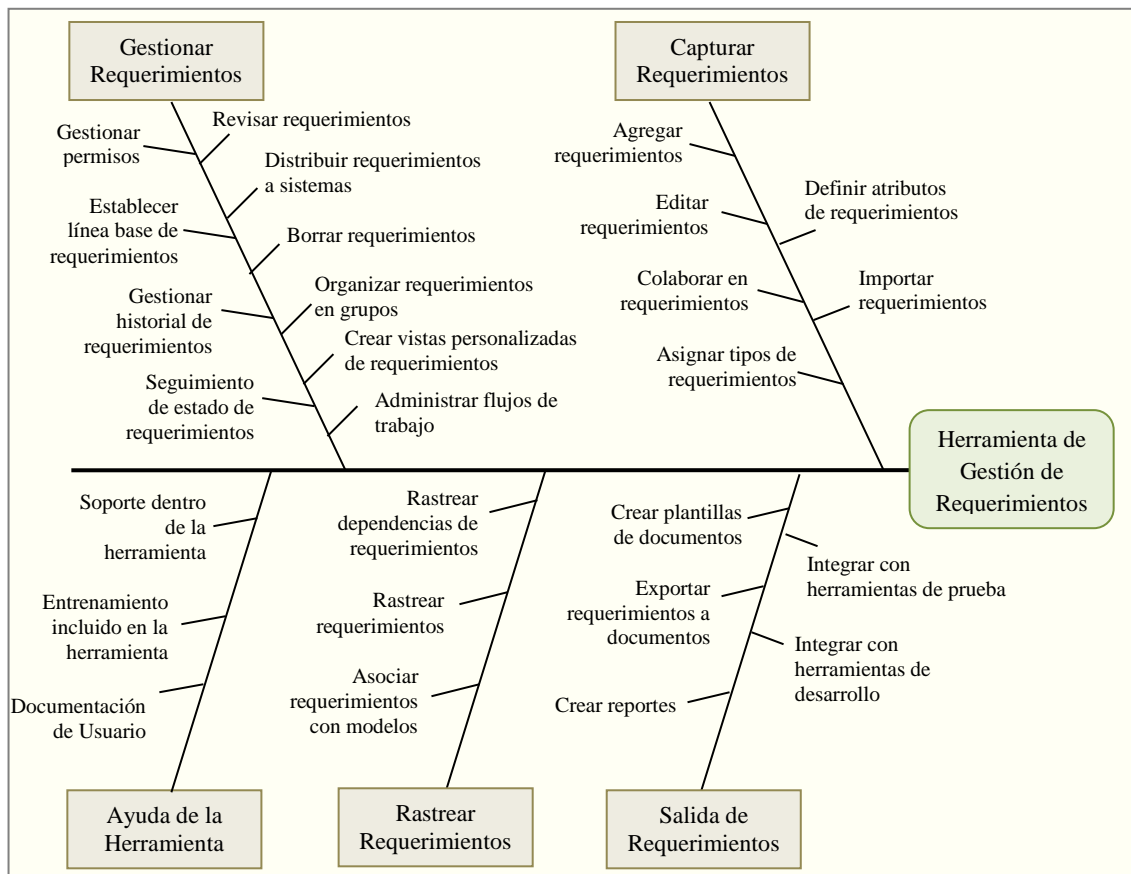


Figura 9: Características de una herramienta de gestión de requerimientos

Fuente: “Software Requirements” por Wiegers y Beatty, 2013..

2.5.5 Herramientas de gestión de requerimientos existentes en el mercado

En el sitio web de Volere (Atlantic Systems Guild, 2016) se mantiene un conjunto de recursos relacionados con la Ingeniería de Requerimientos, que son resultado de muchos años de investigación y recopilación de buenas prácticas en la industria del desarrollo de software. Entre estos recursos se tienen procesos, guías, plantillas, así como un listado de herramientas que soportan la Ingeniería de Requerimientos e Ingeniería de Software, tal como se puede observar en la **Tabla 16**.

Tabla 16: *Herramientas comerciales de Ingeniería de Requerimientos e Ingeniería de Software*
Fuente: *Elaborado con contenido de “Volere Requirements Resources” por Atlantic Systems Guild, 2016.*

HERRAMIENTA	COMPAÑÍA	SITIO WEB
Caliber 11.5	Micro Focus – Reino Unido y Estados Unidos	https://www.microfocus.com/products/requirements-management/caliber
Axure RP 8	Axure Software Solutions – California, Estados Unidos	http://www.axure.com
Accompa	Accompa, Inc – California, Estados Unidos	http://web.accompa.com
Avenqo PEP 1.9	Avenqo GmbH - Alemania	http://www.avenqo.com
Agosense Requirements	Agosense - Alemania	http://www.agosense.com/agosense/requirements
Rational DOORS 9.6.1	IBM Corporation – Nueva York, Estados Unidos	http://www-03.ibm.com/software/products/en/ratidoor
CaseComplete	Serlio Software – Wisconsin, Estados Unidos	http://www.casecomplete.com
codeBeamerRequirements Management 8	Intland Software - Alemania	https://intland.com/requirements-management
CognitionCockpit	CognitionCorporation – Massachusetts, Estados Unidos	http://www.cognition.us
CradleVersion 7.1	Structured Software SystemsLtd -	https://www.threesl.com/
Gatherspace	Gatherspace – California, Estados Unidos	http://www.gatherspace.com
Polarion REQUIREMENTS	Polarion Software – California, Estados Unidos	https://polarion.plm.automation.siemens.com/products/polarion-requirements
ProR	Eclipse Foundation – Ottawa, Canadá	http://www.eclipse.org/rmf/pror
REQCHECKER 1.5.2	SGSOFT France - Francia	http://reqchecker.eu
RequirementONE	RequirementOneInc – California, Estados Unidos	http://www.requirementone.com
ReqSuite	OSSENO Software GmbH - Alemania	http://www.osseno.de
RMTrak	Prometeo Technologies – Nueva York, Estados Unidos	https://www.rmtrak.com
VisureRequirements	VisureSolutions – San Francisco, Estados Unidos	http://www.visuresolutions.com/requirements-engineering-tool

3. CAPÍTULO III. MATERIALES Y MÉTODOS

3.1 Lugar de desarrollo de la investigación

3.1.1 Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra

El Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra GAD-I es una persona jurídica de derecho público con autonomía política, administrativa y financiera. Cumple con las funciones de Legislación, normatividad y fiscalización; de Ejecución y administración; de Participación ciudadana y control social en el cantón Ibarra.

De acuerdo a lo establecido en el Código Orgánico de Organización Territorial, Autonomía y Descentralización COOTAD, las competencias del GAD-I son:

- a) Planificar el desarrollo cantonal y formular los planes de ordenamiento territorial.
- b) Ejercer el control sobre el uso y ocupación del suelo.
- c) Planificar, construir y mantener la vialidad urbana.
- d) Prestar los servicios públicos de agua potable, alcantarillado, depuración de aguas residuales, manejo de desechos sólidos, actividades de saneamiento ambiental.
- e) Crear, modificar, exonerar o suprimir mediante ordenanzas, tasas, tarifas y contribuciones especiales de mejoras.
- f) Planificar, regular y controlar el tránsito y el transporte terrestre dentro de su circunscripción cantonal.
- g) Preservar, mantener y difundir el patrimonio arquitectónico, cultural y natural del cantón y construir los espacios públicos para estos fines.
- h) Elaborar y administrar los catastros inmobiliarios urbanos y rurales.
- i) Delimitar, regular, autorizar y controlar el uso de las playas de mar, riberas y lechos de ríos, lagos y lagunas.
- j) Preservar y garantizar el acceso de las personas al uso de las playas de mar, riberas de ríos, lagos y lagunas.
- k) Regular, autorizar y controlar la explotación de materiales áridos y pétreos, que se encuentren en los lechos de los ríos, lagos, playas de mar y canteras.
- l) Gestionar los servicios de prevención, protección, socorro y extinción de incendios.
- m) Gestionar la cooperación internacional para el cumplimiento de sus competencias.

Con el objetivo de regular, organizar y administrar el funcionamiento de todos sus órganos administrativos, el GAD-I expide el Estatuto Orgánico de Gestión Organizacional, en el cual se definen las funciones y responsabilidades de cada uno de los procesos Gobernantes, de Asesoría, Agregadores de Valor, y de Apoyo (GAD-I, 2016).

3.1.2 Dirección de Tecnologías de la Información y Comunicación

Dentro de la Estructura Organizacional del GAD-I, la Dirección de Tecnologías de la Información y Comunicación TIC tiene la responsabilidad de aportar con soluciones tecnológicas de software, hardware y comunicaciones que soporten las actividades y procesos institucionales. Para ello, en la Dirección de TIC se establecen las unidades administrativas que se muestran en la **Figura 10** y que son: Desarrollo de Software, Infraestructura y Comunicaciones, Innovación y Gobierno Electrónico.

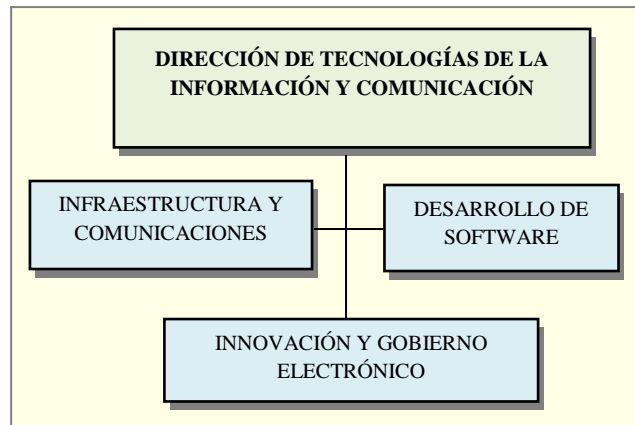


Figura 10: Estructura Orgánica de la Dirección de TIC del GAD-I

Fuente: “Estatuto Orgánico de Gestión Organizacional por Procesos” por Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra, 2016.

3.1.3 Unidad de Desarrollo de Software

De acuerdo a lo establecido en el Estatuto Orgánico de Gestión Organizacional por Procesos del GAD-I, la Unidad de Desarrollo de Software realiza actividades de Respaldo y restauración de datos electrónicos; Gestión de acceso a servicios y aplicaciones; Atención, capacitación y soporte técnico al usuario; Desarrollo de software; Mantenimiento de software desarrollado y de terceros. Todos los requerimientos de software de la institución son asignados a esta Unidad; donde se tienen sistemas informáticos desarrollados internamente, desarrollados por empresas externas, e implantaciones de software libre.

Como se puede ver en la **Tabla 17**, el 85.2% de sistemas informáticos existentes en el GAD-I fueron desarrollados al interior de la institución, mientras que el 14.8% fueron desarrollados por empresas proveedoras de servicios. Con el tiempo, el número de sistemas informáticos internos o externos puede crecer y los existentes pueden someterse a modificaciones; todo esto debido a cambios en la normativa legal vigente, expedición de nuevas leyes u ordenanzas municipales, o requerimientos institucionales.

Tabla 17: Sistemas informáticos existentes en la Dirección de TIC – 2017

Fuente: Unidad de Desarrollo de Software del GAD-I, 2017

Nº	SISTEMA INFORMÁTICO	TIPO DESARROLLO
1	Avalúos y Catastros	Interno
2	Recaudación y Tesorería	Interno
3	Transferencias de Dominio	Interno
4	Actividades Económicas	Interno
5	Control Vehicular IMI	Interno
6	Control de Obras Públicas	Interno
7	Control Urbano – Multas de Construcciones	Interno
8	Tránsito y Transporte	Interno
9	Inquilinato	Interno
10	Archivo Documental	Interno
11	Recursos Humanos	Interno
12	Financiero Contable	Externo

Continuación de la **Tabla 17**

N°	SISTEMA INFORMÁTICO	TIPO DESARROLLO
13	Información Geográfica	Interno
14	Participación Ciudadana	Interno
15	Estacionamiento SISMERT	Interno
16	Gestión Documental Quipux	Externo
17	Informe de Reglamentación Cantonal	Interno
18	Publicidad Exterior	Interno
19	Aplicación consulta Registro Civil	Interno
20	Aplicación consulta SRI	Interno
21	Aplicación reportes Parquímetros	Interno
22	Temáticos Catastros	Interno
23	Kiosko Electrónico	Interno
24	Audiencias Ciudadanas	Interno
25	Sistema Integrado de Gestión Municipal	Externo
26	Portal Ciudadano	Externo
27	Administración de Sistemas	Interno

3.2 Tipo de Investigación

Para el desarrollo del presente trabajo se utilizó un enfoque de investigación cualitativo, que de acuerdo con Hernández, Fernández & Baptista (2010), se orienta a comprender y profundizar los fenómenos explorándolos desde la perspectiva de los participantes en un ambiente natural y en relación con el contexto. Este enfoque tienen las siguientes características:

- Se basa en la lógica y sigue un proceso inductivo.
- Utiliza métodos de recolección de datos no estandarizados ni completamente predeterminados.
- No se efectúa medición numérica, por ello el análisis no es estadístico.
- La recolección de los datos consiste en obtener las perspectivas, opiniones y puntos de vista de los participantes, tales como: emociones, prioridades, experiencias, significados, y vivencias.

En ese sentido, las técnicas que se usaron para la recolección de datos fueron: observación, entrevistas abiertas, revisión de documentos, talleres de trabajo, tal como se describe en la **Tabla 18**.

3.3 Diseño de la Investigación

Se utilizaron los diseños de Investigación aplicada, Investigación de Campo e Investigación Documental, tal como se describe a continuación:

- **Investigación Aplicada:** En esta modalidad de investigación se aplican y utilizan los conocimientos existentes y resultantes de la investigación básica. Tiene por finalidad la búsqueda y afirmación del saber.

Tabla 18: Técnicas usadas en la investigación

Fuente: Elaborado por la autora

TÉCNICA	DESCRIPCIÓN
Observación	Observación directa de los procesos que se ejecutan en las instalaciones de la Unidad de Desarrollo de Software del GAD-I: <ul style="list-style-type: none">- Proceso de desarrollo de software- Proceso de mantenimiento de software.
Revisión de documentos	Revisión de toda la documentación existente en la Unidad de Desarrollo de Software, tales como: procesos, procedimientos, registros históricos, información de proyectos, registro de peticiones y atenciones, documentación de sistemas, inventario de sistemas, informes de trabajo.
Entrevista	Entrevistas al personal de la Dirección de TIC: Director, Responsable de la Unidad de Desarrollo de Software y Analistas de Sistemas; con el objetivo de recopilar información acerca del problema de estudio y de las opciones de mejora.
Encuesta	Encuestas a los funcionarios del GAD-I que sean usuarios de los sistemas informáticos, con el objetivo de recopilar información del nivel de satisfacción de los servicios y productos entregados por la Unidad de Desarrollo de Software. Encuestas al siguiente personal de la Dirección de TIC: Responsable de la Unidad de Desarrollo de Software y Analistas de Sistemas, con el objetivo de evaluar los conocimientos de los profesionales en temas de Ingeniería de Requerimientos.
Talleres de trabajo	Se realizaron varios talleres en los que se definió el proceso de gestión de requerimientos de software mediante el análisis de experiencias de la Unidad y la discusión en grupo con los Analistas de Sistemas.

- **Investigación de Campo:** En esta modalidad de investigación se trabaja en el ambiente natural en que se desarrolla el problema, de dónde se obtendrán los datos a ser analizados.
- **Investigación Documental– Bibliográfica:** En esta modalidad se trata de indagar, interpretar y presentar datos sobre un tema determinado, usando el análisis, síntesis, deducción e inducción. Para ello se consultan las fuentes de información primaria

(libros, revistas científicas, diarios, documentos oficiales, informes técnicos, patentes, normas técnicas) y las fuentes de información secundaria (enciclopedias, directorios, libros o artículos que interpretan otros trabajos o investigaciones).

3.4 Población de Estudio

La población con la que se trabajó fueron los funcionarios de la Unidad de Desarrollo de Software de la Dirección de TIC y funcionarios de Direcciones Departamentales que son potenciales usuarios de sistemas informáticos, tal como se puede observar en la **Tabla 19**.

Tabla 19: Población de estudio

Fuente: Elaborado por la autora

POBLACIONES	FRECUENCIA	PORCENTAJE
Director de Tecnologías Informáticas	1	0.32%
Responsable del Área de Desarrollo de Software	1	0.32%
Analistas de Sistemas	7	2.26%
Funcionarios del GAD-I (potenciales usuarios de sistemas informáticos)	100	97.1%
• Dirección Financiera (20)		
• Dirección de Gestión Tributaria (10)		
• Dirección de Avalúos y Catastros (10)		
• Dirección de Planificación de Desarrollo Urbano y Rural (15)		
• Dirección de Obras Públicas (10)		
• Dirección de Gestión Ambiental (10)		
• Dirección Administrativa (10)		
• Dirección de Participación Ciudadana (15)		
TOTAL	109	100%

Con estos antecedentes, se han identificado las características de los funcionarios de la Unidad de Desarrollo de Software que se muestran en la **Tabla 20**, mismas que se consideraron durante la elaboración del estudio. No se analizaron características de los usuarios de sistemas informáticos, debido a que la guía metodológica va a ser usada y diligenciada por los funcionarios de la Unidad indicada.

Tabla 20: Características de la Unidad de Desarrollo de Software del GAD-I

Fuente: Elaborado por la autora

CARACTERÍSTICA	DESCRIPCIÓN
Roles del equipo de trabajo:	Un Director de Tecnologías de la Información y Comunicación. Siete Analistas de Sistemas. Un Responsable de Desarrollo de Software.
Estabilidad del personal:	Ocho funcionarios tienen nombramiento. Un funcionario tiene nombramiento de libre remoción.
Perfiles de los miembros del equipo de trabajo:	Licenciados e Ingenieros en Sistemas Computacionales.
Experiencia de los miembros del equipo de trabajo:	De 3 a 10 años de experiencia en el desarrollo de sistemas informáticos para Municipio y conocimiento del negocio.
Distribución geográfica del equipo de trabajo:	No se encuentran distribuidos geográficamente.
Tipos de sistemas:	Sistemas gubernamentales regulados por leyes ecuatorianas y ordenanzas municipales.
Stakeholders distribuidos geográficamente:	Los stakeholders si se encuentran distribuidos geográficamente dentro del área urbana del cantón. (El tiempo y la distancia pueden limitar el acceso a los stakeholders)
Tipo de desarrollo:	Interno: desarrollado por la Unidad de Desarrollo de Software. Externo: desarrollado por empresas externas proveedoras de servicios.

3.5 Metodología de trabajo para el desarrollo del Objetivo Específico 1

En la **Figura 11** y **Figura 12** se puede observar la metodología de trabajo utilizada para el cumplimiento del primer objetivo específico, la cual está dividida en actividades y tareas, donde cada una de ellas tiene una salida que sirve como medio de verificación de su ejecución. Una descripción detallada de las actividades y tareas de la metodología se encuentra en la **Tabla 21**.

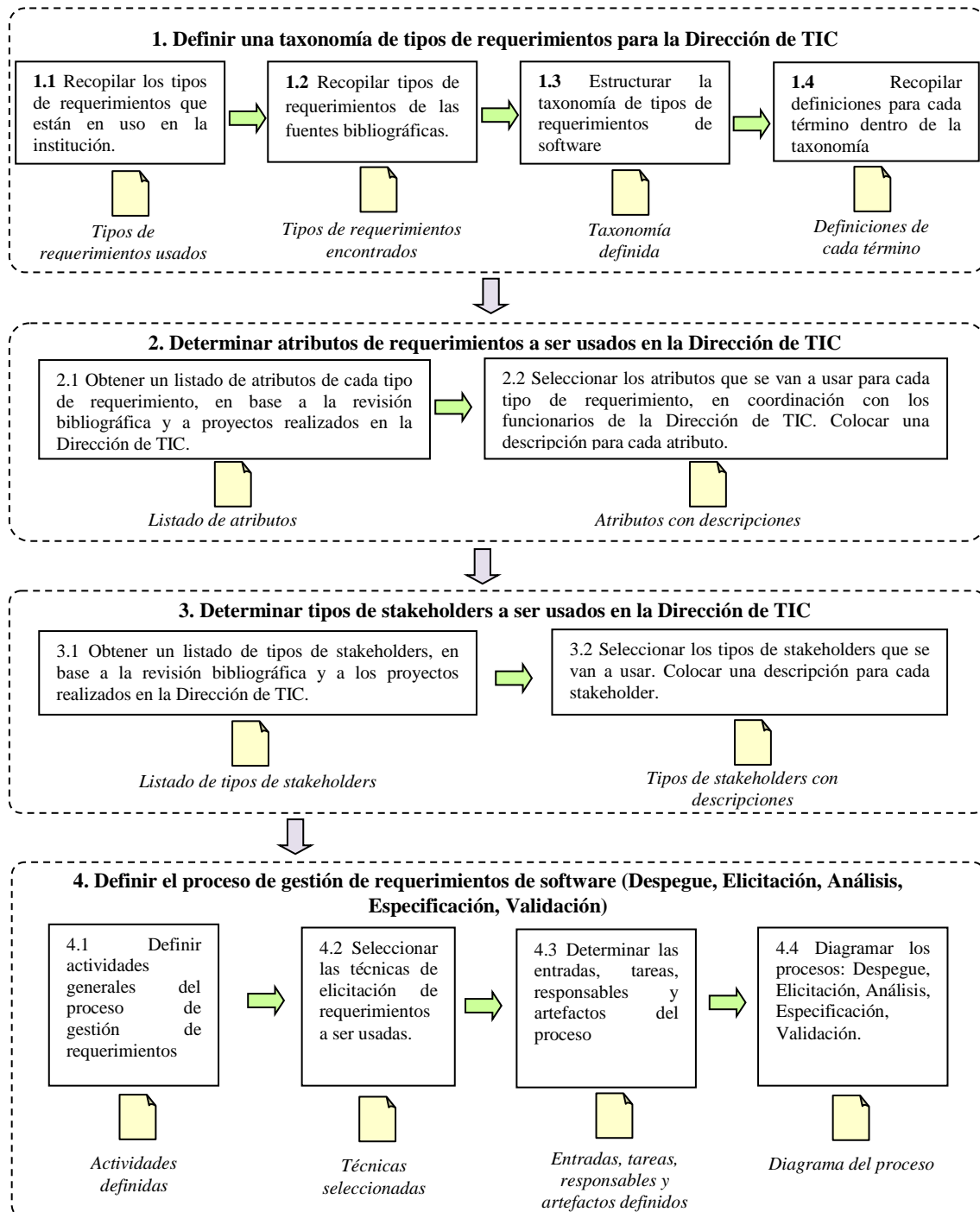


Figura 11: Metodología de trabajo para el objetivo específico 1 - Primera Parte

Fuente: Elaborado por la autora

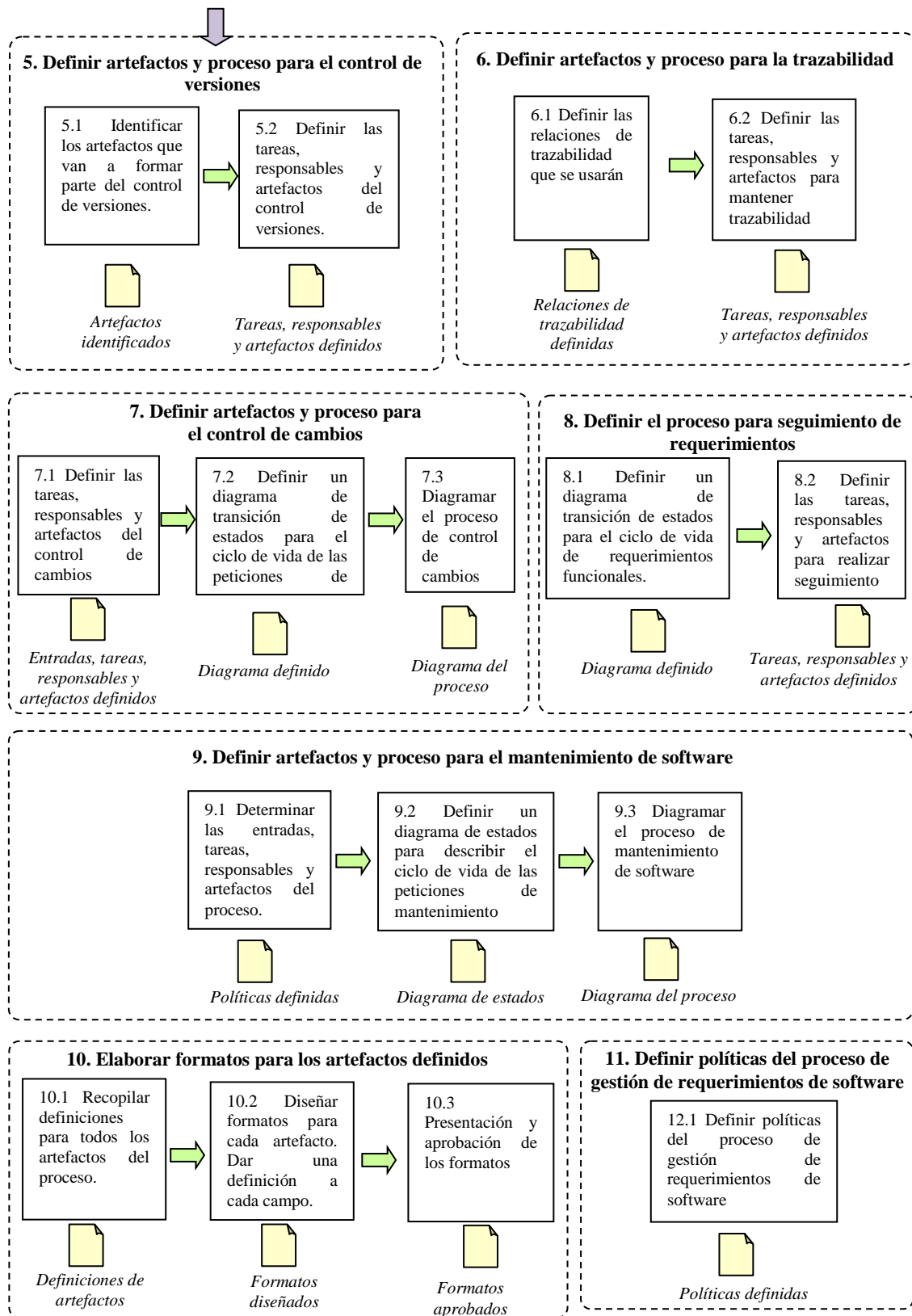


Figura 12: Metodología de trabajo para el objetivo específico 1 - Segunda Parte

Fuente: Elaborado por la autora

Tabla 21: Metodología de trabajo para el desarrollo del primer objetivo específico

Fuente: Elaborado por la autora

Objetivo Específico		Determinar un proceso de gestión de requerimientos de software para la Dirección de TIC del GAD-I, mediante el análisis y aplicación de buenas prácticas de la Ingeniería de Requerimientos.	
N°	Actividades	Tareas	Salidas
1.	Definir una taxonomía de tipos de requerimientos para la Dirección de TIC	1.1 Recopilar todos los tipos de requerimientos que actualmente están en uso en la institución.	- Tipos de requerimientos usados
		1.2 Recopilar tipos de requerimientos de las fuentes bibliográficas.	- Tipos de requerimientos encontrados
		1.3 Estructurar la taxonomía de tipos de requerimientos de software en base a las necesidades institucionales.	- Taxonomía de tipos de requerimientos definida
		1.4 Recopilar definiciones para cada término dentro de la taxonomía	- Definiciones de cada término
2.	Determinar atributos de requerimientos a ser usados en la Dirección de TIC	2.1 Obtener un listado de atributos de cada tipo de requerimiento, en base a la revisión bibliográfica y a proyectos realizados en la Dirección de TIC.	- Listado de atributos de requerimientos encontrados
		2.2 Seleccionar los atributos que se van a usar para cada tipo de requerimiento, en coordinación con los funcionarios de la Dirección de TIC. Colocar una descripción para cada atributo.	- Listado de atributos de requerimientos con descripciones
3.	Determinar tipos de stakeholders a ser usados en la Dirección de TIC	3.1 Obtener un listado de tipos de stakeholders, en base a la revisión bibliográfica y a los proyectos realizados en la Dirección de TIC.	- Listado de tipos de stakeholders
		3.2 Seleccionar los tipos de stakeholders que se van a usar, en coordinación con los funcionarios de las Dirección de TIC. Colocar una descripción para cada stakeholder.	- Listado de stakeholders con descripciones
4.	Definir el proceso de gestión de requerimientos de software de la Dirección de TIC	4.1 Definir actividades generales del proceso de gestión de requerimientos.	- Actividades definidas

Continuación de la **Tabla 21.**

N°	Actividades	Tareas	Salidas
4.	Definir el proceso de gestión de requerimientos de software de la Dirección de TIC	4.2 Seleccionar las técnicas de elicitación de requerimientos a ser usadas.	- Técnicas seleccionadas
		4.3 Determinar las entradas, tareas, responsables y artefactos del proceso de gestión de requerimientos: Elicitación, Análisis, Especificación, Validación	- Entradas, tareas, responsables y artefactos definidos
		4.4 Diagramar el proceso de gestión de requerimientos de software: Elicitación, Análisis, Especificación, Validación.	- Diagrama del proceso
5	Definir artefactos y proceso para el control de versiones	5.1 Identificar los artefactos que van a formar parte del control de versiones.	- Artefactos identificados
		5.2 Definir las tareas, responsables y artefactos del control de versiones.	- Tareas, responsables y artefactos definidos
6	Definir artefactos y proceso para la trazabilidad	6.1 Definir las relaciones de trazabilidad que se usarán.	- Relaciones de trazabilidad definidas
		6.2 Definir las tareas, responsables y artefactos para mantener trazabilidad.	- Tareas, responsables y artefactos definidos
7	Definir artefactos y proceso para el control de cambios	7.1 Definir las tareas, responsables y artefactos del control de cambios.	- Tareas, responsables y artefactos definidos
		7.2 Definir un diagrama de transición de estados para el ciclo de vida de las peticiones de cambio.	- Diagrama de transición de estados de peticiones de cambio
		7.3 Diagramar el proceso de control de cambios	- Diagrama del proceso
8	Definir el proceso para seguimiento de requerimientos	8.1 Definir un diagrama de transición de estados para el ciclo de vida de requerimientos funcionales.	- Diagrama de transición de estados requerimientos funcionales
		8.2 Definir las tareas, responsables y artefactos para realizar seguimiento	- Tareas, responsables y artefactos definidos

Continuación de la **Tabla 21.**

N°	Actividades	Tareas	Salidas
9	Definir artefactos y proceso para el mantenimiento de software	9.1 Determinar las entradas, tareas, responsables y artefactos del proceso de mantenimiento de software desarrollado.	- Entradas, tareas, responsables y artefactos definidos
		9.2 Definir un diagrama de transición de estados para describir el ciclo de vida de las peticiones de mantenimiento.	- Diagrama de transición de estados definido.
		9.3 Diagramar el proceso de mantenimiento de software.	- Diagrama del proceso
10	Elaborar formatos para todos los artefactos definidos	10.1 Recopilar definiciones para todos los artefactos del proceso.	- Definiciones de artefactos del proceso.
		10.2 Diseñar formatos para cada artefacto. Dar una definición a cada campo.	- Diseños de formatos de artefactos
		10.3 Presentar los formatos a los funcionarios de la Dirección de TIC, realizar cambios y obtener aprobación.	- Formatos de artefactos aprobados
11	Definir políticas del proceso de gestión de requerimientos de software	11.1 Definir políticas del proceso de gestión de requerimientos de software de la Dirección de TIC.	- Políticas definidas

3.6 Metodología de trabajo para el desarrollo del Objetivo Específico 2

En la **Figura 13** se puede observar la metodología de trabajo desarrollada para el cumplimiento del segundo objetivo específico, la cual está dividida en actividades y tareas, donde cada una de ellas tiene una salida que sirve como medio de verificación de su ejecución. Una descripción detallada de las actividades y tareas de la metodología se encuentra en la **Tabla 22**.

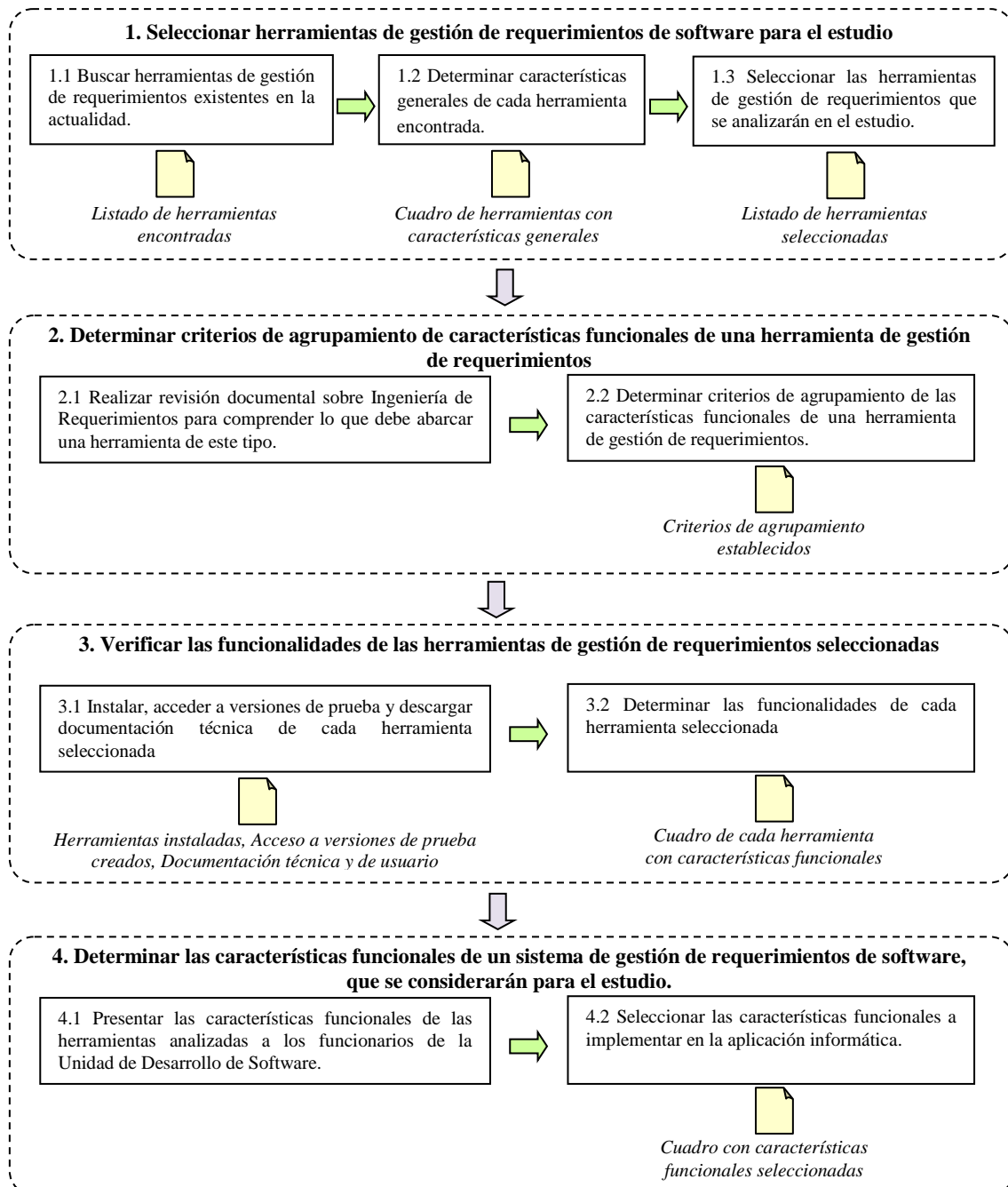


Figura 13: Metodología de trabajo para el cumplimiento del segundo objetivo específico

Fuente: Elaborado por la autora

Se debe aclarar que este objetivo específico no busca realizar un análisis comparativo, sino más bien realizar un análisis de las características funcionales que tienen este tipo de herramientas y seleccionar aquellas características que se puedan implementar en la aplicación informática de gestión de requerimientos de la Dirección de TIC.

Tabla 22: Metodología de trabajo para el desarrollo del Objetivo Específico 2

Fuente: Elaborado por la autora

Objetivo Específico		Determinar las características funcionales de un sistema de gestión de requerimientos de software para la Dirección de TIC, mediante el análisis de herramientas existentes en el mercado.	
Nº	Actividades	Tareas	Salidas
1	Seleccionar un conjunto de herramientas de gestión de requerimientos de software existentes en el mercado para el estudio.	1.1 Buscar en libros, revistas y sitios web, herramientas de gestión de requerimientos existen en la actualidad.	- Listado de herramientas encontradas
		1.2 De cada herramienta encontrada, determinar las siguientes características generales: <ul style="list-style-type: none"> - Vigencia de la herramienta. - Acceso a versiones de prueba. - Acceso a documentación. - Coherencia de la herramienta con el alcance del presente estudio. - Idioma de la herramienta. 	- Cuadro resumen de las herramientas encontradas con sus características generales
		1.3 Seleccionar el conjunto de herramientas de gestión de requerimientos de software para su análisis, en base a los resultados obtenidos en la tarea anterior.	- Listado de herramientas seleccionadas
2	Determinar criterios de agrupamiento de las características funcionales de una herramienta de gestión de requerimientos de software.	2.1 Realizar una revisión documental sobre la Ingeniería de Requerimientos, con el fin de comprender lo que debe contener una herramienta de este tipo.	
		2.2 Determinar los criterios de agrupamiento para el estudio.	- Criterios de agrupamiento establecidos
3	Verificar las funcionalidades de cada herramienta mediante la instalación, uso, y revisión de documentación técnica, dependiendo del caso.	3.1 Instalar o solicitar acceso a versiones de prueba de cada herramienta, y obtener documentación técnica y de usuario a través de su sitio oficial.	- Herramientas instaladas. - Acceso a versiones de prueba creados. - Documentación técnica y de usuario.

Continuación de la **Tabla 22**.

Nº	Actividades	Tareas	Salidas
3	Verificar las funcionalidades de cada herramienta mediante la instalación, uso, y revisión de documentación técnica, dependiendo del caso.	3.2 Determinar las funcionalidades de cada herramienta, mediante el uso, revisión de la documentación técnica, observación de video tutoriales, y/o demostraciones por parte de la empresa.	- Cuadro resumen de cada herramienta con las características funcionales encontradas
4	Determinar las características funcionales de un sistema de gestión de requerimientos de software, que se considerarán para el presente estudio.	4.1 Realizar una presentación de las características funcionales de las herramientas analizadas a los funcionarios de la Unidad de Desarrollo de Software.	
		4.2 Seleccionar las características funcionales a implementar en la aplicación informática, en base a las necesidades de la Unidad de Desarrollo de Software y al alcance del presente estudio.	- Cuadro resumen con las características funcionales seleccionadas.

3.7 Metodología de trabajo para el desarrollo del Objetivo Específico 3

En la **Figura 14** se muestra la metodología de trabajo para el desarrollo de la aplicación informática, que básicamente consta de la etapa de Levantamientos y Análisis de Requerimientos, Diseño, Construcción, Pruebas e Implantación. El modelo elegido para el ciclo de vida de desarrollo fue el iterativo.

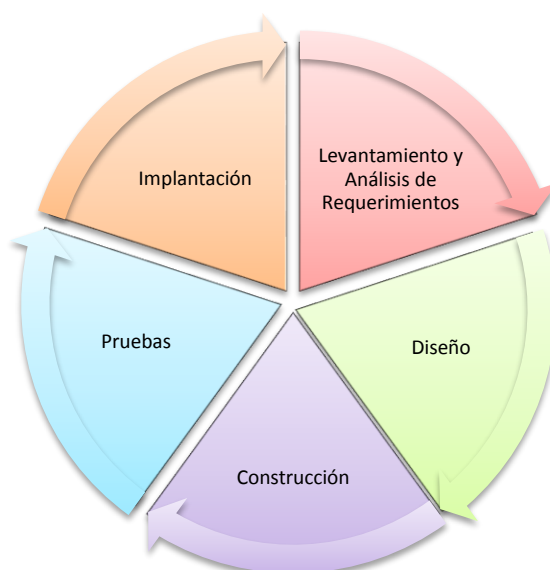


Figura 14: Metodología de desarrollo de la aplicación informática

Fuente: Elaborado por la autora

3.7.1 Levantamiento y Análisis de requerimientos

Debido a que la aplicación informática del presente estudio se obtuvo como resultado de la definición de un proceso de gestión de requerimientos de software y del análisis de un grupo de herramientas existentes en el mercado, los requerimientos que guiaron su construcción fueron los definidos al aplicar las metodologías detalladas anteriormente.

3.7.2 Diseño

En la etapa de Diseño se trabajó en base a lo establecido en la Unidad de Desarrollo de Software, elaborando el Diagrama de Arquitectura de la aplicación informática y el Modelo Entidad Relación mostrado en el **Anexo 3**.

3.7.3 Construcción

La construcción de la aplicación informática se basó en lo definido en la Unidad de Desarrollo de Software en cuanto a herramientas, paquetes y estándares de desarrollo de software.

3.7.4 Pruebas

Las pruebas de la aplicación informática se basaron en lo definido en la Unidad de Desarrollo de Software, ejecutando pruebas unitarias y de integración.

3.7.5 Implantación

La implantación de la aplicación informática se basó en las políticas existentes en la Unidad de Desarrollo de Software, desplegando el proyecto en los servidores del Data Center institucional.

4. CAPÍTULO IV. DESARROLLO DE LA PROPUESTA

En este capítulo se detalla todo el trabajo realizado para la consecución de los objetivos específicos, en base a la metodología de trabajo expuesta en el capítulo anterior.

4.1. Desarrollo del Objetivo Específico 1

El primer objetivo específico es el de determinar un proceso de gestión de requerimientos de software para la Dirección de TIC, mediante el análisis y aplicación de las buenas prácticas de la Ingeniería de Requerimientos.

4.1.1. Definir una taxonomía de tipos de requerimientos

- **Recopilar todos los tipos de requerimientos que actualmente están en uso en la institución:** Para efectuar esta tarea se realizó una revisión de los procesos de desarrollo y mantenimiento de software, revisión del historial de peticiones de desarrollo o mantenimiento, así como consultas a los funcionarios que laboran en la Unidad. Como resultado, se concluye que en la Unidad de Desarrollo de Software de la Dirección de TIC se manejan requerimientos de tipo funcional, tal como se describe a continuación:
 - **Revisión de procesos de desarrollo y mantenimiento de software.**

En los procedimientos y formatos manejados, únicamente se especifica el levantamiento de requerimientos funcionales. En el procedimiento de desarrollo de software se tiene el formato de “Historia de Usuario”, que permite recolectar los requerimientos funcionales desde el punto de vista del usuario. En el procedimiento de mantenimiento de software se tiene el formato de “Orden de Mantenimiento y Pedidos varios”, en el que se coloca la descripción textual del requerimiento funcional.
 - **Revisión del historial de peticiones de desarrollo o mantenimiento de software.**

En los pedidos de desarrollo y mantenimiento de software, así como en la documentación que justifica su atención, se han encontrado únicamente referencias a requerimientos funcionales. En algunos Términos de Referencia para la adquisición de software, se encontraron especificaciones de requerimientos no funcionales, sin embargo no se encontró documentación técnica que muestre cómo se realizó la gestión y el cumplimiento de ese tipo de requerimientos.
 - **Consultas a los funcionarios de la Unidad.**

La Responsable de la Unidad de Desarrollo de Software y los Analistas de Sistemas manifiestan que todos los requerimientos manejados son funcionales, es decir aquellos que describen el comportamiento de los sistemas informáticos requeridos por los usuarios. Indican que es posible que se tengan otros tipos de requerimientos como de restricciones o interfaces, pero todos se manejan bajo el mismo término de “requerimientos funcionales”. Esto no significa que los sistemas informáticos no cumplan con criterios de calidad, sino más bien que no existe un mecanismo para la identificación, especificación, seguimiento y validación de todos los tipos de requerimientos existentes.

- **Recopilar tipos de requerimientos de las fuentes bibliográficas:** En el Capítulo II de Marco Referencial, se pueden encontrar referencias a los tipos de requerimientos de software encontrados en las fuentes bibliográficas, que servirán para definir la taxonomía de requerimientos de la Dirección de TIC.
- **Estructurar la taxonomía de tipos de requerimientos de software en base a las necesidades institucionales:** Luego de reuniones realizadas con la Unidad de Desarrollo de Software, se pudo estructurar la taxonomía de tipos de requerimientos que se observa en la **Figura 15**. Cada tipo de requerimiento fue analizado con el objetivo de garantizar que los funcionarios de la Unidad entiendan su definición, así como las diferencias entre uno y otro.

De acuerdo a lo expuesto por Wiegers & Beatty (2013) las “Reglas de Negocio” no son un tipo de requerimiento en sí mismo ya que existen fuera de la existencia de cualquier software, sin embargo son el origen de varios tipos de requerimientos. Por este motivo, las reglas de negocio no se han incluido dentro de la taxonomía, pero si se las considera como fuente de información de requerimientos para la Dirección de TIC.

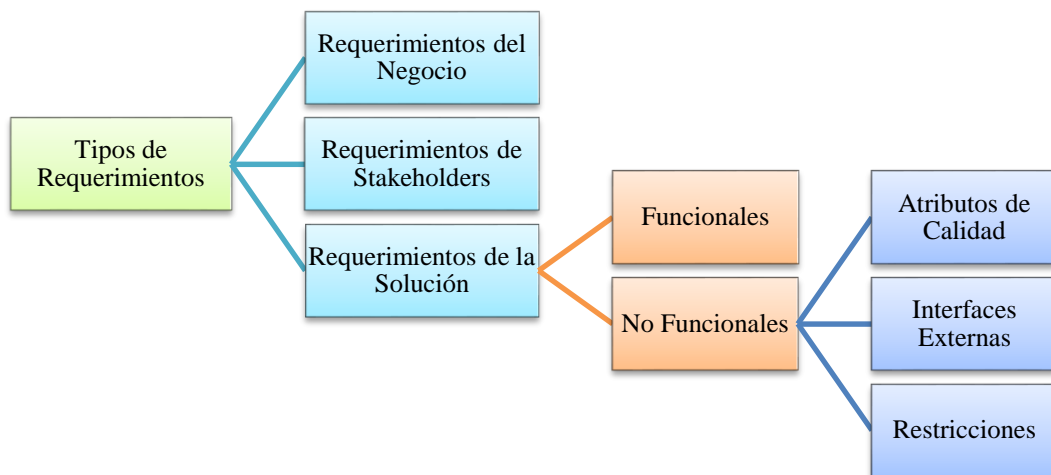


Figura 15: Taxonomía de tipos de requerimientos de software para la Dirección de TIC
Fuente: Elaborado por la autora

- **Recopilar definiciones para cada término dentro de la taxonomía:** Las definiciones que se presentan en la **Tabla 23**, han sido formadas a partir del contenido encontrado en las siguientes fuentes bibliográficas: The Requirements Engineering Handbook (Young, 2004), A Guide to the Business Analysis Body of Knowledge - BABOK V2.0 (IIBA, 2009). Además, se han definido siglas que inician con REQ y continúan con dos letras que identifican al tipo de requerimiento.

Se coloca también la definición para el término “**Reglas de Negocio**”: Políticas, guías, estándares, o regulaciones que definen o restringen las actividades del negocio, tales como:

- Regulaciones gubernamentales: leyes, decretos, reglamentos.
- Regulaciones de la organización: políticas, normas, procedimientos, resoluciones.
- Regulaciones propias de la aplicación: estándares y mejores prácticas de desarrollo.

Tabla 23: Definiciones de tipos de requerimientos a ser usados en la Dirección de TIC

Fuente: Tabla elaborada en base a las definiciones de fuentes bibliográficas

Tipo	Sigla	Definición
Requerimientos del Negocio	REQNE	<ul style="list-style-type: none"> - Son declaraciones de alto nivel de las metas, objetivos o necesidades que la organización espera lograr con la implementación del sistema. - Describen las razones de porque un proyecto debe ser iniciado y las necesidades de la organización como un todo, y no de grupos o stakeholders específicos. - Permiten capturar la visión de la organización para la definición del alcance del sistema, estimación de costo y tiempo requeridos para su implementación.
Requerimientos de Stakeholders	REQST	<ul style="list-style-type: none"> - Son declaraciones de las necesidades de un stakeholder o tipo de stakeholder específico. - Describen la manera en que un stakeholder va a interactuar con la solución, así como las tareas que debe ser capaz de realizar con el sistema. - Sirven como un puente entre los requerimientos de negocio y los requerimientos de la solución.
Requerimientos de la Solución	REQSO	<ul style="list-style-type: none"> - Describen las características que una solución debe poseer para cumplir los requerimientos de negocio y de los stakeholders. Se clasifican en funcionales y No Funcionales.
Requerimientos Funcionales	REQFU	<ul style="list-style-type: none"> - Describen las características que los desarrolladores deben implementar en uno o más componentes del sistema para cumplir con los requerimientos de los stakeholders. - Describen el comportamiento, operaciones y la información (entradas - salidas) que un sistema debe tener y manipular bajo condiciones específicas (sin considerar restricciones físicas).
Requerimientos no Funcionales	REQNF	<ul style="list-style-type: none"> - Describen propiedades que un sistema debe exhibir o restricciones del entorno que debe respetar, y que no tienen que ver con su comportamiento o funcionalidad. - Se aplican al sistema como un todo y no a sus características individuales.
Atributos de Calidad	REQAC	<ul style="list-style-type: none"> - Son las propiedades de calidad de un sistema, tales como: confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

Continuación de la **Tabla 23**.

Tipo	Sigla	Definición
Interfaces Externas	REQIE	- Describen las conexiones entre un sistema de software, un usuario, otro sistema o un dispositivo de hardware.
Restricciones	REQRE	- Imposiciones sobre el diseño y construcción de un sistema, que deben ser respetadas por el desarrollador. - Puede incluir fecha de entrega, especificaciones de hardware, limitaciones, especificaciones de software, colores de preferencia, uso de gráficos y logos, entre otros.

4.1.2. Determinar atributos de requerimientos

- **Obtener un listado de atributos de cada tipo de requerimiento, en base a la revisión bibliográfica y a proyectos realizados en la Dirección de TIC:** En el Capítulo II de Marco Referencial, se pueden encontrar referencias a algunos atributos comunes de requerimientos de software encontrados en las fuentes bibliográficas, que servirán para determinar los atributos a usar.

En cuanto a los proyectos realizados en la Dirección de TIC, se ha revisado documentación de procedimientos y pedidos de requerimientos de software existentes, de los cuales se han obtenido atributos de los formatos de “Historia de Usuario” y “Orden de Mantenimiento y Pedidos varios”. Cabe aclarar que el formato de “Historia de Usuario” fue una propuesta de un estudio anterior, pero no ha sido utilizado en la Unidad de Desarrollo de Software.

- **Historia de Usuario**

ID de historia, Fecha de creación, Proyecto, Nombre de la historia, Dirección requirente, Unidad requirente, Prioridad, Iteración asignada, Funcionario requirente y Cargo, Estimación de la historia, Analista programador responsable, Descripción de la historia, Pruebas de aceptación y Observaciones.

- **Orden de Mantenimiento y Pedidos varios:**

Sección del requerimiento: Número de Orden, Fecha, Nombre y Cargo del solicitante, Tipo de solicitante, Dirección y Unidad solicitante, Descripción del requerimiento, Firma y sello del Director departamental solicitante, Firma del solicitante, Firma de aprobación del Director de TIC.

Sección de atención del requerimiento: Factibilidad del requerimiento, Nombre de la aplicación, Descripción del trabajo realizado, Fecha de inicio, Fecha de terminación, Fecha de inicio de pruebas, Fecha de fin de pruebas, Observaciones durante el período de pruebas, Fecha de inicio de producción, Constancia de entrega del requerimiento (Nombre y Firma de: Programador, Solicitante y Responsable de la Unidad de Desarrollo de Software).

- **Seleccionar los atributos que se van a usar para cada tipo de requerimiento, en coordinación con los funcionarios de la Dirección de TIC. Colocar una descripción para cada atributo.**

Para el presente estudio, se definirán atributos para los requerimientos de la solución, es decir funcionales y no funcionales. Los requerimientos del negocio se van a declarar en el documento de proyecto de desarrollo y de software, y los requerimientos de los stakeholders serán la base para el establecimiento de los requerimientos de la solución.

Tomando como base los resultados de la tarea anterior, se han seleccionado y distribuido atributos para cada tipo de requerimiento a ser usado en la Dirección de TIC, mostrados en la **Tabla 24**.

Es importante señalar que algunos atributos de requerimientos no se han considerado en este punto, ya que pertenecen a etapas posteriores del procedimiento de gestión de requerimientos, como es el caso de los atributos relacionados a control de cambios, pruebas, constancia de entrega, y que serán descritos más adelante en el formato de cada artefacto.

Tabla 24: Atributos de requerimientos a ser usados en la Dirección de TIC

Fuente: Elaborado por la autora

Atributo	Requerimientos de la Solución			
	Funcionales	No Funcionales		
		Atributos de Calidad	Interfaces Externas	Restricciones
Código	X	X	X	X
Nombre corto	X	X	X	X
Descripción	X	X	X	X
Tipo	X	X	X	X
Fecha de creación	X	X	X	X
Versión	X	X	X	X
Solicitante	X	X	X	X
Base Legal	X		X	X
Levantado por	X	X	X	X
Prioridad	X		X	X
Complejidad	X		X	X
Estimación	X		X	
De Gestión				
Estado	X	X	X	X
Responsable implementación	X	X	X	X
Módulo	X		X	

En la **Tabla 25** se colocan las descripciones para cada uno de los atributos de requerimientos a ser usados en la Dirección de TIC.

Tabla 25: Descripciones de atributos de requerimientos a ser usados en la Dirección de TIC
Fuente: Elaborado por la autora

Atributo	Descripción
Generales	
Código	- Código único que identifica al requerimiento. - Este código no puede cambiar, ni se debe reutilizar en caso de eliminación de un requerimiento.
Nombre corto	Nombre corto que identifica al requerimiento.
Descripción	Declaración textual completa del requerimiento.
Tipo	Tipo de requerimiento basado en la taxonomía definida para la Dirección de TIC.
Fecha de Creación	Fecha de levantamiento del requerimiento.
Versión	Determinar el número de versión de acuerdo al procedimiento de control de versiones establecido.
Solicitante	Stakeholder que declara la necesidad del requerimiento.
Base Legal	Referencia a la base legal que contiene las reglas de negocio que definen o restringen el requerimiento. Se debe hacer referencia al nombre del documento de la base legal, título, sección, capítulo, artículo, entre otros.
Levantado por	Analista de Sistemas que elicitó y documentó el requerimiento.
Prioridad	Alta, Media, Baja
Complejidad	El Analista de Requerimientos debe determinar el nivel de complejidad de implementación del requerimiento: Alta, Media, Baja.
Estimación	El Analista de Requerimientos debe estimar el número de horas de trabajo que le tomará la implementación del requerimiento.
De Gestión	
Estado	Permite conocer el avance del requerimiento en su ciclo de vida. Puede tener uno de los siguientes valores: Levantado, Analizado, Documentado, Validado, Diseñado, Implementado, Probado, Implantado, Cancelado.
Responsable implementación	Nombre del Analista de Sistemas responsable de la implementación del requerimiento.
Módulo	Componente de software al cual el requerimiento ha sido asignado.

4.1.3. Determinar tipos de stakeholders

- **Obtener un listado de tipos de stakeholders, en base a la revisión bibliográfica y a los proyectos realizados en la Dirección de TIC**

En el Capítulo II de Marco Referencial, se detallan las referencias a tipos de stakeholders encontrados en las fuentes bibliográficas a ser usadas en el presente estudio. Además se ha revisado documentación de procedimientos y pedidos de requerimientos de software existentes en la Dirección de TIC, de los cuales se han obtenido stakeholders que están relacionados a:

- **Usuarios solicitantes:** Funcionario solicitante, Director Departamental de funcionario solicitante. Se debe recalcar que en los requerimientos existentes se diferencian usuarios internos y externos, en el caso de externos se consideran a personal de empresas públicas municipales y empresas proveedoras de servicios.
- **Miembros del equipo de desarrollo:** Analista de Sistemas, Responsable de la Unidad de Desarrollo de Software, Director de Tecnologías de la Información y Comunicación.
- **Seleccionar los tipos de stakeholders que se van a usar en coordinación con los funcionarios de las Dirección de TIC. Colocar una descripción para cada stakeholder.**

En la **Tabla 26** se detallan los tipos de stakeholders que se van a usar en la Dirección de TIC. Se debe señalar que a futuro se pueden incrementar tipos de stakeholders de acuerdo a la necesidad de cada proyecto.

Tabla 26: Tipos de stakeholders a ser usados en la Dirección de TIC

Fuente: Elaborado por la autora

Tipo Stakeholder	I/E	Descripción	Ejemplo
Usuario	Interno	Funcionario interno del GAD-I que declara requerimientos, aprueba requerimientos, utiliza el sistema.	<ul style="list-style-type: none"> • Funcionarios de Unidades y Direcciones Departamentales del GAD-I.
Usuario	Externo	Funcionario externo del GAD-I que declara requerimientos, aprueba requerimientos, utiliza el sistema.	<ul style="list-style-type: none"> • Funcionarios o empleados de instituciones externas: notarías, instituciones financieras, empresas públicas municipales, otras.
Contribuyente	Externo	Ciudadano que utiliza los sistemas y servicios provistos por el GAD-I.	<ul style="list-style-type: none"> • Ciudadanos.
Consultor	Externo	Empresa externa que presta un bien o servicio a la entidad a través de un contrato.	<ul style="list-style-type: none"> • Empleados y gerente de la empresa consultora.
Patrocinador	Externo	Organización externa que financia todo o parte de un proyecto.	<ul style="list-style-type: none"> • Empleados de la organización: Banco de Desarrollo, Asociación de Municipalidades del Ecuador, otros.
Miembros del equipo de desarrollo	Interno	<ul style="list-style-type: none"> • Funcionarios operativos de la Dirección de TIC que realizan las actividades de análisis, diseño, construcción, pruebas, implantación. • Funcionarios directivos de la Dirección de TIC que realizan actividades de gestión del proyecto. 	<ul style="list-style-type: none"> • Funcionarios de la Unidad de Desarrollo de Software de la Dirección de TIC: Analista de Sistemas, Responsable de Desarrollo de Software, Director de Tecnologías de la Información y Comunicación

4.1.4. Definir el proceso de gestión de requerimientos de software

- **Definir actividades generales del proceso de gestión de requerimientos**

Dentro del proceso de gestión de requerimientos de software de la Dirección de TIC se considerará el desarrollo y mantenimiento de software, tal como se establece en las Normas de Control Interno (Contraloría General del Estado, 2009) en los puntos 410-07 denominado Desarrollo y adquisición de software aplicativo y 410-09 Mantenimiento y control de la infraestructura tecnológica. En ese sentido, se definen los procesos y actividades mostradas en la **Tabla 27** y **Tabla 28**.

- **Desarrollo de software:** se considerará como desarrollo de software a todas las peticiones realizadas por usuarios internos del GAD-I y que involucren la construcción de sistemas y/o módulos informáticos nuevos.

Tabla 27: Actividades definidas para el proceso de desarrollo de software

Fuente: Elaborado por la autora

Actividades del proceso de desarrollo de software	
1. Realizar el despegue del proyecto	1. Realizar reuniones para definición del proyecto de desarrollo de software.
2. Levantamiento y de Análisis de requerimientos	<p>Desarrollo de requerimientos</p> <ol style="list-style-type: none"> 1. Elicitar y Recolectar Requerimientos 2. Analizar requerimientos 3. Documentar requerimientos 4. Validar requerimientos <p>Administración de requerimientos</p> <ol style="list-style-type: none"> 1. Control de Cambios 2. Control de versiones 3. Seguimiento a requerimientos 4. Trazabilidad de requerimientos
3. Diseño	Debido al alcance del presente estudio, no se analizarán en detalle las actividades de diseño, construcción, pruebas e implantación.
4. Construcción	
5. Pruebas	
6. Implantación	
Sin embargo, todas las actividades del ciclo de desarrollo de software dependen y están directamente relacionadas con la actividad de Levantamiento y Análisis de requerimientos.	

- **Mantenimiento de Software:** De acuerdo con (Chemuturi, 2013), existen dos tipos de mantenimiento, uno es la modificación y otro es la expansión de software.
 - **Sub proceso - Modificación de Software:** Abarca la corrección de defectos, modificación de código para cumplir un requerimiento que ha cambiado, o cambios para mejorar la productividad. Debido a que en la Dirección de TIC se atienden peticiones de mantenimiento de software desarrollado internamente y desarrollado por empresas externas, se deben diferenciar las actividades realizadas para cada tipo de software.

Tabla 28: Actividades definidas para el proceso de modificación de software
Fuente: Elaborado por la autora

Actividades – Software desarrollado internamente	Actividades – Software desarrollado por empresas externas
1. Registrar la petición de mantenimiento	1. Registrar la petición de mantenimiento
2. Analizar petición	2. Analizar la petición
3. Replicar escenario en ambiente de pruebas	3. Enviar petición a empresa externa
4. Modificar código	4. Recibir la modificación del software
5. Realizar pruebas	
6. Realizar cambios en ambiente de producción	

- **Sub proceso - Expansión de Funcionalidad:** abarca la adición de funcionalidad a un software existente, lo que implica construir nuevo código en lugar de modificarlo. En este caso, para la Dirección de TIC se utilizará el mismo proceso definido para el desarrollo de software. Dependiendo del alcance de la expansión, el Analista de Sistemas deberá determinar si es necesario realizar el despegue del proyecto, establecido en el proceso de desarrollo.
- **Seleccionar técnicas de elicitación de requerimientos a ser usadas en la Dirección de TIC.**

De acuerdo con el Instituto Internacional de Análisis del Negocio (IIBA, 2009), existen varios factores que guían la selección de una u otra técnica de elicitación, tales como el dominio del negocio, el ambiente y cultura organizacional, las habilidades del analista, y los entregables que deben generarse.

Para el presente estudio se seleccionaron las técnicas de entrevista, observación y análisis del dominio, debido a que en el diagnóstico realizado en la Unidad de Desarrollo de Software se obtuvo como resultado que los Analistas de Sistemas tienen mayor conocimiento y experiencia en estas técnicas, tal como se puede ver en la **Tabla 29**.

En ese sentido, se propondrán procedimientos y formatos que ayuden a los funcionarios durante la elicitación y recolección de requerimientos, mediante el uso de las técnicas seleccionadas. Sin embargo, los Analistas de Sistemas podrán utilizar otras técnicas dependiendo del tipo de proyecto que se encuentren realizando.

Tabla 29: Conocimiento y experiencia en técnicas de elicitación de requerimientos
Fuente: Elaborado por la autora

Técnica	Conocimiento	Experiencia
<i>Entrevista</i>	Todos los Analistas de Sistemas tienen conocimientos teóricos sobre esta técnica.	Todos los Analistas de Sistemas han aplicado al menos una vez esta técnica para levantar requerimientos.
<i>Encuesta</i>	Todos los Analistas de Sistemas tienen conocimientos teóricos sobre esta técnica.	Ninguno de los Analistas de Sistemas ha aplicado esta técnica para levantar de requerimientos.
<i>Análisis de tareas</i>	No todos los Analistas de Sistemas tienen conocimientos teóricos sobre esta técnica.	Ninguno de los Analistas de Sistemas ha aplicado esta técnica para levantar de requerimientos.
<i>Observación</i>	Todos los Analistas de Sistemas tienen conocimientos teóricos sobre esta técnica.	No todos los Analistas de Sistemas han aplicado esta técnica para levantar requerimientos.
<i>Análisis del Dominio</i>	Todos los Analistas de Sistemas tienen conocimientos teóricos sobre esta técnica.	Todos los Analistas de Sistemas han aplicado al menos una vez esta técnica para levantar requerimientos.
<i>Lluvia de ideas</i>	No todos los Analistas de Sistemas tienen conocimientos teóricos sobre esta técnica.	No todos los Analistas de Sistemas han aplicado esta técnica para levantar requerimientos.

- **Determinar las entradas, tareas, responsables y artefactos del proceso de gestión de requerimientos (Elicitación, Análisis, Especificación, Validación)**

Para la definición del proceso de gestión de requerimientos de software adaptado a las necesidades de la Dirección de TIC del GAD-I, se consideraron los factores establecidos en el Instituto Internacional de Análisis del Negocio (IIBA, 2009) y que se listan a continuación:

- **Cultura organizacional:** En organizaciones donde la cultura organizacional no soporta la formalidad, pero la informalidad compromete el producto final, es necesario trabajar con los stakeholders para negociar un proceso apropiado.
- **Preferencias de los stakeholders:** Algunos stakeholders pueden preferir más o menos formalidad, por ejemplo se puede requerir aprobación formal pero no se requiere documentar la elicitación de requerimientos.
- **Complejidad del proyecto o producto:** Procesos formales para la gestión de la configuración y gestión del cambio son usados para:
 - Proyectos que tienen muchas interfaces, mucho negocio, o los impactos del sistema se expanden a muchas áreas funcionales.

- Productos que son construidos con muchos componentes o subcomponentes, que tienen interfaces complejas, que van a ser usados por una variedad de stakeholders, entre otros.
- **Madurez de la Organización:** Organizaciones menos maduras tienden a querer gastar menos tiempo y dinero en la creación de un proceso de requerimientos y se resisten a la idea de tener un proceso para definir requerimientos.
- **Disponibilidad de recursos:** Existencia de recursos necesarios para soportar el esfuerzo de crear un proceso y gestionarlo.

En ese sentido, se realizaron reuniones de trabajo con los funcionarios de la Unidad de Desarrollo de Software con el objetivo de analizar cada uno de los factores indicados como se muestra a continuación:

- **Cultura Organizacional:** Se debe aplicar un nivel de formalidad mayor al actual, ya que esto es muy importante en instituciones que deben cumplir con regulaciones gubernamentales, donde los requerimientos deben ser documentados, rastreados, aprobados y probados.
- **Preferencias de los stakeholders:** Los stakeholders deben adaptarse al proceso establecido por la Dirección de TIC.
- **Complejidad del proyecto o producto:** Se definirá un proceso para el control de versiones y control de cambios que tenga un nivel adecuado de formalidad.
- **Madurez de la organización:** La organización se encuentra madura en el tema de procesos, además la Dirección de TIC participó en un proyecto de levantamiento de procesos del GAD-I en el año 2012 por lo que tiene experiencia y bases previas.
- **Disponibilidad de recursos:** Existe disponibilidad de recursos y predisposición del personal de la Dirección de TIC para mantener y mejorar el proceso de gestión de requerimientos en el tiempo.

Con estos antecedentes, mediante la aplicación de talleres de trabajo con los funcionarios de la Unidad de Desarrollo de Software se definieron las entradas, tareas, responsables y artefactos del proceso de gestión de requerimientos para las actividades de Elicitación, Análisis, Especificación y Validación, los mismos que son mostrados en la **Tabla 30, Tabla 31, Tabla 32, Tabla 33 y Tabla 34.**

Las elicitación y recolección de requerimientos se deben realizar paralelamente durante el levantamiento de requerimientos, tal como se muestra en la **Tabla 31.** Para el presente proceso, se propone el uso de una plantilla para la entrevista estructurada y para la observación, sin embargo el Analista de Sistemas podrá usar entrevistas no estructuradas o semi estructuradas dependiendo de su entendimiento en el negocio.

La técnica de análisis del dominio se usará para la recolección de información y sistemas relacionados, así como para la elaboración del glosario de términos y abreviaturas.

Tabla 30: Entradas, tareas, responsables y artefactos para el despegue del proyecto

Fuente: Elaborado por la autora

Actividad:		Realizar el despegue del proyecto	Entrada:	Petición de desarrollo de software	
N°	Tarea	Descripción		Responsables	Artefactos
1	Planificar las reuniones de despegue	<ul style="list-style-type: none"> - Coordinar las fechas y horas de las reuniones de despegue con la Dirección o Unidad Administrativa que realizó la petición de desarrollo de software. - Ubicar y solicitar los recursos necesarios para las reuniones de despegue, tales como equipos e instalaciones. - Elaborar y enviar las convocatorias para las reuniones de despegue a las partes involucradas. 		Administrador del Proyecto.	Convocatorias a reuniones
2	Ejecutar las reuniones de despegue	Reunirse con los interesados del proyecto para realizar las siguientes tareas:		Administrador del Proyecto. Analista de Sistemas. Stakeholders.	Acta de Reunión
		2.1 Determinar el problema y objetivos del proyecto: solicitar al stakeholder que realizó la petición, que explique el problema que tiene en su área de trabajo y los objetivos (requerimientos) de negocio que quiere alcanzar con el proyecto.			
		2.2 Determinar el alcance del proyecto: mediante la elaboración de un diagrama de bloques, definir el listado de funcionalidades que serán parte del sistema.			
		2.3 Identificar stakeholders del proyecto: <ul style="list-style-type: none"> - Reconocer a todas las personas, áreas u organizaciones que se ven impactadas de forma directa o indirecta por la ejecución del proyecto. - Documentar información relacionada a cada uno de ellos: nombre, cargo, departamento, tipo, interno o externo, contactos. - Identificar los stakeholders que tienen autoridad para: aprobar entregables, revisar y aprobar requerimientos, pedir y aprobar cambios, aprobar o rechazar soluciones propuestas. - Determinar las responsabilidades de cada stakeholder dentro del proyecto. 			Registro de Stakeholders
3	Documentar el proyecto de software	Elaborar el documento de Proyecto de Software en el que se documentará la descripción, alcance, objetivos, justificación, responsables y recursos del sistema.		Administrador del Proyecto. Analista de Sistemas.	Proyecto de Desarrollo de Software

4	Solicitar aprobación del proyecto	Solicitar la revisión del proyecto de software a los stakeholders. Realizar las correcciones necesarias y solicitar la aprobación de las partes interesadas.	Administrador del Proyecto.	Proyecto de Desarrollo de Software aprobado
Salida:		Proyecto de Desarrollo de Software aprobado		

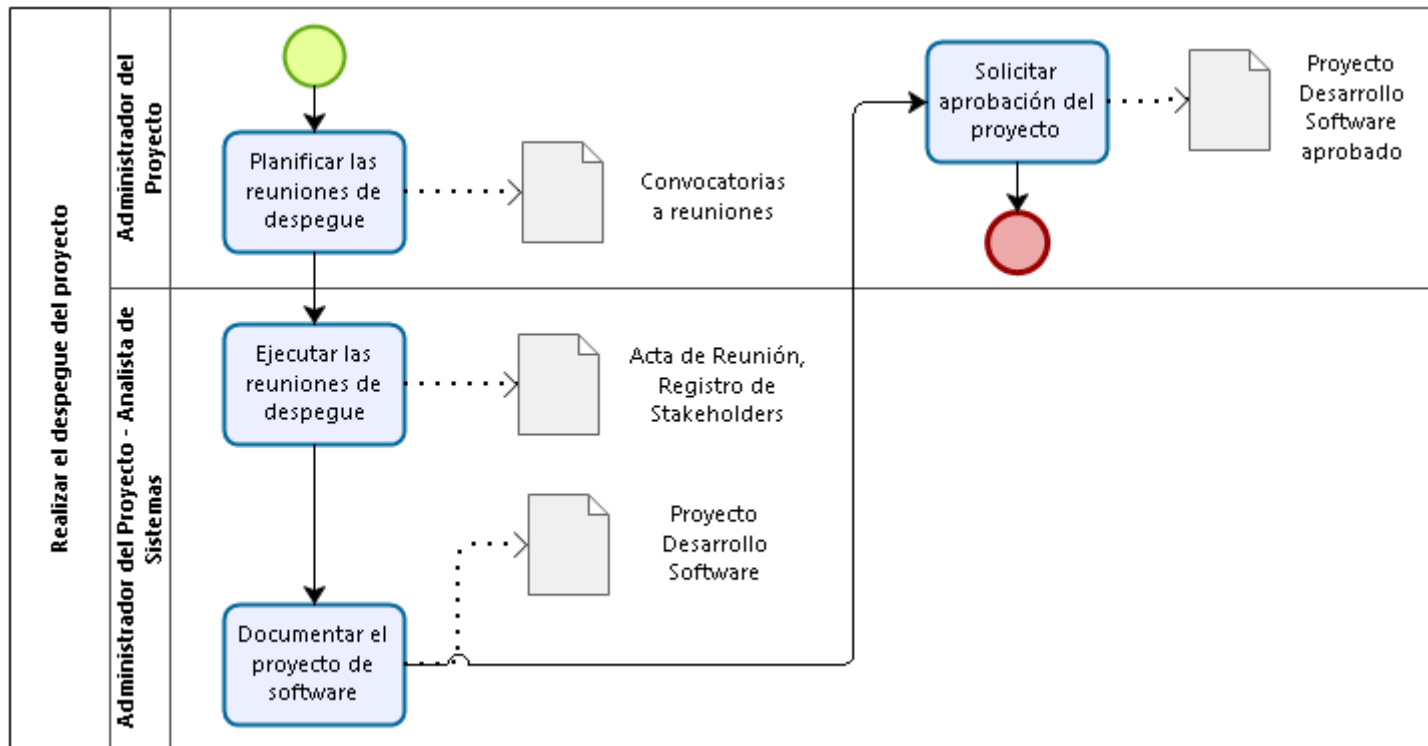


Figura 16: Diagrama - Realizar despegue del proyecto
Fuente: Elaborado por la autora

Tabla 31: Entradas, tareas, responsables y artefactos para la elicitación de requerimientos

Fuente: Elaborado por la autora

Actividad:		Elicitar y Recolectar Requerimientos	Entradas:	Proyecto de Desarrollo de Software aprobado	
N°	Tarea	Descripción		Responsables	Artefactos
1	Elicitar	Recopilar información de las necesidades de los stakeholders del proyecto.			
1.1	Preparar la logística de la elicitación	<p>Asegurar que todos los recursos necesarios estén disponibles, organizados y programados, mediante los siguientes pasos:</p> <ul style="list-style-type: none"> - Estudiar la información preliminar para entender el dominio: si se tiene conocimiento del dominio se puede saltar este paso. - Hacer un listado de los stakeholders que van a participar en la elicitación. - Elaborar una agenda de temas a tratar y de actividades a realizar en cada elicitación de requerimientos. - Acordar fecha y hora con cada stakeholder o grupo de stakeholders para realizar la elicitación. - Programar los recursos necesarios (logística) para cada reunión de elicitación: equipos requeridos, instalaciones, pizarras. - Comunicar la programación de elicitación a todos los stakeholders, con el objetivo de que se preparen para la misma. La programación de elicitación debe contener fecha, hora, lugar, nombres de los participantes, temas y actividades a tratar. 		Analista de Sistemas	Convocatoria a reuniones
1.2	Preparar la elicitación	<p>Preparar los formatos requeridos para levantamiento de información durante la elicitación, de acuerdo a las técnicas que se vayan a usar:</p> <ul style="list-style-type: none"> - Entrevista: preparar preguntas que permitan obtener información sobre los procesos, tal como el disparador, entradas, salidas, cálculos y validaciones a realizar. - Observación: realizar revisiones previas sobre el entorno en el que se va a ejecutar el sistema, tales como información y sistemas relacionados. 		Analista de Sistemas	Preguntas para la entrevista.

1.3	Ejecutar la elicitación	Reunirse con los stakeholders de acuerdo a la programación realizada. Aplicar las técnicas de elicitación para capturar información de las entradas, salidas, pasos, ítems de datos y formularios usados en todos los procesos ejecutados por el stakeholder. Esto se debe documentar en el Formato de entrevista u observación para levantar información.	Analista de Sistemas	Formato de entrevista para levantar información. Formato de observación para levantar información.
1.3	Documentar los resultados de la elicitación	Elaborar un resumen de todo lo ocurrido durante la elicitación (requerimientos de usuario). Almacenar los registros visuales o de audio que se hayan recopilado durante las reuniones.	Analista de Sistemas	Actas de reuniones
2	Recolectar	Recopilar información procedente de documentos, aplicaciones existentes, estándares y guías, mediante las siguientes tareas:		
2.1	Recopilar, revisar y organizar información relacionada	<ul style="list-style-type: none"> - Registros organizacionales técnicos: información para el diseño de la base de datos, longitudes máximas para los campos, tablas existentes, estándares de desarrollo de software. - Normativa Legal: legislación que soporta el sistema como leyes, ordenanzas, reglamentos, políticas, resoluciones, entre otros. - Documentación de procesos: estructura orgánica funcional, mapa de procesos, procedimientos, instructivos, estándares, guías. - Información histórica: estudios previos, encuestas de satisfacción al cliente (para identificar ineficiencias en los procesos), quejas del cliente (para identificar mejoras). 	Analista de Sistemas	Registro de información relacionada
2.2	Determinar los sistemas relacionados	Determinar los sistemas que tengan relación con el nuevo sistema. Recopilar las descripciones funcionales y técnicas de los sistemas relacionados.	Analista de Sistemas	Registro de sistemas relacionados
2.3	Realizar un glosario de términos y abreviaturas para el proyecto	Acordar las definiciones de palabras a ser usadas para manejar un solo lenguaje entre todos los stakeholders. Este glosario se va a ir formando a lo largo del ciclo de vida de desarrollo.	Analista de Sistemas. Stakeholders.	Glosario de términos y abreviaturas
Salidas:		Actas de reuniones, Formatos de entrevistas u observaciones (llenos), Registro de información relacionada, Registro de sistemas relacionados, Glosario de términos y abreviaturas		

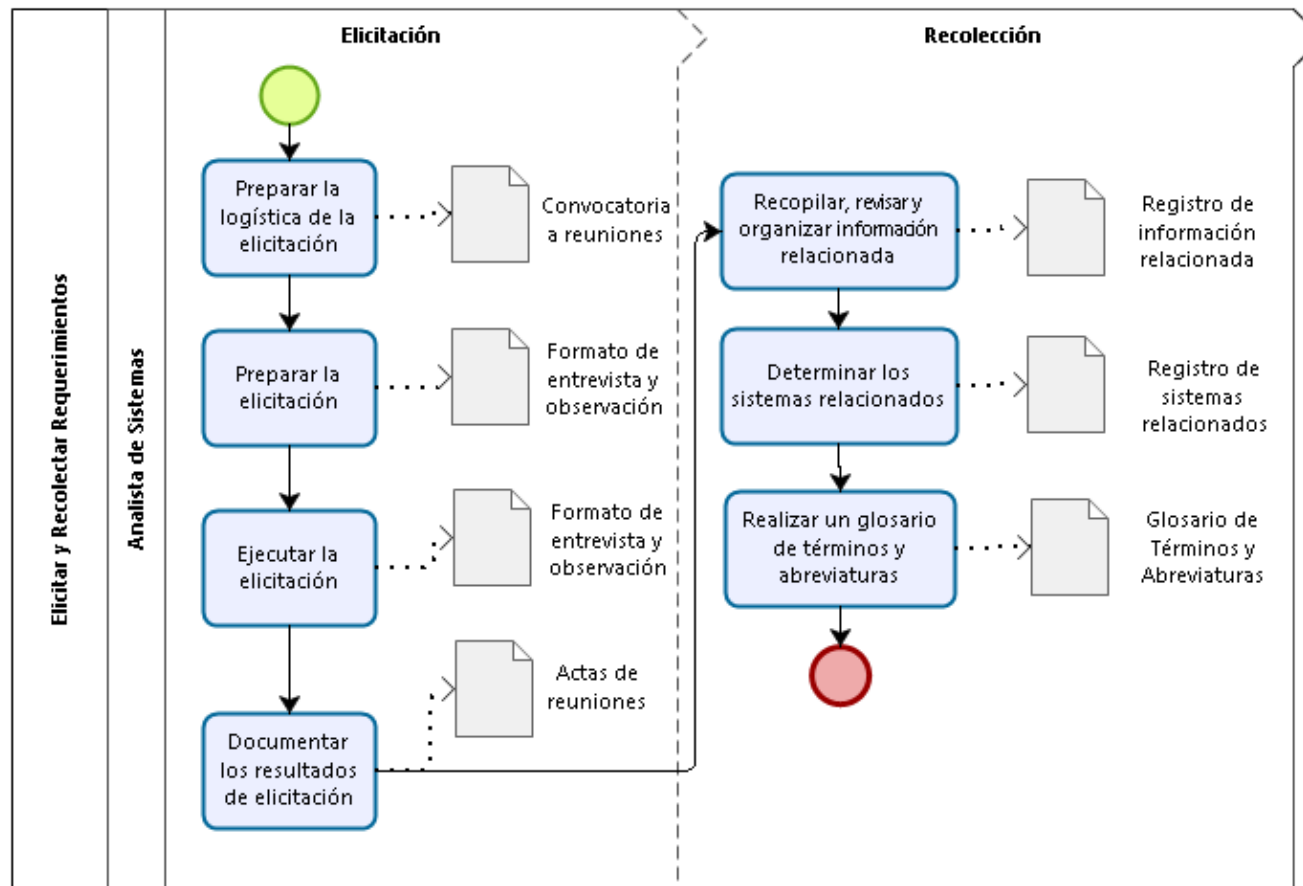


Figura 17: Diagrama - Elicitar y Recolectar Requerimientos
Fuente: Elaborado por la autora

Tabla 32: Entradas, tareas, responsables y artefactos para el análisis de requerimientos

Fuente: Elaborado por la autora

Actividad:		Entradas:		
N°	Tarea	Descripción	Responsables	Artefactos
	Analizar requerimientos		Actas de reuniones, Formatos de entrevistas u observaciones (llenos), Registro de información relacionada, Registro de sistemas relacionados, Glosario de términos y abreviaturas, Formularios usados en los procesos.	
1	Identificar requerimientos y clasificarlos	Revisar la información levantada en la actividad anterior y registrar los requerimientos. Determinar si los requerimientos se duplican o se contradicen entre sí. En caso de que lo último suceda, se debe acudir al stakeholder para aclarar las inconsistencias.	Analista de Sistemas	Registro de Requerimientos – Análisis
2	Evaluar factibilidad de cada requerimiento	Mediante una reunión con los stakeholders, evaluar la factibilidad técnica, económica y de tiempo de cada requerimiento: <ul style="list-style-type: none"> - Técnica: Se refiere a que la implementación del requerimiento no está limitada por hardware, software, el estado actual de la tecnología o el estado actual de la organización en tecnología. - Económica: Se refiere al que la implementación del requerimiento no está limitada por el presupuesto. - De tiempo: Se refiere a que la implementación del requerimiento es factible dentro del tiempo disponible. <p>La participación los stakeholders es importante ya que es necesario tomar decisiones en el caso de que un requerimiento no sea factible, lo que se deberá dejar constancia en el Acta de Reunión.</p>	Analista de Sistemas. Stakeholders	Acta de Reunión
3	Priorizar los requerimientos	Colocar el orden en el cual se van a implementar los requerimientos de acuerdo a la necesidad del stakeholder. El orden se determinará mediante la asignación de prioridad Alta, Media o Baja.	Analista de Sistemas. Stakeholders	Registro de Requerimientos – Análisis

4	Estimar tiempo de implementación de cada requerimiento	Estimar complejidad de cada requerimiento uno de acuerdo a la escala Alta, Media o Baja; además estimar tiempo de implementación en horas de cada uno.	Analista de Sistemas.	Registro de Requerimientos – Análisis
Salidas		Registro de Requerimientos – Análisis, Acta de Reunión		

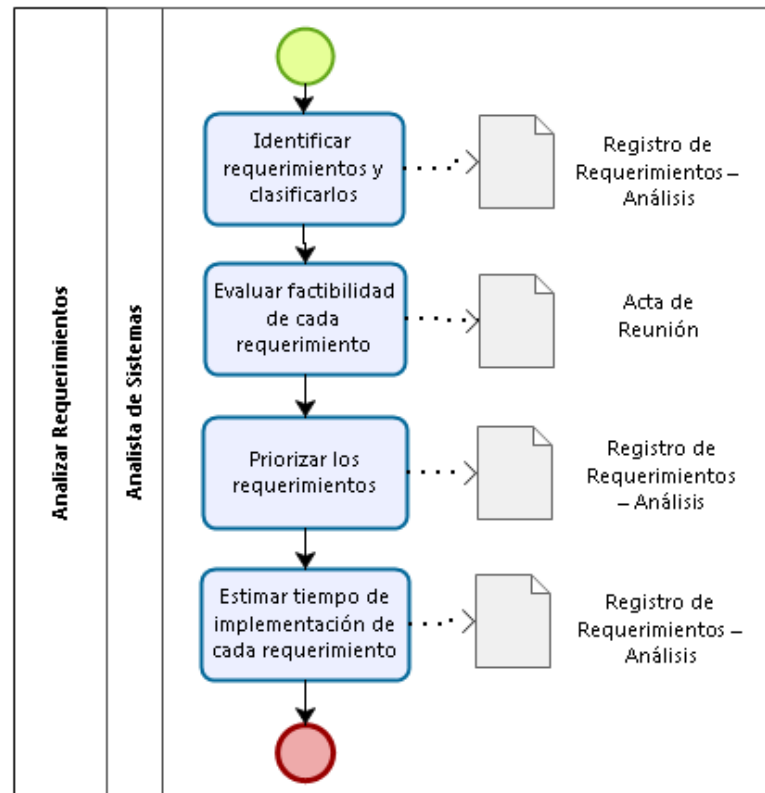


Figura 18: Diagrama - Analizar Requerimientos
Fuente: Elaborado por la autora

Tabla 33: Entradas, responsables y artefactos para la especificación de requerimientos

Fuente: Elaborado por la autora

Actividad:		Entradas:			
Especificar requerimientos			Registro de Requerimientos – Análisis, Formato de entrevista y observación para levantar información (llenos), Formularios usados en los procesos, Actas de reuniones.		
N°	Tarea	Descripción		Responsables	Artefactos
1	Registrar Base Legal de los requerimientos	Registrar las referencias a la base legal que contiene las reglas de negocio que definen o restringen el requerimiento. Se debe hacer referencia al nombre del documento de la base legal, título, sección, capítulo, artículo, entre otros.		Analista de Sistemas	Registro de Requerimientos – Especificación
2	Registrar información complementaria de cada requerimiento	Registrar la siguiente información complementaria de cada requerimiento funcional: <ul style="list-style-type: none"> - Entradas - Validaciones para los datos de entrada - Proceso para convertir entradas en salidas - Actores: primarios (ejecutan el proceso) o secundarios (proveen de entradas al proceso o usan las salidas del proceso) - Las salidas del proceso - Plantillas y formatos usados en el proceso 		Analista de Sistemas	Registro de Requerimientos – Especificación
3	Distribuir requerimientos y designar responsable de implementación	Distribuir los requerimientos en componentes lógicos del sistema, es decir módulos y sub módulos. Con la ayuda del Administrador del Proyecto, designar al Analista de Sistemas responsable de su implementación.		Analista de Sistemas. Administrador del Proyecto.	Registro de Requerimientos – Especificación
Salidas		Registro de Requerimiento - Especificación			

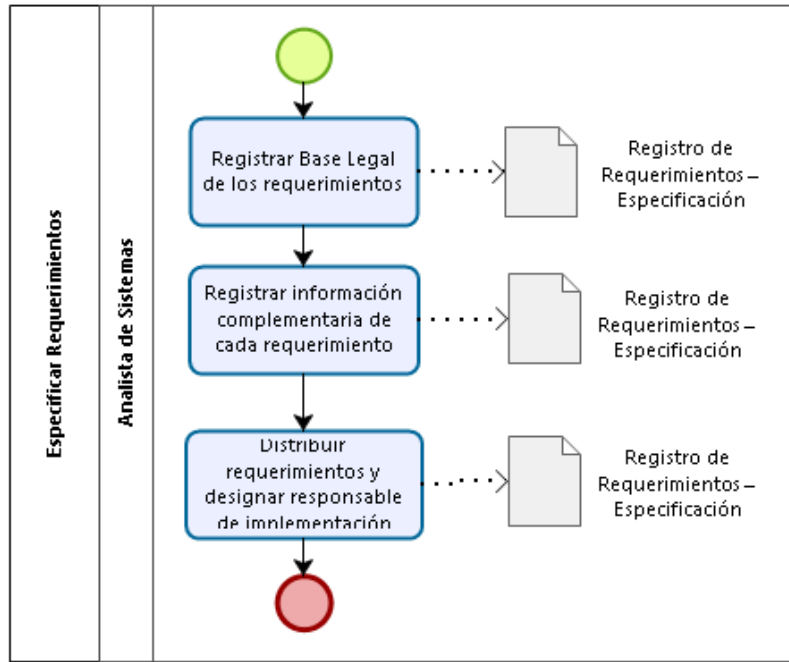


Figura 19: Diagrama - Especificar Requerimientos
Fuente: Elaborado por la autora

Tabla 34: Entradas, tareas, responsables y artefactos para la validación de requerimientos
Fuente: Elaborado por la autora

Actividad:	Validar y Verificar requerimientos		Entradas:	Registro de Requerimientos	
Nº	Tarea	Descripción	Responsables	Artefactos	
1	Preparar la reunión para validar y verificar requerimientos	Determinar cuándo y dónde va a ser la reunión, quién va a participar, el material que se va a revisar y el tiempo requerido. Previo a la reunión, entregar el material a los participantes para su revisión.	Analista de Sistemas	Convocatoria a reunión.	
2	Ejecutar la validación y verificación de requerimientos	Realizar la reunión de inspección en la que se deberá leer cada requerimiento, solicitar a los participantes que indiquen los problemas o errores encontrados y resolverlos.	Analista de Sistemas. Stakeholders	Registro de Requerimientos	
3	Corregir los requerimientos.	Corregir los requerimientos de acuerdo a las observaciones realizadas por los stakeholders.	Analista de Sistemas	Registro de Requerimientos	
4	Gestionar la aprobación de los requerimientos.	Gestionar la aprobación de los requerimientos por parte de los stakeholders que tengan autoridad para ello. Se debe registrar la firma en un documento impreso.	Analista de Sistemas. Stakeholders	Registro de Requerimientos - Validación	
Salidas		Requerimientos aprobados			

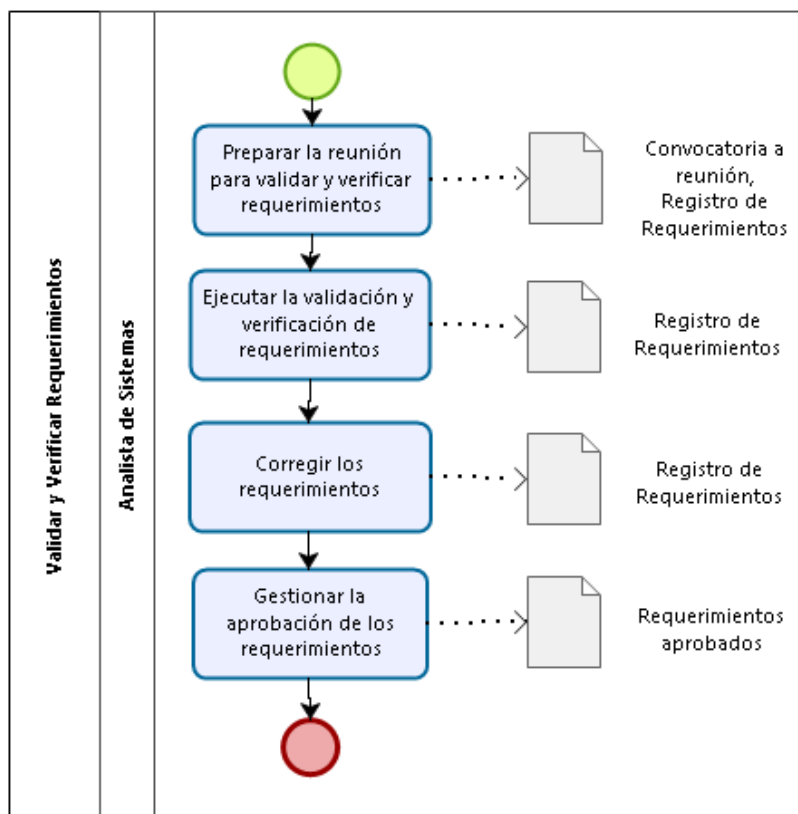


Figura 20: Diagrama - Validar y Verificar Requerimientos
Fuente: Elaborado por la autora

4.1.5. Definir artefactos y proceso para el control de versiones

- **Identificar los artefactos que van a formar parte del control de versiones**

Para la selección de los artefactos que formarán parte del control de versiones, se realizó una reunión con los funcionarios de la Unidad de Desarrollo de Software en la que se analizó la importancia del versionamiento de cada artefacto, teniéndose como resultado que se llevará control de versiones del “Proyecto de Desarrollo de Software” y del “Registro de Requerimientos”, tal como se muestra en la **Tabla 35**.

Tabla 35: Análisis de la importancia de control de versiones en artefactos del proceso
Fuente: Elaborado por la autora

Artefacto	Justificación	Control de versiones
Petición de desarrollo de software	Documento que tiene una sola versión.	NO
Convocatoria a reunión	Documento que se genera una sola vez por cada reunión y no se actualiza con el tiempo.	NO
Acta de Reunión	Documento que se genera una sola vez por cada reunión y no se actualiza con el tiempo.	NO
Registro de Stakeholders	El registro de stakeholders se va actualizando con el tiempo, sin embargo no es necesario mantener un historial de sus versiones.	NO

Continuación de la **Tabla 35**.

Artefacto	Justificación	Control de versiones
Proyecto de Desarrollo Software	Es importante mantener un versionamiento de este artefacto debido a que almacena información sobre el alcance, objetivos, y descripción del proyecto. Cada cambio a este documento debe ser registrado en un historial de versiones.	SI
Registro de información relacionada	El registro de información relacionada se va actualizando con el tiempo, sin embargo no es necesario mantener un historial de sus versiones.	NO
Registro de sistemas relacionados	El registro de sistemas relacionados se va actualizando con el tiempo, sin embargo no es necesario mantener un historial de sus versiones.	NO
Glosario de términos y abreviaturas	El glosario se va actualizando con el tiempo, sin embargo no es necesario mantener un historial de sus versiones.	NO
Convocatoria a reunión con programación de elicitación	Documento que se genera una sola vez por cada reunión y no se actualiza con el tiempo.	NO
Formato de entrevista para levantar información.	Este formato se llena durante las entrevistas y no necesita control de versiones.	NO
Formato de observación	Este formato se llena durante las entrevistas y no necesita control de versiones.	NO
Registro de Requerimientos	Este artefacto mantiene información sobre los requerimientos funcionales, y que están sujetos a cambios en el tiempo, por lo tanto se debe mantener un historial de todas las versiones del mismo. Incluye la información levantada en el Registro de Requerimientos durante el Análisis, Especificación y Validación.	SI
Orden de Mantenimiento	Este artefacto mantiene información sobre las solicitudes de mantenimiento de sistemas, así como su análisis y atención. No es cambiante en el tiempo, por lo tanto no requiere versionamiento.	NO

- **Definir las tareas, responsables y artefactos del control de versiones**

Las tareas mostradas en la **Tabla 36** son aplicables cuando se modifica o agrega un artefacto; en el caso de eliminación, el artefacto se quedará con la última versión aprobada. Para la definición de las tareas, se realizó reuniones de trabajo en las que se determinó que se debía manejar un sistema de versionamiento simple, debido a que anteriormente no se ha usado control de versiones en el proceso de desarrollo de software.

Tabla 36: Tareas, responsables y artefactos del control de versiones

Fuente: Elaborado por la autora

Actividad:		Control de versiones	Entradas:	Proyecto de Desarrollo de Software, Registro de Requerimientos	
Nº	Tarea	Descripción		Responsable	Artefacto
1	Ingresar artefacto a la línea base	Una vez que el artefacto ha sido aprobado, se colocará en su atributo de Versión el número 1.		Analista de Sistemas	N/A
2	Realizar control de versiones	Cada vez que una petición de cambio altere un artefacto sometido a control, se deberán realizar las siguientes tareas: <ul style="list-style-type: none"> - Guardar la última versión del artefacto. - Modificar el artefacto. - Realizar control de calidad del artefacto (verificación y validación). - Corregir los defectos encontrados. - Aprobar el nuevo artefacto. - Identificar la nueva versión del artefacto aumentando en 1 al número de versión. - Guardar la nueva versión del artefacto. - Registrar el historial de cambios. 		Analista de Sistemas	Historial de cambios
Salidas:		Proyecto de Desarrollo de Software y Registro de Requerimientos con control de versiones			

4.1.6. Definir artefactos y proceso para mantener trazabilidad

- **Definir las relaciones de trazabilidad que se usarán**

Se ha educado a los funcionarios sobre los conceptos e importancia de usar trazabilidad en el proceso de desarrollo de software. Debido a que se va a iniciar el uso de trazabilidad en los proyectos de desarrollo de la Unidad y para no sobrecargar a los Analistas de Sistemas con registro de información, se ha decidido conjuntamente con ellos que se usará únicamente la siguiente relación de trazabilidad:

- De Requerimientos Funcionales a Casos de Prueba

En ese sentido, se ha definido también un artefacto de Caso de Prueba con el fin de registrar los resultados de las pruebas realizadas a cada uno de los requerimientos funcionales, el mismo que se puede observar en la **Tabla 51**.

- **Definir las tareas, responsables y artefactos para mantener trazabilidad**

Para cumplir esta actividad, se realizaron reuniones de trabajo con los funcionarios de la Unidad en las que se definieron las tareas, responsables y artefactos mostrados en la **Tabla 37**, en base a la revisión de la literatura y a las relaciones de trazabilidad seleccionadas en el punto anterior.

Tabla 37: Tareas, responsables y artefactos para mantener trazabilidad

Fuente: Elaborado por la autora

Actividad:		Trazabilidad de requerimientos		Entradas:	Artefactos para aplicar trazabilidad	
Nº	Tarea	Descripción		Responsable	Artefacto	
1	Crear trazabilidad de Requerimientos Funcionales a Elementos de Diseño	Crear la relación de trazabilidad cuando se ha completado el diseño de software para cualquier módulo.		Analista de Sistemas	Matriz de trazabilidad	
2	Crear trazabilidad de Requerimientos Funcionales a Código Fuente	Crear la relación de trazabilidad cuando se ha finalizado el código para cualquier componente.		Analista de Sistemas	Matriz de trazabilidad	
3	Crear trazabilidad de Requerimientos Funcionales a Casos de Prueba	Crear la relación de trazabilidad cuando se ha finalizado las pruebas unitarias o de integración de un componente o módulo.		Analista de Sistemas	Matriz de trazabilidad	
4	Crear trazabilidad de Peticiones de Cambio a Requerimientos Funcionales	Crear la relación de trazabilidad cuando una petición de cambio ha sido implementada.		Analista de Sistemas	Matriz de trazabilidad	
5	Controlar la información de trazabilidad periódicamente	Controlar el registro de la trazabilidad para asegurar que todos los requerimientos han sido cubiertos.		Administrador del proyecto. Analista de Sistemas	Matriz de trazabilidad	
Salidas:		Matriz con relaciones de trazabilidad creadas				

4.1.7. Definir artefactos y proceso para el control de cambios

- **Definir las tareas, responsables y artefactos del control de cambios**

Las tareas mostradas en la **Tabla 38** son aplicables cuando se desea añadir, modificar o eliminar un requerimiento, posterior al establecimiento de una línea base. Para la definición de las tareas, se realizaron reuniones de trabajo con los funcionarios en las que se determinó la importancia de registrar y gestionar las peticiones de cambio.

La importancia del control de cambios radica en la responsabilidad de la Unidad de Desarrollo de Software para rastrear el origen de un cambio en los sistemas informáticos, especialmente en aquellos relacionados con manejo económico, aplicación de leyes y reglamentos, cálculos, entre otros.

- **Definir un diagrama de transición de estados para el ciclo de vida de las peticiones de cambio**

En la **Figura 21** se muestra el diagrama de transición de estados definido para las peticiones de cambio del presente estudio. La definición de los estados se la realizó en base a las actividades del proceso de control de cambios establecido anteriormente.

Cabe aclarar que los estados “Registrado”, “En análisis” y “Aprobado” implican que la petición de cambios está abierta; por otro lado los estados “Rechazado”, “Cancelado” e “Implementado” van seguidos inmediatamente del estado “Cerrado”.

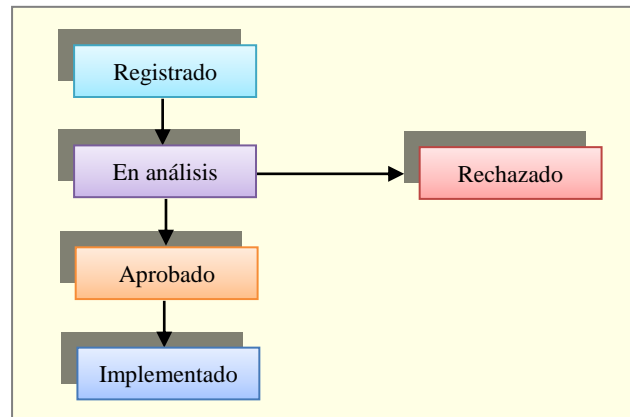


Figura 21: Diagrama de transición de estados de la petición de cambio

Fuente: Elaborado por la autora

Tabla 38: Tareas, responsables y artefactos del control de cambios

Fuente: Elaborado por la autora

Actividad:		Control de cambios	Entrada:	Petición de cambio	
Nº	Tarea	Descripción		Responsable	Artefacto
1	Realizar el registro de la petición de cambio	El solicitante debe registrar la petición de cambio en el formato establecido.		Stakeholder	Petición de cambio [Estado: Registrada]
2	Realizar análisis y evaluación de la petición de cambio	Analizar, en coordinación con los stakeholders involucrados, los siguientes aspectos de la petición de cambio: - Factibilidad técnica y económica. - Cantidad de esfuerzo y tiempo requeridos. - Impacto en el proyecto: cambios en el cronograma, cambios en el presupuesto, cambios al alcance de la funcionalidad, modificación de artefactos (especificación de requerimientos, documento de diseño, pruebas, código), calidad del software, tareas requeridas, recursos requeridos.		Analista de Sistemas. Administrador del proyecto.	Petición de cambio [Estado: En Análisis]

Continuación de la **Tabla 38**.

Actividad:		Control de cambios	Entrada:	Petición de cambio	
N°	Tarea	Descripción		Responsable	Artefacto
3	Tomar una decisión sobre la petición de cambio	Con el análisis realizado, se debe aprobar o rechazar la petición de cambio. Si se aprueba, continúa con el siguiente paso; caso contrario finaliza.		Administrador del proyecto	Petición de cambio [Estado: Aprobada, Rechazada]
4	Implementar la petición de cambio	Realizar las acciones necesarias para implementar la petición del cambio, tales como: - Realizar cambios en el cronograma. - Realizar cambios en el presupuesto. - Actualizar artefactos afectados por el cambio haciendo uso de la información de trazabilidad y el procedimiento de control de versiones. - Comunicar los cambios a todos los miembros del equipo del proyecto.		Analista de Sistemas	Petición de cambio [Estado: Implementada]
Salidas:		Petición de cambio [Estado: Rechazada, Implementada]			

- **Diagramar el proceso de control de cambios**

En la **Figura 22** se puede observar el diagrama del proceso de control de cambios de requerimientos definido para la Dirección de TIC.

4.1.8. Definir el proceso para seguimiento de requerimientos

- **Definir un diagrama de transición de estados para el ciclo de vida de requerimientos funcionales.**

En la **Tabla 39** se describen los estados para los requerimientos funcionales de las diferentes actividades del proceso de desarrollo de software de la Dirección de TIC del GAD-I. En la **Figura 23** se muestra el diagrama de transición de estados para los requerimientos funcionales, considerando que un requerimiento puede ser Cancelado debido a una petición de cambio.

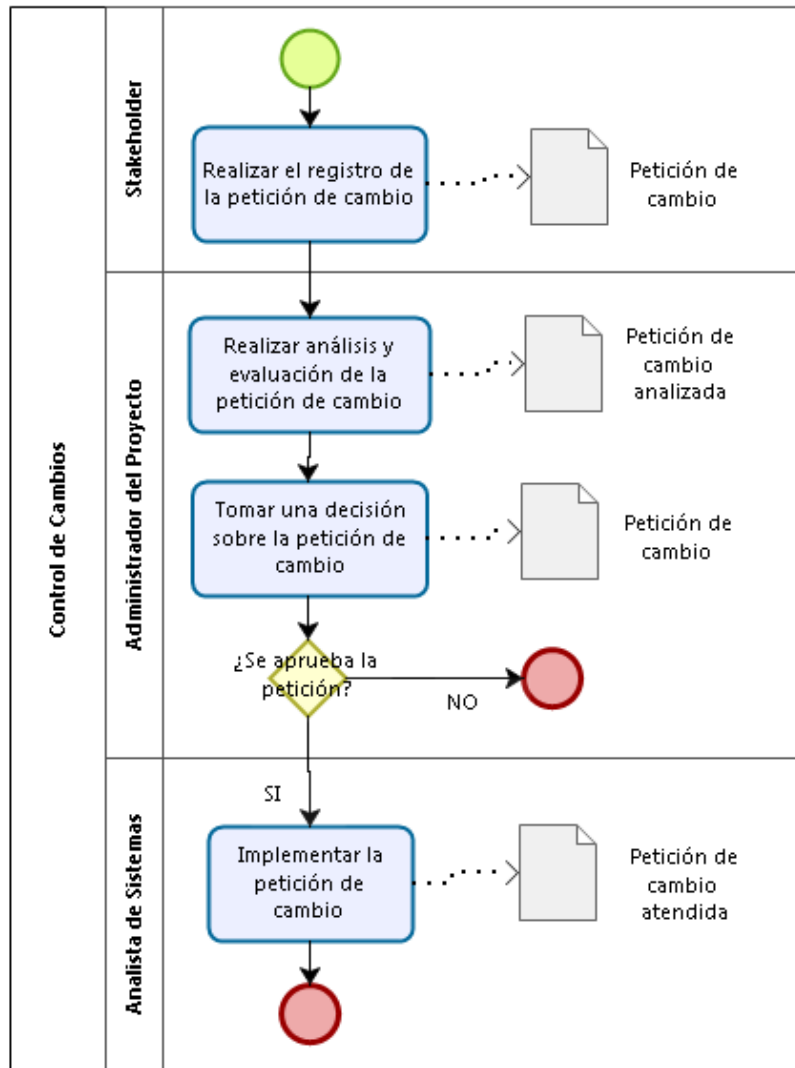


Figura 22: Diagrama - Control de Cambios en Requerimientos
Fuente: Elaborado por la autora

Tabla 39: Estados de requerimientos funcionales
Fuente: Elaborado por la autora

Actividades del proceso de desarrollo de software		
Nº	Actividad	Estado de requerimiento funcional
1	Realizar el despegue del proyecto	N/A - En esta actividad aún no se establecen requerimientos funcionales
2	Levantamiento y Análisis de requerimientos	
	2.1 Elicitar y Recolectar Requerimientos	Analizado
	2.2 Analizar requerimientos	
	2.3 Documentar requerimientos	Documentado
	2.4 Validar requerimientos	Validado
3	Diseño	Diseñado
4	Construcción	Implementado
5	Pruebas	Probado
6	Implantación	Implantado

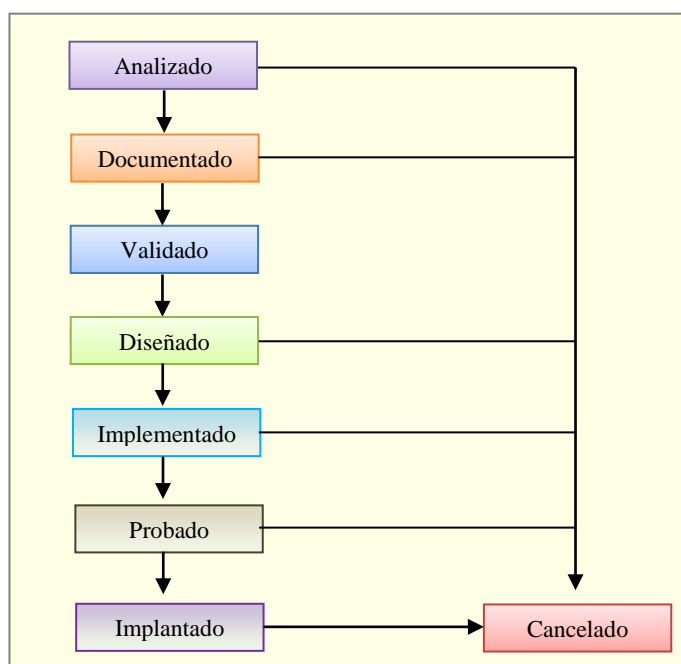


Figura 23: Diagrama de transición de estados de requerimientos funcionales

Fuente: Elaborado por la autora

- **Definir las tareas, responsables y artefactos para realizar seguimiento**

En la **Tabla 40** se han definido dos tareas que son parte del seguimiento de requerimientos de software. Las tareas descritas no son consecutivas y deben realizarse de acuerdo a la necesidad de cada proyecto.

Tabla 40: Tareas y responsables del seguimiento a requerimientos

Fuente: Elaborado por la autora

Actividad:	Seguimiento a requerimientos		Entrada:	Todos los artefactos de los procesos	
Nº	Tarea	Descripción	Responsable	Artefacto	
1	Monitorear estado y avance de cada requerimiento	Realizar revisiones periódicas del estado de cada requerimiento en el proceso de desarrollo de software con el fin de determinar avances en el proyecto. Para ello se debe revisar la información de trazabilidad y atributos de cada requerimiento.	Administrador del Proyecto	Reporte de Avance de proyecto	
2	Resolver problemas de requerimientos	Apoyar a los Analistas de Sistemas en la resolución de problemas de requerimientos, tales como información incompleta, conflictos, toma de decisiones, dudas e inquietudes, para lo cual se debe coordinar reuniones con todos los involucrados	Administrador del Proyecto	Acta de Reunión	

4.1.9. Definir artefactos y proceso para el mantenimiento de software

- **Determinar las tareas, responsables y artefactos del proceso de mantenimiento de software desarrollado.**

En la **Tabla 41** se pueden observar el detalle de entradas, actividades, responsables y artefactos del proceso de mantenimiento de software desarrollado para la Dirección de TIC.

Tabla 41: Tareas, responsables y artefactos del proceso de mantenimiento de software

Fuente: Elaborado por la autora

Proceso		Realizar mantenimiento de sistemas desarrollados	Entrada:	Orden de Mantenimiento [Estado: Registrada]	
Nº	Actividad	Descripción	Responsables	Artefactos	
1	Receptar la Orden de Mantenimiento y designar Analista	Receptar la Orden de Mantenimiento registrada por el usuario y designar al Analista de Sistemas responsable de su atención.	Responsable de Desarrollo de Software	Orden de Mantenimiento [Estado: En Revisión]	
2	Analizar Orden de Mantenimiento	Analizar la factibilidad técnica de la Orden de Mantenimiento y registrarla. Si se aprueba, continuar con la actividad 3. Caso contrario finalizar proceso.	Analista de Sistemas	Orden de Mantenimiento [Estado: Aprobada, No Aprobada]	
3	Dar atención a la Orden de Mantenimiento	Dependiendo de la Orden de Mantenimiento, se deberán realizar las siguientes tareas: - Replicar escenario reportado en ambiente de desarrollo (en caso de errores o mal funcionamiento del sistema). - Modificar código. - Realizar pruebas unitarias y de integración. - Realizar los cambios requeridos en ambiente de producción	Analista de Sistemas	[Estado: Aprobada]	
4	Registrar la atención de la Orden de Mantenimiento.	Registrar el trabajo realizado y entregar el mantenimiento al usuario solicitante.	Analista de Sistemas	Orden de Mantenimiento [Estado: Implementada/ Cerrada]	
Salida:		Orden de Mantenimiento [Estado: Implementada/ Cerrada, No Aprobada]			

- **Definir un diagrama de transición de estados para describir el ciclo de vida de las peticiones de mantenimiento.**

En la **Figura 24** se pueden observar los estados definidos para las órdenes de mantenimiento de software de la Unidad. El estado “Registrada” aparecerá cuando un usuario haya registrado su solicitud. Posteriormente, el Responsable de Desarrollo de Software revisará la solicitud y designará un Analista de Sistemas para su atención, momento en el cual la orden cambiará de estado a “En Revisión”.

El Analista analizará la factibilidad de implementación de la solicitud y determinará si es factible - “Aprobada” o no es factible – “No Aprobada”. En caso de ser factible, dará atención a la solicitud y registrará el trabajo realizado, momento en el cual la orden cambiará a estado “Implementada/ Cerrada”.

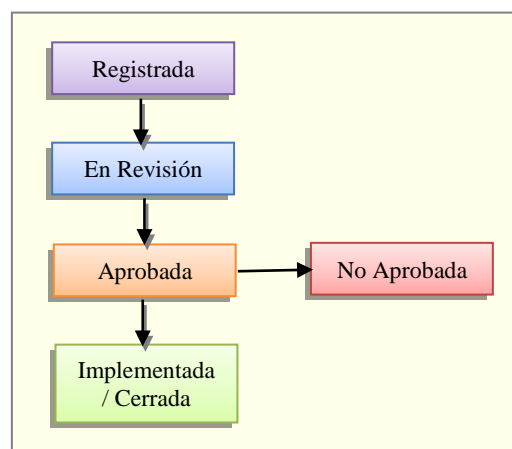


Figura 24: Estados de órdenes de mantenimiento

Fuente: Elaborado por la autora

- **Diagramar el proceso de mantenimiento de software**

En la **Figura 25** se puede observar el diagrama del proceso de mantenimiento de software definido para la Dirección de TIC.

4.1.10. Elaborar formatos para los artefactos definidos

- **Recopilar definiciones para todos los artefactos del proceso**

En la **Tabla 42** se listan todos los artefactos definidos en los procesos de Despegue, Elicitación, Análisis, Especificación, Validación y Verificación dentro del Desarrollo de Software, así como en el proceso de Mantenimiento de Software. Además se colocan definiciones para cada artefacto y referencias al formato elaborado.

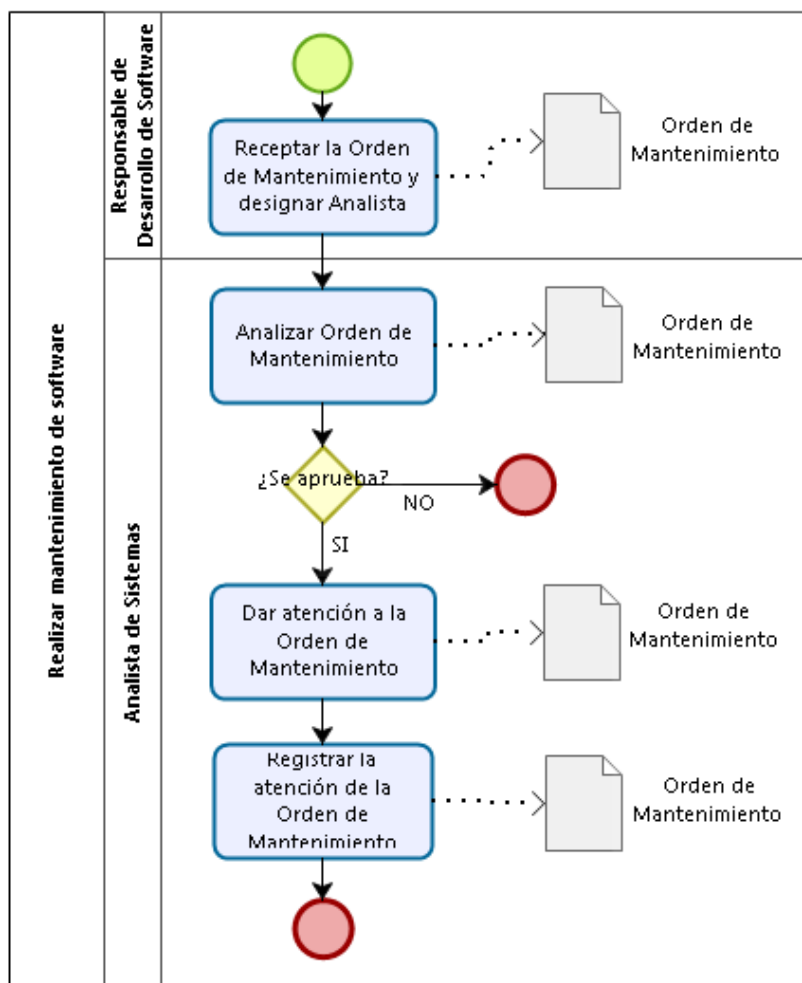


Figura 25: Diagrama - Mantenimiento de sistemas
Fuente: Elaborado por la autora

Tabla 42: Definiciones de artefactos del proceso
Fuente: Elaborado por la autora

Artefacto	Descripción	Formato	Referencia
Petición de desarrollo de software	La petición de desarrollo de software debe ser enviada a la Dirección de TIC a través del Sistema de Gestión Documental Quipux, mediante Memorando desde la Dirección requirente.	N/A	N/A
Convocatoria a reunión	La convocatoria se debe realizar formalmente a través del Sistema Documental Quipux, mediante un Memorando (interno) u Oficio (externo).	N/A	N/A
Acta de Reunión	Documento en el que se registra información de la reunión como: fecha, hora, lugar, objetivo, lista de convocados, temas tratados, compromisos adquiridos, listado de asistentes.	Formato de Acta de Reunión	Anexo 4

Continuación de la **Tabla 42.**

Artefacto	Descripción	Formato	Referencia
Registro de Stakeholders	Registro que permite almacenar la información de los stakeholders como: nombre, cargo, tipo, contactos.	Formato de Registro de Stakeholder	Tabla 43
Proyecto de Desarrollo de Software	Documento en el que se registra el alcance, objetivos, metas, responsables, recursos y cronograma del proyecto.	Formato de Proyecto de Software	Anexo 5
Registro de información relacionada	Registro que permite almacenar detalles de la documentación relacionada con el proyecto, tal como normativa legal, documentación de procesos, información histórica, entre otras.	Formato de Registro de información relacionada	Tabla 44
Registro de sistemas relacionados	Registro que permite almacenar información sobre los sistemas internos o externos que están relacionados con el sistema que se va a desarrollar.	Formato de Registro de sistemas relacionados	Tabla 45
Glosario de términos y abreviaturas	Registro que permite almacenar información sobre los términos y abreviaturas usadas en el proyecto.	Formato de Glosario de términos y abreviaturas	Tabla 46
Formato de observación para levantar información.	Formato que permite registrar la información de un proceso, misma que se ha levantado mediante la observación.	Formato de observación para levantar información.	Tabla 47
Formato de entrevista para levantar información.	Formato que permite registrar la información de un proceso, misma que se ha levantado con una entrevista.	Formato de entrevista para levantar información	Tabla 48
Registro de Requerimientos –Análisis	Registro que permite almacenar información del análisis de requerimientos, como la factibilidad, prioridad, clasificación, distribución y estimación.	Formato de Registro de Requerimientos	Tabla 52
Registro de Requerimientos - Especificación	Registro que permite almacenar información completa y detallada de cada requerimiento.	Registro de Requerimientos - Especificación	Tabla 53
Registro de Requerimientos - Validación	Registro que permite registrar información sobre la validación y verificación de los requerimientos.	Registro de Requerimientos - Validación	Tabla 54
Historial de cambios	Permite almacenar información relacionada a los cambios en artefactos y que implican nuevas versiones de los mismos.	Formato de Historial de cambios	Tabla 49

Continuación de la **Tabla 42.**

Artefacto	Descripción	Formato	Referencia
Matriz de trazabilidad	Permite registrar la información de trazabilidad de los artefactos del proyecto.	Formato de Matriz de trazabilidad	Tabla 50
Petición de cambios	Permite registrar una petición de cambio y su proceso de atención	Formato de Petición de cambios	Tabla 55
Orden de Mantenimiento	Permite registrar solicitudes de mantenimiento de software, así como su análisis y atención.	Formato Orden de Mantenimiento	Tabla 56
Caso de Prueba	Registra un conjunto de condiciones o variables bajo las cuáles se determinará si el sistema cumple un requerimiento específico.	Formato de Caso de Prueba	Tabla 51

- **Diseñar formatos para cada artefacto. Dar una definición a cada campo.**

A continuación se muestran los formatos diseñados para cada artefacto dentro del proceso de gestión de requerimientos de software, así como las definiciones de sus campos.

Tabla 43: Formato de Registro de Stakeholders

Fuente: Elaborado por la autora

Nombre	Cargo	Institución	Departamento	Tipo	Interno/ Externo	Contacto
<i>Nombre y Apellidos del stakeholder</i>	<i>Cargo que tiene el stakeholder en la institución</i>	<i>Nombre de la institución a la que pertenece el stakeholder</i>	<i>Dirección u Unidad administrativa a la que pertenece el stakeholder</i>	<i>Tipo de stakeholder:: Usuario, Contribuyente, Consultor, Patrocinador, Miembro del equipo de desarrollo</i>	<i>Interno o externo al GAD-I</i>	<i>Número de teléfono y extensión, Número celular, correo electrónico</i>

Tabla 44: Formato de registro de información relacionada

Fuente: Elaborado por la autora

Título	Tipo / Subtipo	Autor	Fecha de expedición	Descripción
<i>Título del documento</i>	<ul style="list-style-type: none"> - Normativa Legal: Ley, Ordenanza, Reglamento, Política, Resolución Administrativa. - Documentación de Procesos: Estructura Orgánica Funcional, Mapa de Proceso, Procedimiento, Instructivo, Estándar, Guía. - Información histórica: Estudios previos, Encuestas de satisfacción al ciudadano, Quejas del ciudadano. 	<i>Persona u Organización que ha producido el documento</i>	<i>Fecha de publicación del documento</i>	<i>Colocar el capítulo, sección, artículo y páginas que estén relacionadas con el proyecto.</i>

Tabla 45: Formato de Registro de sistemas relacionados

Fuente: Elaborado por la autora

Sistema	Descripción	Documentación de Usuario	Documentación Técnica
<i>Nombre del sistema relacionado</i>	<i>Descripción del sistema relacionado</i>	<i>Documento anexo de la documentación de usuario del sistema</i>	<i>Documento anexo de la documentación técnica del sistema</i>

Tabla 46: Formato de Glosario de términos y abreviaturas

Fuente: Elaborador por la autora

Término / Abreviatura	Significado
<i>Colocar el término o abreviatura</i>	<i>Colocar el significado o descripción del término o abreviatura</i>

Tabla 47: Formato de observación para levantar información

Fuente: Elaborado por la autora

Formato de observación para levantar información			
Proyecto:	<i>Nombre del proyecto</i>		
Fecha:	<i>Fecha de la entrevista</i>		
Proceso observado:	<i>Nombre del proceso objeto de observación</i>		
Analista de Sistemas	<i>Nombre del Analista de Sistemas responsable</i>		
Hora	Actividad	Responsable	Observación
<i>Hora inicio y fin de la actividad</i>	<i>Descripción detallada de la actividad</i>	<i>Cargo y Nombre del responsable de la ejecución de la actividad</i>	<i>Observaciones recopiladas durante la observación de la ejecución de la actividad</i>

Tabla 48: Formato de entrevista para levantar información

Fuente: Elaborado por la autora

Formato de entrevista para levantar información			
Proyecto:	<i>Nombre del proyecto</i>	Fecha:	<i>Fecha de la entrevista</i>
Entrevistado:	<i>Nombre del stakeholder entrevistado</i>	Cargo entrevistado:	<i>Cargo del stakeholder entrevistado</i>
Analista de Sistemas	<i>Nombre del Analista de Sistemas entrevistador</i>		
Entradas del proceso			
Entrada		Descripción	
Salidas del proceso			
Salida		Descripción	
Detalles del Proceso			
Nombre:	<i>Nombre del proceso</i>		
Objetivo:	<i>Objetivo del proceso</i>		
Evento de inicio:	<i>Describir el evento que da inicio al proceso</i>		
Actividad	Responsable	Formulario	
<i>Descripción detallada de la actividad</i>	<i>Cargo del responsable de la ejecución de la actividad</i>	<i>Nombre y referencia al formulario usado en este proceso</i>	

Tabla 49: Formato de Historial de cambios

Fuente: Elaborado por la autora

Historial de cambios			
Fecha	Responsable	Descripción del cambio	Justificación
<i>Fecha del cambio</i>	<i>Nombre del Responsable de implementar el cambio</i>	<i>Descripción de la solicitud de cambio</i>	<i>Referirse a la justificación de la petición de cambio</i>

Tabla 50: Formato de Matriz de Trazabilidad

Fuente: Elaborado por la autora

Matriz de trazabilidad				
Requerimiento Funcional	Elemento de Diseño	Elemento de Código	Caso de Prueba	Petición de cambio
<i>Descripción del requerimiento funcionales</i>	<i>Descripción de los elementos de diseño relacionados con el requerimiento</i>	<i>Descripción de los elementos de código relacionados con el requerimiento</i>	<i>Caso de prueba del requerimiento funcional</i>	<i>Referencia a las peticiones de cambio relacionadas con el requerimiento (En caso de existir)</i>

Tabla 51: Formato Caso de Prueba

Fuente: Elaborado por la autora

CASO DE PRUEBA								
PLANIFICACIÓN					EJECUCIÓN			Estado
Código Req.	Código Caso	Nombre Caso	Resultado esperado	Datos de entrada	Resultado obtenido	Observaciones	Adjuntos	
<i>Código del Requerimiento relacionado con el Caso de Prueba</i>	<i>Código del Caso: CP-00N</i>	<i>Nombre descriptivo del caso</i>	<i>Resultado esperado (funcionamiento normal del sistema)</i>	<i>Descripción de los datos de entrada a utilizar</i>	<i>Éxito / Falla</i>	<i>Observaciones a los resultados (mensajes, advertencias, entre otros)</i>	<i>Por ejemplo archivos con capturas de pantalla del resultado en el sistema.</i>	<i>Por Ejecutar, Ejecutado, Cancelado</i>

Tabla 52: Registro de Requerimientos – Análisis

Fuente: Elaborado por la autora

Registro de Requerimientos – Análisis										
Código	Nombre Corto	Descripción	Levantado por	Solicitante	Fecha de creación	Versión	Tipo	Prioridad	Complejidad	Estimación
<i>Código del requerimiento</i>	<i>Nombre corto que identifica al requerimiento</i>	<i>Descripción del requerimiento</i>	<i>Analista de Sistemas que elicito y documentó el requerimiento.</i>	<i>Nombre del Stakeholder que declaró el requerimiento</i>	<i>Fecha de levantamiento</i>	<i>Número de versión.</i>	<i>Funcional - No Funcional</i>	<i>ALTA, MEDIA, BAJA</i>	<i>ALTA, MEDIA, BAJA</i>	<i>En horas</i>

Tabla 53: Registro de Requerimientos – Especificación

Fuente: Elaborado por la autora

Registro de Requerimientos – Especificación								
Código	Base Legal	Precondiciones	Actores	Flujo de eventos principal	Post condición	Flujos alternos	Módulo	Responsable Implementación
<i>Código del requerimiento</i>	<i>Base legal que define el requerimiento.</i>	<i>Condiciones que se tienen que cumplir para que se pueda iniciar el caso de uso</i>	<i>Entidad externa al sistema que le demanda una funcionalidad</i>	<i>Describe el comportamiento ideal y principal del sistema</i>	<i>Estado en que queda el sistema una vez ejecutado el caso de uso</i>	<i>Describe excepciones o desviaciones del flujo básico y cómo el sistema se comporta en esos casos</i>	<i>Nombre del módulo en el cual se va a desarrollar el caso de uso</i>	<i>Nombre del Analista de Sistemas responsable de la implementación del requerimiento.</i>

Tabla 54: Registro de Requerimientos – Validación

Fuente: Elaborado por la autora

Registro de Requerimientos –Validación y Verificación				
Código	Nombre Corto	Descripción	Aprobado por	Fecha de aprobación
<i>Código del requerimiento</i>	<i>Nombre corto que identifica al requerimiento</i>	<i>Descripción del requerimiento</i>	<i>Nombre del stakeholder que aprobó el requerimiento</i>	<i>Fecha de aprobación del requerimientos</i>

Tabla 55: Formato de Petición de cambios

Fuente: Elaborado por la autora

PETICIÓN DE CAMBIO			
Fecha:	<i>Fecha de registro de la petición de cambio</i>	Número:	<i>Identificador único</i>
Estado:	<i>Registrado, En análisis, Cancelado, Implementado, Cerrado</i>		
Datos del proyecto			
Nombre proyecto:	<i>Nombre del proyecto del sistema afectado</i>	Módulo	<i>Módulo del sistema afectado</i>
Datos del solicitante			
Nombre:	<i>Nombre del solicitante</i>	Cargo:	<i>Cargo del solicitante</i>
Dirección:	<i>Área funcional que solicita el cambio</i>	Teléfono:	<i>Teléfono de contacto del solicitante</i>
Correo electrónico:	<i>Correo del solicitante</i>		
REGISTRO DE LA PETICIÓN DE CAMBIO			
Título:	<i>Resumen de una línea de la petición de cambio</i>		
Tipo de cambio:	<i>Nuevo requerimiento, Cambio de requerimiento, Eliminación de requerimiento</i>		
Descripción de la petición:	<i>Descripción del requerimiento a añadir, borrar, o cambiar.</i>		
Justificación:	<i>Colocar una descripción de la justificación legal o técnica del cambio</i>		
Prioridad:	<i>Alta, Media, Baja</i>		
ANÁLISIS Y EVALUACIÓN DE LA PETICIÓN DE CAMBIO			
Responsable	<i>Nombre del funcionario responsable de realizar el análisis de la petición de cambio</i>		
Descripción del análisis	<i>Descripción de análisis realizado:</i> - Factibilidad técnica y económica. - Cantidad de esfuerzo y tiempo requeridos. - Impacto en el proyecto: cambios en el cronograma, cambios en el presupuesto, cambios al alcance de la funcionalidad, modificación de artefactos (especificación de requerimientos, documento de diseño, pruebas, código), calidad del software, tareas requeridas, recursos requeridos.		
Decisión adoptada	<i>Aprobado, Rechazado</i>	Fecha de decisión	<i>Fecha</i>
Responsable de decisión:	<i>Nombre del responsable que tomó la decisión</i>	Razón de la decisión:	<i>Razón legal, técnica o económica de la decisión</i>
IMPLEMENTACIÓN DE LA PETICIÓN DE CAMBIO			
Responsable:	<i>Nombre del responsable de la implementación</i>		
Fecha inicio:	<i>Inicio de implementación</i>	Fecha fin:	<i>Fin de implementación</i>
Artefactos:	<i>Descripción de módulos, componentes, artefactos modificados.</i>		
CIERRE			
Cerrado por:	<i>Nombre funcionario que cierra la petición</i>	Fecha	<i>Fecha de cierre</i>

Tabla 56: Formato Orden de Mantenimiento

Fuente: Unidad de Desarrollo de Software del GAD-I (2017)

ORDEN DE MANTENIMIENTO/ PEDIDOS VARIOS DE SOFTWARE					
Nro. Orden:		Fecha:			
Solicitante:	<i>Nombre del Solicitante</i>				
Cargo:	<i>Cargo del Solicitante</i>				
Tipo Solicitante:	Interno		Interno (TIC)		Externo
Dirección/ Entidad	<i>Dirección Administrativa</i>	Unidad:	<i>Unidad Administrativa</i>		
Descripción del requerimiento					
<i>Descripción detallada del requerimiento, documentada por el usuario.</i>					
			Autorizado por:		
Firma/sello Director Dpto.	Firma/sello del Solicitante	DIRECTOR TIC			
Análisis del Requerimiento (En caso de que aplique)					
Factible		¿Por qué?			
No Factible					
Control de Cambios (En caso de que aplique)					
Nombre de la Aplicación:					
Descripción del trabajo realizado					
<i>Descripción del trabajo realizado por el Analista de Sistemas</i>					
Fecha de inicio:		Fecha de terminación:			
Constancia de Entrega del Requerimiento					
ANALISTA DE SISTEMAS	SOLICITANTE		RESPONSABLE DE DESARROLLO DE SOFTWARE		

4.1.11. Definir políticas del proceso de gestión de requerimientos de software

En coordinación con los funcionarios de la Unidad de Desarrollo de Software se ha definido las siguientes políticas del proceso de gestión de requerimientos de software:

- Para iniciar un proyecto de desarrollo de software, siempre se debe tener una solicitud escrita por parte del Director del área requirente.
- Los stakeholders de nivel directivo son los responsables de revisar y aprobar el Proyecto de Software, así como de garantizar que se disponga de todos los recursos necesarios.
- El Analista de Sistemas no debe añadir características o capacidades que no hayan sido solicitadas por el usuario, ya que esto puede afectar el costo y el tiempo de implementación.
- El Analista de Sistemas no debe tomar decisiones sobre requerimientos, ya que son responsabilidad del usuario dentro del equipo de trabajo.
- Es obligatoria la participación de todos los stakeholders en los procesos de despegue, elicitación, análisis y validación de requerimientos.
- La aprobación de los requerimientos la deben realizar los stakeholders de cada proyecto.

- El Analista de Sistemas no debe aceptar cambios no autorizados en requerimientos. Todas las peticiones de cambios, pequeñas o grandes, deben ser canalizadas a través del mecanismo de control de cambios para el caso de desarrollo de software.
- Es obligatorio realizar control de versiones en los artefactos apenas se realice un cambio, con el fin de garantizar que los miembros del proyecto tengan siempre acceso a las versiones vigentes de los mismos.
- Todas las peticiones de mantenimiento de software, deben ser canalizadas a través de la Orden de Mantenimiento.

4.1.12. Documentación de la guía metodológica

El **Anexo 8** contiene la guía metodológica que se desarrolló para el presente estudio, con la descripción documentada de cada una de las actividades del proceso de gestión de requerimientos de software: Elicitar y Recolectar, Analizar, Especificar, Validar y Verificar, Control de Versiones, Control de Cambios, Trazabilidad y Seguimiento.

4.2. Desarrollo del Objetivo Específico 2

El segundo objetivo específico es el de determinar las características funcionales de una aplicación de gestión de requerimientos de software para la Dirección de TIC, mediante el análisis de herramientas existentes en el mercado.

4.2.1. Seleccionar un conjunto de herramientas de gestión de requerimientos de software existentes en el mercado para el estudio

- **Buscar en libros, revistas y sitios web, herramientas de gestión de requerimientos existen en la actualidad:** La **Tabla 57** ha sido elaborada en base a referencias de herramientas encontradas en el sitio web de Volere (Atlantic Systems Guild, 2016) y en el sitio web de Administración de Requerimientos (Ludwig Consulting Services, 2009).

Tabla 57: Listado de herramientas de gestión de requerimientos encontradas

Fuente: Elaborada con información tomada de las fuentes bibliográficas

Herramienta	Compañía	Sitio Web
Caliber 11.5	Micro Focus – Reino Unido y Estados Unidos	https://www.microfocus.com/products/requirements-management/caliber
Axure RP 8	Axure Software Solutions – California, Estados Unidos	http://www.axure.com
Accompa	Accompa, Inc – California, Estados Unidos	http://web.accompa.com
Avenqo PEP 1.9 (Project Engineering Platform)	Avenqo GmbH - Alemania	http://www.avenqo.com
Agosense Requirements	Agosense - Alemania	http://www.agosense.com/agosense/requirements
Rational DOORS 9.6.1	IBM Corporation – Nueva York, Estados Unidos	http://www-03.ibm.com/software/products/en/ratidoor
CaseComplete	Serlio Software – Wisconsin, Estados Unidos	http://www.casecomplete.com
codeBeamerRequirements Management 8	Intland Software - Alemania	https://intland.com/requirements-management
CognitionCockpit	Cognition Corporation – Massachusetts, Estados Unidos	http://www.cognition.us
CradleVersion 7.1	Structured Software SystemsLtd -	https://www.threesl.com/
Gatherspace	Gatherspace – California, Estados Unidos	http://www.gatherspace.com
Polarion REQUIREMENTS	Polarion Software – California, Estados Unidos	https://polarion.plm.automation.siemens.com/products/polarion-requirements
ProR	Eclipse Foundation – Ottawa, Canadá	http://www.eclipse.org/rmf/pror
REQCHECKER 1.5.2	SGSOFT France - Francia	http://reqchecker.eu
RequirementONE	RequirementOneInc – California, Estados Unidos	http://www.requirementone.com
ReqSuite	OSSENO Software GmbH - Alemania	http://www.osseno.de

Continuación de la **Tabla 57**.

Herramienta	Compañía	Sitio Web
RMTrak	Prometeo Technologies – Nueva York, Estados Unidos	https://www.rmtrak.com
VeroTrace	Verocel – Massachusetts, Estados Unidos	http://www.verocel.com/products/requirements-traceability/
VisureRequirements	VisureSolutions – San Francisco, Estados Unidos	http://www.visuresolutions.com/requirements-engineering-tool

- **Realizar una revisión de cada herramienta encontrada, con el objetivo de determinar las siguientes características:**
 - **Vigencia de la herramienta:** determinar si se encuentra vigente en la actualidad, si se puede acceder a su sitio oficial y si existe una empresa o comunidad que le de soporte.
 - **Acceso a versiones de prueba:** dependiendo del tipo de licencia, determinar si se puede descargar el paquete de la herramienta para instalarlo, o si se puede acceder a una versión en la nube.
 - **Acceso a documentación:** determinar si en el sitio oficial de la herramienta se encuentra disponible la documentación técnica y de usuario, sea en manuales o video tutoriales.
 - **Coherencia de la herramienta:** determinar si las funcionalidades de la herramienta se encuentran dentro del alcance del presente estudio, debido a que pueden existir herramientas especializadas en una actividad específica como por ejemplo el modelado de requerimientos.
 - **Idioma de la herramienta:** de preferencia se elegirá una herramienta en idioma inglés o español.

En la **Tabla 58** se puede observar el resultado del análisis de cada herramienta de gestión de requerimientos, así como las observaciones que justifican el por qué una herramienta no cumple o cumple parcialmente con las características de: vigencia, acceso a versiones de prueba, acceso a documentación, coherencia e idioma.

- **Seleccionar un conjunto de herramientas de gestión de requerimientos de software para su análisis, en base a los resultados obtenidos en la tarea anterior:** En la **Tabla 59** se pueden observar las seis herramientas que se han elegido y que cumplen al menos cuatro de las 5 características analizadas.

Tabla 58: Herramientas de gestión de requerimientos con sus características generales
Fuente: Elaborado por la autora

Herramienta	Vigencia	Versiones de prueba	Documentación	Coherencia	Idioma	Observaciones
Accompa	✓	✓	✓	✓	✓	
Avenqo PEP 1.9 (Project Engineering Platform)	✓	X	X	X	X	Al momento de la revisión de esta herramienta, no se encontró documentación técnica o de usuario disponible. Para usar la herramienta era necesario enviar un correo a la empresa, pero dicha empresa no se contactó para entregar el serial de prueba. Al no poder usar la herramienta ni tener acceso a su documentación, no se puede determinar la coherencia de la misma con el presente estudio.
Caliber 11.5	✓	✓	✓	✓	✓	
Axure RP 8	✓	✓	✓	X	✓	La herramienta permite modelar requerimientos de sitios web en wireframes, prototipos y diagramas de flujo. Su funcionalidad no se encuentra dentro del alcance del presente estudio.
Agosense Requirements	✓	X	X	X	X	El sitio oficial no permite descarga de versión de prueba ni tiene demo online. Sólo existen videos demostrativos de su integración con otras herramientas. No existe suficiente documentación y la que existe está en idioma alemán.
Rational DOORS 9.6.1	✓	✓	✓	✓	✓	
CaseComplete	✓	✓	✓	✓	✓	
codeBeamerRequirements Management 8	✓	✓	✓	✓	✓	
CognitionCockpit	✓	X	X	X	✓	Al momento de la revisión de esta herramienta no se tuvo acceso a una versión de prueba. No existía suficiente documentación para descargar. La herramienta está orientada mayormente a proyectos médicos o farmacéuticos.
CradleVersion 7.1	✓	✓	✓	✓	✓	
Gatherspace	✓	✓	✓	✓	✓	
Polarion REQUIREMENTS	✓	✓	✓	✓	✓	
ProR	✓	✓	✓	✓	✓	
REQCHECKER 1.5.2	✓	✓	✓	X	✓	La funcionalidad de la herramienta no está acorde al alcance del presente estudio.

Tabla 59: Listado de herramientas seleccionadas para el estudio

Fuente: Elaborado por la autora

Herramienta	Compañía	Idioma	Licencia	Sitio Web
Caliber 11.5	Micro Focus – Reino Unido y Estados Unidos	Inglés	Comercial	https://www.microfocus.com/products/requirements-management/caliber
Accompa	Accompa, Inc – California, Estados Unidos	Inglés	Comercial	http://web.accompa.com
CaseComplete	Serlio Software – Wisconsin, Estados Unidos	Inglés	Comercial	http://www.casecomplete.com
CodeBeamer Requirements Management 8	Intland Software - Alemania	Inglés	Comercial	https://intland.com/requirements-management
Gatherspace	Gatherspace – California, Estados Unidos	Inglés	Comercial	http://www.gatherspace.com
ProR	Eclipse Foundation – Ottawa, Canadá	Inglés	Open Source	http://www.eclipse.org/rmf/pror

4.2.2. Determinar criterios de agrupamiento de las características funcionales de una herramienta de gestión de requerimientos de software

- **Realizar una revisión del estado del arte de Ingeniería de Requerimientos, con el fin de comprender lo que debe abarcar una herramienta de gestión de requerimientos:** La revisión del estado del arte consta en el Capítulo II - Marco Referencial.
- **Determinar los criterios de agrupamiento para el estudio:** Cualquier herramienta de gestión de requerimientos de software debe ser capaz de soportar las tareas que se deriven de las actividades de la Ingeniería de Requerimientos, por ello se han determinado los siguientes criterios de agrupamiento:

Desarrollo de Requerimientos

- Elicitación
- Análisis
- Especificación
- Validación

Administración de Requerimientos

- Control de cambios
- Control de versiones
- Trazabilidad
- Seguimiento

4.2.3. Verificar las funcionalidades de cada herramienta mediante la instalación, uso, y revisión de documentación técnica, dependiendo del caso.

- **Instalar o solicitar acceso a versiones de prueba de cada herramienta, y obtener documentación técnica y de usuario a través de su sitio oficial**

Las herramientas listadas en la **Tabla 60** tienen una versión de evaluación por 30 días para descargar o para usar como servicio, mediante las cuales se pudo verificar y constatar las funcionalidades. Además en los sitios web de cada herramienta se encontró documentación como manuales de usuario y videos de uso en idioma inglés.

Tabla 60: Paquetes e información obtenida de las herramientas seleccionadas

Fuente: Elaborado por la autora

Herramienta	Observación
Caliber 11.5	- Paquete con versión de evaluación por 30 días. - Acceso a documentación técnica y de usuario.
Accompa	- Acceso a documentación técnica y de usuario
Rational DOORS 9.6.1	- Paquete con versión de evaluación por 30 días. - Acceso a documentación técnica y de usuario.
CaseComplete	- Paquete con versión de evaluación por 30 días. - Acceso a documentación técnica y de usuario.
CodeBeamerRequirements Management 8	- Paquete con versión de evaluación por 30 días. - Acceso a documentación técnica y de usuario.
Gatherspace	- Versión de prueba en la nube. - Acceso a documentación técnica y de usuario.
ProR	- Paquete Open Source. Acceso a documentación.

- **Determinar las funcionalidades de cada herramienta, mediante el uso, revisión de la documentación técnica, observación de video tutoriales, y/o demostraciones por parte de la empresa**

Desde la **Tabla 61** hasta la **Tabla 65**, se presentan las características funcionales de las herramientas de gestión de requerimientos Case Complete, Accompa, Caliber, CodeBeamerRequirements Management y Gatherspace. En el caso de ProR no se elaboró una tabla con las características debido a que tiene funcionalidades muy básicas, sin embargo se puede destacar que: forma parte del proyecto Eclipse, es de código abierto y soporta el estándar ReqIF 1.0.1.

Tabla 61: Características funcionales de la herramienta Case Complete 2016

Fuente: Elaborado por la autora

Herramienta	Case Complete 2016
Compañía	Serlio Software – Wisconsin, Estados Unidos
Sitio Web	http://www.casecomplete.com
Licencia	Versión de evaluación por 30 días - Comercial
Tipo	De Escritorio
Curva de aprendizaje	Bajo
Desarrollo de Requerimientos	
Actividad	Característica
Elicitación	<ul style="list-style-type: none"> - Cada proyecto tiene una hoja de ruta que muestra las actividades y tareas, con descripciones detalladas, que se deben realizar para completar un proceso de gestión de requerimientos. - Permite registrar la visión y alcance del proyecto en una plantilla de documento de texto. Esta plantilla contiene: Introducción, Propósito, Stakeholders, Descripción del producto, Requerimientos externos y Restricciones.
Análisis	<ul style="list-style-type: none"> - Permite crear y mantener un diccionario de términos y definiciones. - Permite crear y mantener un diccionario para los campos de un objeto del dominio de la aplicación, así como el registro de las restricciones de dichos campos.
Especificación	<ul style="list-style-type: none"> - Permite registrar a cada actor, con su descripción y metas dentro del proyecto. - Permite la creación de Casos de Uso a partir de las metas de un stakeholder, describiendo el propósito, precondiciones, flujo de eventos principal, y resultados esperados. - Permite la creación de Historias de Usuario, que contienen una descripción y criterios de aceptación. - Permite registrar requerimientos no funcionales y relacionarlos con otros ítems dentro del proyecto. - Permite registrar información adicional para los casos de uso, actores, paquetes, como: notas, problemas y documentos relacionados. - Permite la creación de diagramas UML: Diagrama de Contexto, Diagrama de Casos de Uso, Diagrama de Actividades. - Permite crear prototipos de pantallas del sistema. - Permite compartir el contenido a través de la generación de reportes y exportación del proyecto.
Validación	<ul style="list-style-type: none"> - Permite crear Casos de Prueba a partir de un Caso de Uso.

Continuación de la **Tabla 61**.

Administración de Requerimientos	
Actividad	Característica
Control de cambios	Ninguna encontrada
Control de versiones	Ninguna encontrada
Trazabilidad	Ninguna encontrada
Seguimiento	Ninguna encontrada

Tabla 62: Características funcionales de la herramienta Accompa

Fuente: Elaborado por la autora

Herramienta	Accompa
Compañía	Accompa, Inc – California, Estados Unidos
Sitio Web	http://web.accompa.com
Licencia	Versión de evaluación por 30 días - Comercial
Tipo	Web
Curva de aprendizaje	Medio
Desarrollo de Requerimientos	
Actividad	Características
Elicitación	<ul style="list-style-type: none"> - Tiene tableros de discusión que permiten realizar preguntas relacionadas con un requerimiento a los stakeholders. - Permite crear formularios web personalizados que sirven para la recolección de requerimientos. - Los stakeholders pueden hacer peticiones de características desde su computador o dispositivo móvil.
Análisis	Ninguna encontrada
Especificación	<ul style="list-style-type: none"> - Permite agregar descripciones de requerimientos a través del uso de un editor de texto rico. - Permite importar requerimientos desde archivos Excel. - Permite agregar documentos adjuntos para cada requerimiento. - Permite crear atributos personalizados. Estos atributos se pueden usar para rastrear los requerimientos. - Permite crear documentos de requerimientos en formato PDF y HTML. - Permite manejar roles de usuario personalizados para definir quién puede actualizar los requerimientos y prevenir cambios no autorizados.
Validación	<ul style="list-style-type: none"> - Permite compartir requerimientos en tiempo real para lograr colaboración de todos los stakeholders. Todas las conversaciones son capturadas y archivadas para referencias futuras. - Permite obtener aprobación electrónica de los stakeholders.

Continuación de la **Tabla 62**

Administración de Requerimientos	
Actividad	Característica
Control de cambios	- Permite bloquear/desbloquear un conjunto de requerimientos para evitar cambios a requerimientos no aprobados o inesperados.
Control de versiones	<ul style="list-style-type: none"> - Permite rastrear el historial completo de cambios de cada requerimiento. En cualquier momento se puede consultar y revertir un requerimiento a cualquiera de sus versiones previas. - Es posible visualizar versiones previas de cada requerimiento. Esto se puede usar para documentar, auditar y gestionar el cambio. - Cada vez que se hace un cambio a un requerimiento, automáticamente se incrementa el número de versión de ese requerimiento. Se pueden comparar dos versiones y ver cuáles son los cambios, así como quién y cuándo los hizo.
Trazabilidad	<ul style="list-style-type: none"> - Permite crear y rastrear las relaciones y dependencias entre requerimientos. - Permite crear una matriz de trazabilidad para analizar relaciones entre los artefactos de requerimientos. Por ejemplo, asegurar que todas las características tengan al menos un requerimiento funcional.
Seguimiento	<ul style="list-style-type: none"> - Permite crear y rastrear requerimientos, casos de uso y artefactos. - Permite manejar y rastrear diferentes tipos de requerimientos (de negocio, funcionales, de interfaz, no funcionales: de rendimiento, de seguridad).

Tabla 63: Características funcionales de la herramienta Caliber 11.5

Fuente: Elaborado por la autora

Herramienta	Caliber 11.5
Compañía	Micro Focus – Reino Unido y Estados Unidos
Sitio Web	https://www.microfocus.com/products/requirements-management/caliber
Licencia	Versión de evaluación por 30 días - Comercial
Tipo	Escritorio - Web
Curva de aprendizaje	Alto
Desarrollo de Requerimientos	
Actividad	Característica
Elicitación	Ninguna encontrada
Análisis	Ninguna encontrada

Continuación de la **Tabla 63**

Actividad	Característica
Especificación	<ul style="list-style-type: none"> - Tiene una base de datos centralizada para los requerimientos. - Permite crear historias de usuario y asociarlas a requerimientos de negocio. - Permite crear atributos de requerimientos personalizados. - Permite ingresar datos como texto rico, tablas, imágenes. - Permite la creación de escenarios para representar casos de uso, que incluyen pasos de actividades, decisiones, actores. - Permite crear atributos definidos por el usuario que pueden ser de los siguientes tipos; booleano, fecha, tiempo, entero largo, lista de selección múltiple, entre otros. - Permite capturar procesos y flujos de trabajo para los requerimientos.
Validación	<ul style="list-style-type: none"> - Permite a los stakeholders realizar revisiones de los proyectos e iniciar discusiones sobre los mismos. - Permite realizar aprobaciones mediante firma electrónica y bloquear las líneas base. - Permite crear simulaciones para mostrar como una interface de usuario va a funcionar para diferentes usuarios. Los stakeholders pueden probar las interfaces y asegurarse de que todo esté correcto antes de iniciar el desarrollo. - Permite generar una serie de casos de prueba para cada posible camino de un escenario predefinido. - Tiene un procedimiento de validación que contiene los pasos los criterios de aceptación para asegurar que el requerimiento cumple con las necesidades del negocio.
Administración de Requerimientos	
Actividad	Característica
Control de cambios	<ul style="list-style-type: none"> - Presenta el historial de todos los cambios hechos a un requerimiento, incluyendo la versión, fecha/hora, usuario que hizo el cambio, comentarios, y los datos que fueron cambiados.
Control de versiones	<ul style="list-style-type: none"> - Permite hacer comparaciones entre dos líneas base, para ver lo que se ha añadido, borrado, cambiado o movido. - Permite comparar dos versiones de requerimientos y ver las diferencias, resaltando las diferencias y comparando cada atributo.
Trazabilidad	<ul style="list-style-type: none"> - Permite realizar trazabilidad de requerimientos a través de la Matriz de Trazabilidad. - Provee un diagrama de trazabilidad que permite visualizar los requerimientos y los objetos hasta y hacia el requerimiento.
Seguimiento	<ul style="list-style-type: none"> - Permite realizar seguimiento del estado de los requerimientos.

Tabla 64: Características funcionales herramienta CodeBeamerRequirements Management 8
Fuente: Elaborado por la autora

Herramienta	CodeBeamerRequirements Management 8
Compañía	Intland Software - Alemania
Sitio Web	https://intland.com/requirements-management
Licencia	Versión de evaluación por 30 días - Comercial
Tipo	Web
Curva de aprendizaje	Medio
Desarrollo de Requerimientos	
Actividad	Característica
Elicitación	Ninguna encontrada
Análisis	Ninguna encontrada
Especificación	<ul style="list-style-type: none"> - Permite realizar la administración de los requerimientos a través del uso de las siguientes vistas: <ul style="list-style-type: none"> o Vista de Tabla: presenta una lista de los requerimientos registrados con un resumen de sus atributos. o Vista de Documento: presenta una vista en un editor en el cual se puede editar los requerimientos, agregar adjuntos como imágenes o diagramas, comentarios. o Vista de Ítem: permite agregar o editar un requerimiento. Los campos que tiene son descripción, estado, comentarios, adjuntos, y otros campos personalizados. - Permite priorizar requerimientos. - Permite usar plantillas personalizadas de documentos para la exportación de requerimientos.
Validación	<ul style="list-style-type: none"> - Permite crear casos de prueba a partir de los requerimientos. - Permite definir y ejecutar un conjunto de casos de prueba de requerimientos.
Administración de Requerimientos	
Actividad	Característica
Control de cambios	Ninguna encontrada
Control de versiones	<ul style="list-style-type: none"> - Tiene una funcionalidad para planear y administrar las versiones del proyecto, permitiéndole monitorear el progreso del mismo.
Trazabilidad	<ul style="list-style-type: none"> - Permite realizar trazabilidad desde los requerimientos hacia el código fuente, tareas, casos de prueba, versiones y otros artefactos.
Seguimiento	<ul style="list-style-type: none"> - Permite crear flujos de trabajo personalizados para la gestión de requerimientos. - Permite capturar el estado del proyecto guardando una línea base de todos los requerimientos, relaciones, comentarios, documentos, sus relaciones e historial de un punto específico en el tiempo.

Tabla 65: Características funcionales de la herramienta Gatherspace 2.2

Fuente: Elaborado por la autora

Herramienta	Gatherspace 2.2
Compañía	Gatherspace – California, Estados Unidos
Sitio Web	http://www.gatherspace.com
Licencia	Acceso gratuito a un entorno de pruebas - Comercial
Tipo	Web (SaaS)
Curva de aprendizaje	Bajo
Desarrollo de Requerimientos	
Actividad	Característica
Elicitación	- Permite la creación y administración de un diccionario con términos y definiciones para el proyecto.
Análisis	Ninguna encontrada
Especificación	<ul style="list-style-type: none"> - Permite crear y administrar requerimientos de negocio y requerimientos del sistema. Cada uno de ellos tiene un código, nombre, a quien está asignado, prioridad, estado. - Permite crear y administrar actores. - Permite especificar Casos de Uso con nombre, descripción, actor, precondition, post condición, estado, prioridad, casos de uso asociados, flujo de eventos principal y alternativo. - Permite exportar reportes con datos de actores, requerimientos de negocio y de sistema, casos de uso, diccionario de términos.
Validación	- Permite la especificación de casos de prueba con código, nombre, prioridad, asignado a, estado.
Administración de Requerimientos	
Actividad	Característica
Control de cambios	- Registra la fecha y hora de los cambios realizados a requerimientos y casos de uso.
Control de versiones	Ninguna encontrada
Trazabilidad	- Permite relacionar requerimientos con otros requerimientos y con casos de uso.
Seguimiento	<ul style="list-style-type: none"> - Permite crear y administrar problemas, con los siguientes datos: nombre, descripción, prioridad, complejidad, estado, asignado a, requerimiento asociado. - Permite crear y administrar iteraciones para el proyecto (Código, nombre, descripción, estado, fecha de inicio, fecha de finalización). - Permite crear y administrar proyectos y usuarios del proyecto.

4.2.4. Determinar las características funcionales de un sistema de gestión de requerimientos de software, que se considerarán para el presente estudio.

- **Realizar una presentación de las características funcionales de las herramientas analizadas a los funcionarios de la Unidad de Desarrollo de Software**

Luego de haber analizado las herramientas de gestión de requerimientos de software existentes en el mercado, se realizó una reunión de trabajo con la Unidad de Desarrollo de Software de la Dirección de TIC, en donde se determinaron las características funcionales que se considerarán para el presente estudio y que se detallan en la **Tabla 66**.

- **Seleccionar las características funcionales a implementar en la aplicación informática, en base a las necesidades de la Unidad de Desarrollo de Software y al alcance del presente estudio**

Los criterios que se consideraron para la determinación de las características funcionales que tendrá la aplicación informática de gestión de requerimientos de la Dirección de TIC fueron: alcance del estudio, conocimientos y habilidades de los funcionarios, tiempo requerido para la utilización de la aplicación informática, normativa para el control de desarrollo y mantenimiento de software, necesidades de los funcionarios de la Unidad de Desarrollo de Software.

Tabla 66: Características funcionales seleccionadas para la aplicación informática
Fuente: Elaborado por la autora

Desarrollo de Requerimientos	
Actividad	Característica
Elicitación	<ul style="list-style-type: none"> - Registrar convocatorias a reuniones. - Registrar la información obtenida durante la elicitación de requerimientos. - Registrar actas de reuniones. - Registrar los sistemas relacionados. - Registrar la documentación relacionada. - Crear y mantener un diccionario de términos y definiciones.
Análisis	<ul style="list-style-type: none"> - Registrar los requerimientos levantados durante la elicitación. - Clasificar y priorizar los requerimientos. - Estimar tiempo de implementación de los requerimientos.
Especificación	<ul style="list-style-type: none"> - Base de datos centralizada para los requerimientos de proyectos. - Registrar atributos de requerimientos. - Crear y exportar documentos de requerimientos. - Registrar requerimientos levantados de acuerdo a plantillas establecidas. - Agregar documentos adjuntos para cada requerimiento. - Realizar búsquedas de requerimientos con palabras clave y aplicación de filtros.

Continuación de la **Tabla 66**

Desarrollo de Requerimientos	
Actividad	Característica
Validación	<ul style="list-style-type: none"> - Registrar la aprobación de requerimientos de los stakeholders. - Registrar especificaciones de requerimientos validadas.
Administración de Requerimientos	
Actividad	Característica
Control de cambios	<ul style="list-style-type: none"> - Presenta el historial de todos los cambios hechos a un requerimiento, incluyendo la versión, fecha/hora, usuario que hizo el cambio, comentarios, y los datos que fueron cambiados. - Permite gestionar las peticiones de cambio. - Reportes del total de peticiones de cambio recibidas y sus estados.
Control del versiones	<ul style="list-style-type: none"> - Cada vez que se hace un cambio a un requerimiento, automáticamente se incrementa el número de versión de ese requerimiento.
Trazabilidad	<ul style="list-style-type: none"> - Crear y administrar relaciones de trazabilidad. - Generar matriz de trazabilidad para analizar relaciones entre los artefactos del proyecto.
Seguimiento	<ul style="list-style-type: none"> - Manejar estados para los requerimientos dentro de su ciclo de vida. - Crear reportes del estado de los requerimientos dentro del proyecto.
Configuración del sistema	<ul style="list-style-type: none"> - Crear y administrar proyectos. - Crear y administrar catálogos de datos: tipos de stakeholders, tipos de documentación técnica, tipos de sistemas, artefactos. - Hoja de ruta con las actividades y tareas detalladas, que se deben realizar para completar un proceso de gestión de requerimientos.

4.3. Desarrollo del Objetivo Específico 3

El tercer objetivo específico es el de elaborar una aplicación informática para la gestión de requerimientos de software de la Dirección de TIC, mediante la sistematización del proceso definido anteriormente y la implementación de características funcionales seleccionadas de las herramientas analizadas.

En la **Figura 26** se puede visualizar el diagrama de arquitectura de la aplicación informática, que responde a los requerimientos de la Dirección de TIC en cuanto a modelo de capas y herramientas tecnológicas.

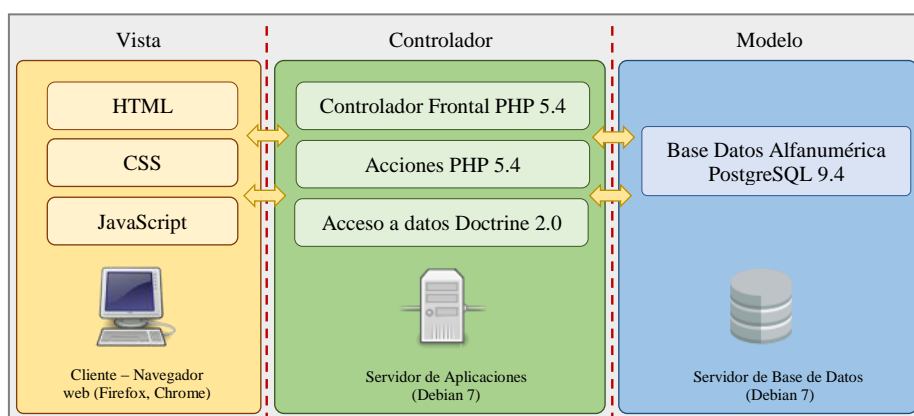


Figura 26: Diagrama de arquitectura de la aplicación informática

Fuente: Elaborado por la autora

Descripción de las herramientas de desarrollo:

- **HTML:** lenguaje de marcado para crear páginas que pueden ser desplegadas en un navegador web.
- **JavaScript:** lenguaje de programación interpretado y orientado a objetos que se utiliza principalmente del lado del cliente y permite mejoras en la interfaz de usuario.
- **PHP 5.4:** lenguaje de código abierto, adecuado para el desarrollo web y que puede ser incrustado en HTML.
- **Doctrine 2.0:** mapeador de objetos-relacional que proporciona una capa de persistencia para objetos PHP sobre una base de datos.
- **PostgreSQL 9.4:** sistema de administración de base de datos objeto relacional.

De la misma manera, en la **Figura 27** se muestran los módulos de la aplicación informática desarrollada para el presente estudio, entre los que figura el módulo de Gestión de Proyectos de Desarrollo de Software, Mantenimiento de Sistemas, Inventario de Sistemas y Configuración del sistema.

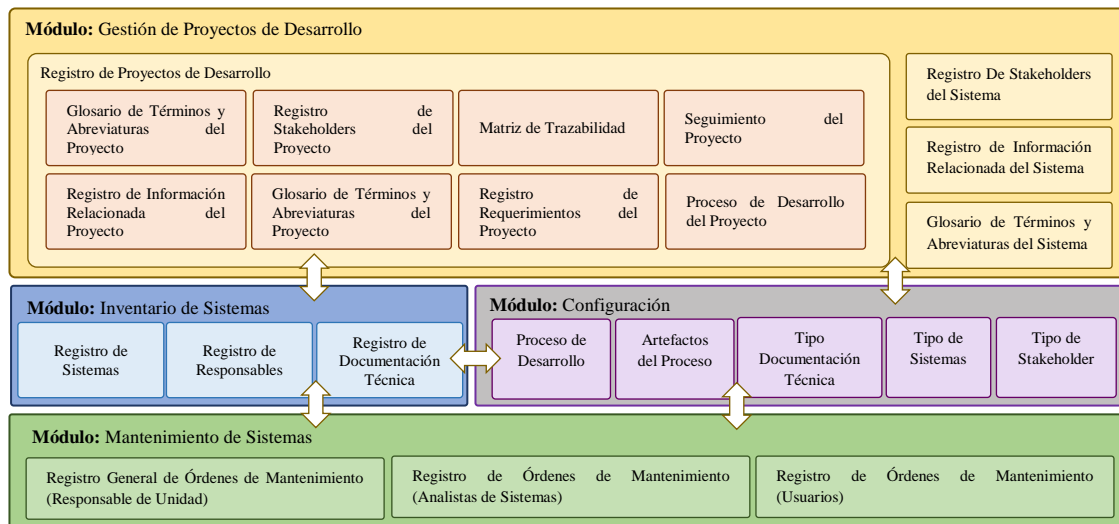


Figura 27: Diagrama de módulos de la aplicación informática

Fuente: Elaborado por la autora

4.3.1. Módulos de la aplicación informática

La aplicación informática tiene los siguientes módulos: Gestión de Proyectos de Desarrollo, Mantenimiento de Sistemas, Inventario de Sistemas y Configuración del Sistema, tal como se muestra en la **Figura 28**.

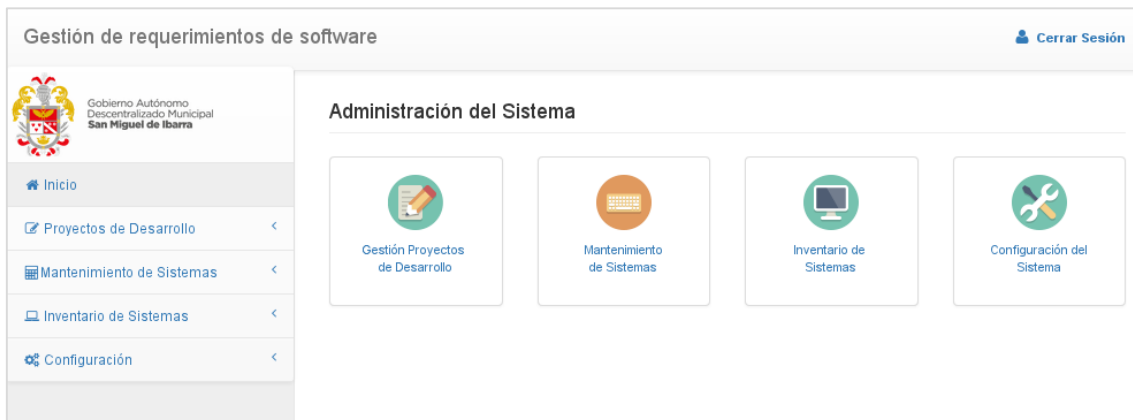


Figura 28: Pantalla - Módulos de la aplicación

Fuente: Elaborado por la autora

4.3.2. Descripción del Módulo de Configuración del Sistema

El módulo de Configuración del Sistema tiene los siguientes sub módulos: Proceso de Desarrollo, Artefactos del Proceso, Tipo de Documentación Técnica, Tipo de Sistemas y Tipo de Stakeholders (**Figura 29**).

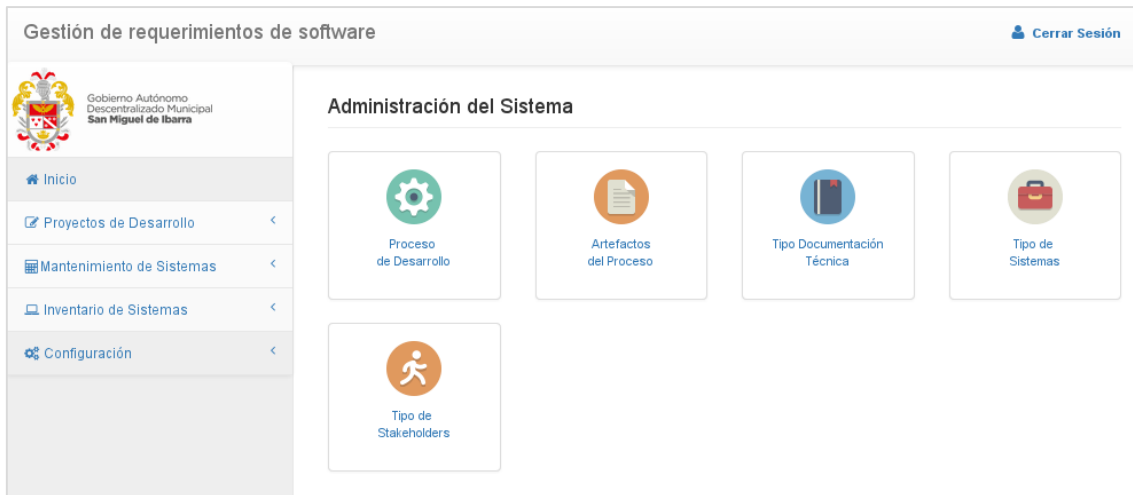


Figura 29: Pantalla - Sub módulos Configuración del Sistema

Fuente: Elaborado por la autora

El sub módulo de Proceso de Desarrollo tiene las funcionalidades mostradas en la **Tabla 67**.

Tabla 67: Funcionalidades módulo Proceso de Desarrollo

Fuente: Elaborado por la autora

FUNCIONALIDAD	DESCRIPCIÓN
<i>Visualizar el Proceso de Desarrollo de Software</i>	<ul style="list-style-type: none"> • Muestra un listado de las etapas, actividades y tareas del proceso registrado con el nombre, descripción y tipo de cada una (Figura 30). • Permite navegar entre los diferentes niveles del proceso registrado. • Permite visualizar la descripción de los artefactos de cada actividad o tarea.
<i>Agregar y editar registros del proceso</i>	<ul style="list-style-type: none"> • Presenta un formulario para registrar o editar etapas, actividades o tareas del proceso de desarrollo con los siguientes campos: Nombre, Tipo, Descripción, Registro padre, Orden y Estado del requerimiento.
<i>Agregar y editar artefactos del proceso</i>	<ul style="list-style-type: none"> • Presenta un formulario para agregar artefactos del catálogo de artefactos a las actividades o tareas (Figura 31).

Proceso de Desarrollo de Software					
Ruta Actual: Administración del Sistema / Configuración del Sistema / Proceso de Desarrollo de Software					
Proceso de Desarrollo de Software + Agregar Registro ← Regresar					
N°	Nombre	Tipo	Artefactos	Descripción	Opciones
1	PLANIFICAR REUNIONES DE DESPEGUE	TAREA	CONVOCATORIA A REUNIÓN	<ul style="list-style-type: none"> COORDINAR LAS FECHAS Y HORAS DE LAS REUNIONES DE DESPEGUE CON LA DIRECCIÓN O UNIDAD ADMINISTRATIVA QUE REALIZÓ LA PETICIÓN DE DESARROLLO DE SOFTWARE. UBICAR Y SOLICITAR LOS RECURSOS NECESARIOS PARA LAS REUNIONES DE DESPEGUE, TALES COMO EQUIPOS E INSTALACIONES. ELABORAR Y ENVIAR LAS CONVOCATORIAS PARA LAS REUNIONES DE DESPEGUE A LAS PARTES INVOLUCRADAS. 	✎ Editar + Artefactos
2	EJECUTAR REUNIONES DE DESPEGUE	TAREA	ACTA DE REUNIÓN REGISTRO DE STAKEHOLDERS	REUNIRSE CON LOS INTERESADOS DEL PROYECTO PARA REALIZAR LAS SIGUIENTES TAREAS: <ul style="list-style-type: none"> DETERMINAR EL PROBLEMA Y OBJETIVOS DEL PROYECTO: SOLICITAR AL STAKEHOLDER QUE REALIZÓ LA PETICIÓN, QUE EXPLIQUE EL PROBLEMA QUE TIENE EN SU ÁREA DE TRABAJO Y LOS OBJETIVOS (REQUERIMIENTOS) DE NEGOCIO QUE QUIERE ALCANZAR CON EL PROYECTO. DETERMINAR EL ALCANCE DEL PROYECTO: MEDIANTE LA ELABORACIÓN DE UN 	✎ Editar + Artefactos

Figura 30: Pantalla - Proceso de Desarrollo de Software

Fuente: Elaborado por la autora

Proceso de Desarrollo de Software			
Ruta Actual: Administración del Sistema / Configuración del Sistema / Proceso de Desarrollo de Software			
Agregar artefactos			
Nombre:	PLANIFICAR REUNIONES DE DESPEGUE	Tipo:	TAREA
Descripción:	<ul style="list-style-type: none"> COORDINAR LAS FECHAS Y HORAS DE LAS REUNIONES DE DESPEGUE CON LA DIRECCIÓN O UNIDAD ADMINISTRATIVA QUE REALIZÓ LA PETICIÓN DE DESARROLLO DE SOFTWARE. UBICAR Y SOLICITAR LOS RECURSOS NECESARIOS PARA LAS REUNIONES DE DESPEGUE, TALES COMO EQUIPOS E INSTALACIONES. ELABORAR Y ENVIAR LAS CONVOCATORIAS PARA LAS REUNIONES DE DESPEGUE A LAS PARTES INVOLUCRADAS. 		
Seleccionar Artefacto:	<input type="text" value="ACTA DE REUNIÓN"/>	+ Agregar	← Regresar
Listado de Artefactos			
Nombre	Descripción	Opciones	
CONVOCATORIA A REUNIÓN	LA CONVOCATORIA SE DEBE REALIZAR FORMALMENTE A TRAVÉS DEL SISTEMA DOCUMENTAL QUIPUX, MEDIANTE UN MEMORANDO (INTERNO) U OFICIO (EXTERNO).	✎ Eliminar	

Figura 31: Pantalla - Agregar y Editar artefactos al proceso

Fuente: Elaborado por la autora

El sub módulo de Artefactos del Proceso tiene las funcionalidades mostradas en la **Tabla 68**.

Tabla 68: Funcionalidades módulo Artefactos del Proceso

Fuente: Elaborado por la autora

FUNCIONALIDAD	DESCRIPCIÓN
Visualizar el catálogo de artefactos	<ul style="list-style-type: none"> Muestra un listado de los artefactos cargados al sistema con nombre, descripción y descarga de plantilla (Figura 32).
Agregar y editar artefactos al catálogo	<ul style="list-style-type: none"> Muestra un formulario para el registro o edición de artefactos del catálogo, con campos de: nombre, descripción, directorio, carga de plantilla, activo/no activo, virtual o no virtual que sirve para identificar si un artefacto debe ser cargado en el sistema o es una funcionalidad existente (Figura 33).

Listado de Artefactos

Ruta Actual: [Administración del Sistema](#) / [Configuración del Sistema](#) / Listado de Artefactos

Listado de Artefactos

[+ Agregar Registro](#)
[← Regresar](#)

Nombre	Descripción	Directorio	Plantilla	Es virtual	Opciones
ACTA DE REUNIÓN	DOCUMENTO EN EL QUE SE REGISTRA INFORMACIÓN DE LA REUNIÓN COMO: FECHA, HORA, LUGAR, OBJETIVO, LISTA DE CONVOCADOS, TEMAS TRATADOS, COMPROMISOS ADQUIRIDOS, LISTADO DE ASISTENTES	ACTA	ActaDeReunion.doc	✘	✎ Editar
ACTA DE TÉRMINO DE PROYECTOS	DOCUMENTO QUE PERMITE REGISTRAR EL TÉRMINO DE UN PROYECTO CON LAS FIRMAS DE ACEPTACIÓN DE LOS USUARIOS	ACTA_TERMINO	ActaTermino.doc	✘	✎ Editar
CONVOCATORIA A REUNIÓN	LA CONVOCATORIA SE DEBE REALIZAR FORMALMENTE A TRAVÉS DEL SISTEMA DOCUMENTAL QUIPUX, MEDIANTE UN MEMORANDO (INTERNO) U OFICIO (EXTERNO).	CONVOCATORIA		✘	✎ Editar
FORMATO DE ENTREVISTA PARA LEVANTAR INFORMACIÓN	FORMATO QUE PERMITE REGISTRAR LA INFORMACIÓN DE UN PROCESO, MISMA QUE SE HA LEVANTADO CON UNA ENTREVISTA.	ENTREVISTA		✘	✎ Editar

Figura 32: Pantalla - Catálogo de Artefactos

Fuente: Elaborado por la autora

Registro de Artefactos

Ruta Actual: Administración del Sistema / Configuración del Sistema / Editar Artefacto

Editar Artefacto

Nombre del Artefacto:*

Descripción:*

Directorio:*

Formato* Archivo subido: ActaDeReunion.doc
 No file selected.

Seleccione para borrar archivo existente

Activo*

Es virtual*

Figura 33: Pantalla - Registro de Artefactos
Fuente: Elaborado por la autora

El sub módulo de Tipo de Documentación Técnica presenta el catálogo de tipos de documentación técnica, así como funcionalidades para registro y edición de nuevos tipos, tal como se puede ver en la **Figura 34**. Este catálogo sirve para cargar la documentación técnica de sistemas en el módulo de Inventario de Sistemas.

Tipo Documentación Técnica

Ruta Actual: Administración del Sistema / Configuración del Sistema / Tipo Documentación Técnica

Tipo Documentación Técnica + Agregar Registro

Número	Descripción	Opciones
1	DIAGRAMA DE ARQUITECTURA	<input type="button" value="Editar"/>
2	MANUAL DE INSTALACIÓN	<input type="button" value="Editar"/>
3	MANUAL TÉCNICO	<input type="button" value="Editar"/>
4	MANUAL USUARIO	<input type="button" value="Editar"/>

4 Tipo(s)

Figura 34: Pantalla – Catálogo de Tipo de Documentación Técnica
Fuente: Elaborado por la autora

El sub módulo de Tipo de Sistemas presenta el catálogo de tipos de sistemas existentes, así como funcionalidades para registro y edición de nuevos tipos, tal como se puede ver en la **Figura 35**. Este catálogo sirve para seleccionar el tipo de sistema al registrar uno nuevo en el módulo de Inventario de Sistemas.

Tipo de Sistemas		
Ruta Actual: Administración del Sistema / Configuración del Sistema / Tipo de Sistemas		
Tipo de Sistemas		+ Agregar Registro ← Regresar
Número	Descripción	Opciones
1	DESARROLLO INTERNO	Editar
2	IMPLEMENTACIÓN DE SOFTWARE LIBRE	Editar
3	SISTEMA ADQUIRIDO	Editar
3 Tipo(s)		

Figura 35: Pantalla – Catálogo de Tipo de Sistemas

Fuente: Elaborado por la autora

El sub módulo de Tipo de Stakeholders presenta el catálogo de tipos de stakeholders existentes, así como funcionalidades para registro y edición de nuevos tipos, tal como se puede ver en la **Figura 36**. Este catálogo sirve para el registro de stakeholders en el módulo de Gestión de Proyectos de Desarrollo.

Tipo de Stakeholders			
Ruta Actual: Administración del Sistema / Configuración del Sistema / Tipo de Stakeholders			
Tipo de Stakeholders			+ Agregar Registro ← Regresar
Número	Descripción	Detalle	Opciones
1	CONSULTOR	EMPRESA EXTERNA QUE PRESTA UN BIEN O SERVICIO A LA ENTIDAD A TRAVÉS DE UN CONTRATO	Editar
2	CONTRIBUYENTE	CIUDADANO QUE UTILIZA LOS SISTEMAS Y SERVICIOS PROVISTOS POR EL GAD-I.	Editar
3	MIEMBRO DEL EQUIPO DE DESARROLLO	FUNCIONARIOS OPERATIVOS DE LA DIRECCIÓN DE TIC QUE REALIZAN LAS ACTIVIDADES DE ANÁLISIS, DISEÑO, CONSTRUCCIÓN, PRUEBAS, IMPLANTACIÓN, FUNCIONARIOS DIRECTIVOS DE LA DIRECCIÓN DE TIC QUE REALIZAN ACTIVIDADES DE GESTIÓN DEL PROYECTO.	Editar
4	PATROCINADOR	ORGANIZACIÓN EXTERNA QUE FINANCIA TODO O PARTE DE UN PROYECTO.	Editar

Figura 36: Pantalla - Catálogo Tipo de Stakeholders

Fuente: Elaborado por la autora

4.3.3. Descripción del Módulo de Inventario de Sistemas

El módulo de Inventario de Sistemas tiene las funcionalidades mostradas en la **Tabla 69**.

Tabla 69: Funcionalidades módulo de Inventario de Sistemas

Fuente: Elaborado por la autora

FUNCIONALIDAD	DESCRIPCIÓN
Visualizar el Inventario de Sistemas	<ul style="list-style-type: none"> Muestra un listado de los sistemas registrados con nombre corto, nombre descriptivo, tipo, activo/no activo (Figura 37).
Agregar y editar registros al inventario	<ul style="list-style-type: none"> Muestra un formulario para registro o edición de sistemas al inventario con datos de nombre del sistema, nombre corto, descripción, características técnicas, tipo, activo/no activo.
Visualizar y modificar detalles del Sistema	<ul style="list-style-type: none"> Muestra la descripción del sistema: nombre, descripción, nombre corto, características técnicas, tipo, activo/no activo, detalle de analistas responsables, detalle de documentación técnica (Figura 38). Tiene las opciones para registrar o eliminar los analistas responsables y la documentación técnica del sistema como manuales y diagramas.

Inventario de Sistemas

Ruta Actual: [Administración del Sistema](#) / Inventario de Sistemas

[+ Agregar Sistema](#) [← Regresar](#)

Nombre Corto	Nombre	Tipo	Activo	Opciones
APP REG	APLICACIÓN CONSULTA REGISTRO CIVIL	DESARROLLO INTERNO	✓	Editar
GIS CENSO	APLICACIÓN DE INFORMACIÓN GEOGRÁFICA CENSO CENTRO HISTÓRICO	DESARROLLO INTERNO		Editar
GIS PDOT	APLICACIÓN DE INFORMACIÓN GEOGRÁFICA CENSO PLAN DE ORDENAMIENTO TERRITORIAL	DESARROLLO INTERNO		Editar
KIOSKO	APLICACION DE KIOSKO ELECTRÓNICO	DESARROLLO INTERNO	✓	Editar
APPTM	APLICACIÓN DE TEMÁTICOS CATASTROS	DESARROLLO INTERNO		Editar

Figura 37: Pantalla - Inventario de Sistemas

Fuente: Elaborado por la autora








Datos del Sistema			
Ruta Actual: Administración del Sistema / Inventario de Sistemas / Detalle del Sistema			
Datos del Sistema			
Nombre:	APLICACIÓN CONSULTA REGISTRO CIVIL	Nombre corto:	APP REG
Tipo:	DESARROLLO INTERNO	Activo:	✓
Descripción:	ESTA APLICACIÓN PERMITE CONSUMIR SERVICIOS WEB, PUBLICADOS POR EL REGISTRO CIVIL, PARA CONSULTAR INFORMACIÓN DE LOS CIUDADANOS, A TRAVÉS DE LA CÉDULA O NOMBRES, A TRAVÉS DE UN CONVENIO FIRMADO ENTRE LAS 2 PARTES		
Características técnicas:	SERVIDOR WEB: APACHE HTTP 2 FRAMEWORK: EXTJS 4.1.1, SYMFONY 1.4 ORM: DOCTRINE 1.2 BASES DE DATOS: POSTGRESQL 9.1 LIBRERÍAS: SFTCPDFPLUGIN		
Analistas de Sistemas responsables			
Nombres	Apellidos	Opciones	
CRISTIAN FABRICIO	ROMERO BENAVIDES	 Eliminar	
VERÓNICA AZUCENA	ROSERO VINUEZA	 Eliminar	
Documentación Técnica			
Tipo	Archivo	Opciones	
MANUAL TÉCNICO	Manual_Tecnico_ConsultaRegistroCivil.doc	 Editar  Eliminar	
MANUAL USUARIO	Manual_Usuario_ConsultaRegistroCivil.doc	 Editar  Eliminar	

Figura 38: Pantalla - Datos del Sistema
Fuente: Elaborado por la autora

4.3.4. Descripción del Módulo de Mantenimiento de Sistemas

El módulo de Mantenimiento de Sistemas tiene los siguientes sub módulos: Órdenes de Mantenimiento, Mis Órdenes de Mantenimiento y Registro de Orden de Mantenimiento (Figura 39).


Gestión de requerimientos de software Cerrar Sesión




Gobierno Autónomo
Descentralizado Municipal
San Miguel de Ibarra

- [Inicio](#)
- [Proyectos de Desarrollo](#)
- [Mantenimiento de Sistemas](#)
- [Inventario de Sistemas](#)
- [Configuración](#)


Administración del Sistema



Órdenes de
Mantenimiento



Mis Órdenes de
Mantenimiento



Registrar Orden
de Mantenimiento

Figura 39: Pantalla - Módulos Mantenimiento de Sistemas
Fuente: Elaborado por la autora

El sub módulo de Registro de Orden de Mantenimiento es para el usuario final, quien puede registrar una nueva orden de mantenimiento (descripción corta, solicitante, descripción del requerimiento, archivo adjunto, sistema y módulos relacionados), visualizar el listado de órdenes registradas por él (**Figura 40**), así como visualizar el estado y avance de cada una de ellas (**Figura 41**).

Registro de Órden de Mantenimiento

Ruta Actual: [Administración del Sistema](#) / [Mantenimiento de Sistemas](#) / Registro de Órden de Mantenimiento

Registro de Órdenes de Mantenimiento						Agregar Registro	Regresar
Código	Fecha	Descripción Corta	Sistema	Estado	Opciones		
3	2017-05-05	ACTUALIZAR REPORTES CONSULTA SRI	APLICACIÓN WEB DE REPORTES SRI	IMPLEMENTADA/CERRADA		Abrir	
1	2017-06-01	CRUCE DE BASE DE DATOS DEL SRI	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	IMPLEMENTADA/CERRADA		Abrir	
4	2017-08-30	REPORTE DE CONTRIBUYENTES OMISOS	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	IMPLEMENTADA/CERRADA		Abrir	

3 Orden(es)

Figura 40: Pantalla - Listado de Órdenes de Mantenimiento (usuario)

Fuente: Elaborado por la autora

El sub módulo de Registro de Orden de Mantenimiento es para el Responsable de Desarrollo de Software, quien puede ver el listado de las órdenes enviadas por los usuarios (**Figura 42**), visualizar el detalle de cada una de ellas (**Figura 43**) y designar al analista de sistemas encargado de su análisis y atención (**Figura 44**). El estado de las órdenes de mantenimiento va cambiando automáticamente de acuerdo a las actividades que se vayan realizando dentro del proceso definido.

Registro de Orden de Mantenimiento

Ruta Actual: [Administración del Sistema](#) / [Mantenimiento de Sistemas](#) / Registro de Orden de Mantenimiento

ORDEN DE MANTENIMIENTO / PEDIDOS VARIOS DE SOFTWARE			
Código:	3	Fecha:	2017-05-05
Solicitante:	REA LOZADA ROSA ANDREA	Cargo Solicitante:	SERVIDOR PÚBLICO
Sistema:	APLICACIÓN WEB DE REPORTES SRI	Módulos:	CONSULTA CONTRIBUYENTES
Descripción Corta:	ACTUALIZAR REPORTES CONSULTA SRI	Estado:	IMPLEMENTADA/CERRADA
Archivo adjunto:		Analista Designado:	REA LOZADA ROSA ANDREA
Descripción Requerimiento:	REALIZAR LA ACTUALIZACIÓN DEL REPORTE DE CONTRIBUYENTES EN EL SISTEMA DE CONSULTA DE DATOS DEL SRI, EN LAS SIGUIENTES SECCIONES: - DATOS DE DECLARACIONES 104 DEL AÑO 2010 AL 2013. - DATOS DE DECLARACIONES 104 DEL AÑO 2014, 2015, 2016 Y 2017. - DATOS DE ESTABLECIMIENTOS: NÚMERO, NOMBRE FANTASÍA, DIRECCIÓN, ESTADO Y TIPO. - DATOS DE CONTACTOS DE ESTABLECIMIENTOS: NÚMERO, CONTACTO (TELÉFONO, CELULAR O CORREO)		
ANÁLISIS DEL REQUERIMIENTO			
Factible:	✓	Fecha:	2017-08-24
Justificación:	SE CUENTA CON LA RÉPLICA DE BASE DE DATOS		
ATENCIÓN DEL REQUERIMIENTO			
Fecha Inicio:	2017-05-05	Fecha Fin:	2017-05-08
Fecha de entrega:	2017-05-08	Archivo adjunto:	Descargar
Descripción trabajo realizado:	DEBIDO A LOS CAMBIOS EN LAS ESTRUCTURAS DE DATOS DEL SRI, SE TUVO PROBLEMAS EN LA OBTENCIÓN DE DATOS DE DECLARACIONES 104. UNA VEZ SOLUCIONADO EL PROBLEMA FUE NECESARIO ACTUALIZAR LOS SQL DEL REPORTE DEL SISTEMA (SISTEMAS.IMI.GOB.EC) DE ACUERDO AL DETALLE QUE SE ADJUNTA		
Fecha Inicio Pruebas:		Fecha Fin Pruebas:	

Figura 41: Pantalla - Detalle Orden de Mantenimiento
Fuente: Elaborado por la autora

Registro de Orden de Mantenimiento

Ruta Actual: [Administración del Sistema](#) / [Mantenimiento de Sistemas](#) / Registro de Orden de Mantenimiento

Registro de Órdenes de Mantenimiento [Regresar](#)

Código	Fecha	Descripción Corta	Sistema	Analista Asignado	Estado	Opciones
3	2017-05-05	ACTUALIZAR REPORTES CONSULTA SRI	APLICACIÓN WEB DE REPORTES SRI	REA LOZADA ROSA ANDREA	IMPLEMENTADA/CERRADA	Abrir
1	2017-06-01	CRUCE DE BASE DE DATOS DEL SRI	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	REA LOZADA ROSA ANDREA	IMPLEMENTADA/CERRADA	Abrir
4	2017-08-30	REPORTE DE CONTRIBUYENTES OMISOS	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS		REGISTRADA	Designar Analista Abrir

Figura 42: Pantalla - Órdenes de Mantenimiento (administrador)
Fuente: Elaborado por la autora

Registro de Orden de Mantenimiento

Ruta Actual: [Administración del Sistema](#) / [Mantenimiento de Sistemas](#) / Registro de Orden de Mantenimiento


ORDEN DE MANTENIMIENTO / PEDIDOS VARIOS DE SOFTWARE			
Código:	4	Fecha:	2017-08-30
Solicitante:	REA LOZADA ROSA ANDREA	Cargo Solicitante:	SERVIDOR PÚBLICO
Sistema:	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	Módulos:	RÉPLICA DE DATOS
Descripción Corta:	REPORTE DE CONTRIBUYENTES OMISOS	Estado:	REGISTRADA
Archivo adjunto:		Analista Designado:	
Descripción Requerimiento:	SOLICITA REPORTE DE LOS CONTRIBUYENTES OMISOS QUE EXISTEN HASTA LA PRESENTE FECHA, INFORMACIÓN QUE SE REQUIERE CON TODOS LOS DATOS QUE PROPORCIONA EL SRI, CON RESPECTO A LOS CONTRIBUYENTES OBLIGADOS A LLEVAR CONTABILIDAD EN LOS QUE SE ENCONTRARAN LAS PERSONAS NATURALES OBLIGADAS A LLEVAR CONTABILIDAD FORMULARIO 102 Y LAS SOCIEDADES FORMULARIO 101, PETICIÓN QUE REALIZA POR NO SER POSIBLE OBTENERLO DESDE EL NUEVO SISTEMA BYPROS. SE REQUIERE LOS SIGUIENTES DATOS: ANIO_FISCAL, TOTAL_ACTIVADO_830, TOT_PASIVO_CORRIENTE_1030, TOT_PASIVO_LARGO_PLAZO_1250, TOT_PATRIMONIO_NETO_1330, TOTAL_INGRESOS_1440, TOTAL_COSTOS_GASTOS_2760, UTILIDAD_NETA_EJERCICIO_2800, PERDIDA_EJERCICIO_2810, SUB_ING_RGR_TYC_SRD_3200, SUB_GTO_DED_TYC_SRD_3210.		
ANÁLISIS DEL REQUERIMIENTO			
No se han registrado datos aún			
ATENCIÓN DEL REQUERIMIENTO			
No se han registrado datos aún			
			

Figura 43: Pantalla - Detalle de Orden de Mantenimiento
Fuente: Elaborado por la autora

Registro de Orden de Mantenimiento

Ruta Actual: [Administración del Sistema](#) / [Mantenimiento de Sistemas](#) / Registro de Orden de Mantenimiento

Designar Analista de Sistemas			
Código:	4	Fecha:	2017-08-30
Solicitante:	REA LOZADA ROSA ANDREA	Cargo Solicitante:	SERVIDOR PÚBLICO
Sistema:	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	Módulos:	RÉPLICA DE DATOS
Descripción Corta:	REPORTE DE CONTRIBUYENTES OMISOS	Estado:	REGISTRADA
Descripción Requerimiento:	SOLICITA REPORTE DE LOS CONTRIBUYENTES OMISOS QUE EXISTEN HASTA LA PRESENTE FECHA, INFORMACIÓN QUE SE REQUIERE CON TODOS LOS DATOS QUE PROPORCIONA EL SRI, CON RESPECTO A LOS CONTRIBUYENTES OBLIGADOS A LLEVAR CONTABILIDAD EN LOS QUE SE ENCONTRARAN LAS PERSONAS NATURALES OBLIGADAS A LLEVAR CONTABILIDAD FORMULARIO 102 Y LAS SOCIEDADES FORMULARIO 101, PETICIÓN QUE REALIZA POR NO SER POSIBLE OBTENERLO DESDE EL NUEVO SISTEMA BYPROS. SE REQUIERE LOS SIGUIENTES DATOS: ANIO_FISCAL, TOTAL_ACTIVADO_830, TOT_PASIVO_CORRIENTE_1030, TOT_PASIVO_LARGO_PLAZO_1250, TOT_PATRIMONIO_NETO_1330, TOTAL_INGRESOS_1440, TOTAL_COSTOS_GASTOS_2760, UTILIDAD_NETA_EJERCICIO_2800, PERDIDA_EJERCICIO_2810, SUB_ING_RGR_TYC_SRD_3200, SUB_GTO_DED_TYC_SRD_3210.		
Designar Analista de Sistemas			
Analista de Sistemas *	<input type="text" value="ALVAREZ HERNANDEZ JAIRO GUSTAVO"/>		
 			

Figura 44: Pantalla - Registro de Analista de Sistemas en Orden
Fuente: Elaborado por la autora

El sub módulo de Mis Órdenes de Mantenimiento es para los Analistas de Sistemas, quien puede ver el listado de las órdenes designadas a él (**Figura 45**), visualizar el detalle de cada una para registrar su análisis con datos de factible/no factible – justificación (**Figura 46**) y su atención con datos de fecha de inicio, fecha de finalización, descripción del trabajo realizado, fecha inicio de pruebas, fecha fin de pruebas, observaciones, fecha entrega, archivos adjuntos (**Figura 47**). Posteriormente puede imprimir el documento de orden de mantenimiento (**Figura 48**).

Registro de Orden de Mantenimiento						
Ruta Actual: Administración del Sistema / Mantenimiento de Sistemas / Registro de Orden de Mantenimiento						
Registro de Órdenes de Mantenimiento						Regresar
Código	Fecha	Descripción Corta	Sistema	Analista Asignado	Estado	Opciones
3	2017-05-05	ACTUALIZAR REPORTES CONSULTA SRI	APLICACIÓN WEB DE REPORTES SRI	REA LOZADA ROSA ANDREA	IMPLEMENTADA/CERRADA	Abrir
1	2017-06-01	CRUCE DE BASE DE DATOS DEL SRI	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	REA LOZADA ROSA ANDREA	IMPLEMENTADA/CERRADA	Abrir
4	2017-08-30	REPORTE DE CONTRIBUYENTES OMISOS	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	REA LOZADA ROSA ANDREA	EN REVISIÓN	Abrir
3 Orden(es)						← ↶ ↷ →

Figura 45: Pantalla - Mis Órdenes de Mantenimiento
Fuente: Elabora por la autora

Registro de Orden de Mantenimiento			
Ruta Actual: Administración del Sistema / Mantenimiento de Sistemas / Mis Órdenes de Mantenimiento			
ORDEN DE MANTENIMIENTO / PEDIDOS VARIOS DE SOFTWARE			
Código:	4	Fecha:	2017-08-30
Solicitante:	REA LOZADA ROSA ANDREA	Cargo Solicitante:	SERVIDOR PÚBLICO
Sistema:	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	Módulos:	RÉPLICA DE DATOS
Descripción Corta:	REPORTE DE CONTRIBUYENTES OMISOS	Estado:	EN REVISIÓN
Archivo adjunto:		Analista Designado:	REA LOZADA ROSA ANDREA
Descripción Requerimiento:	SOLICITA REPORTE DE LOS CONTRIBUYENTES OMISOS QUE EXISTEN HASTA LA PRESENTE FECHA, INFORMACIÓN QUE SE REQUIERE CON TODOS LOS DATOS QUE PROPORCIONA EL SRI, CON RESPECTO A LOS CONTRIBUYENTES OBLIGADOS A LLEVAR CONTABILIDAD EN LOS QUE SE ENCONTRARAN LAS PERSONAS NATURALES OBLIGADAS A LLEVAR CONTABILIDAD FORMULARIO 102 Y LAS SOCIEDADES FORMULARIO 101, PETICIÓN QUE REALIZA POR NO SER POSIBLE OBTENERLO DESDE EL NUEVO SISTEMA BYPROS. SE REQUIERE LOS SIGUIENTES DATOS: ANIO_FISCAL, TOTAL_ACTIVIVO_830, TOT_PASIVO_CORRIENTE_1030, TOT_PASIVO_LARGO_PLAZO_1250, TOT_PATRIMONIO_NETO_1330, TOTAL_INGRESOS_1440, TOTAL_COSTOS_GASTOS_2760, UTILIDAD_NETA_EJERCICIO_2800, PERDIDA_EJERCICIO_2810, SUB_ING_RGR_TYC_SRD_3200, SUB_GTO_DED_TYC_SRD_3210.		
ANÁLISIS DEL REQUERIMIENTO			
Registrar datos de análisis			
ATENCIÓN DEL REQUERIMIENTO			
Registrar datos de atención			
Regresar		Imprimir	

Figura 46: Pantalla - Registro de análisis y atención de orden
Fuente: Elaborado por la autora

Registro de Orden de Mantenimiento

Ruta Actual: Administración del Sistema / Mantenimiento de Sistemas / Registro de Orden de Mantenimiento

Registro de atención del requerimiento

Fecha de Inicio: 2017-04-19 Fecha de Finalización: 2017-04-19

Descripción del trabajo realizado:
 - CONTRIBUYENTES101_OBLIGADOSOMISOS_19-04-2017.XLS: REPORTE DE OMISOS CON SOCIEDADES Y DATOS DE DECLARACIONES 101.
 - CONTRIBUYENTES102_OBLIGADOSOMISOS_19-04-2017.XLS: REPORTE DE OMISOS CON PERSONAS NATURALES OBLIGADAS A LLEVAR CONTABILIDAD Y DATOS DE

Fecha de Inicio de Pruebas: aaaa-mm-dd Fecha de Fin de Pruebas: aaaa-mm-dd

Observaciones del periodo de pruebas: NO APLICA

Fecha de Entrega: 2017-04-19

Archivo: Choose File Orden2 ReportesRentas 19-04-2017.pdf

Guardar Cancelar

Figura 47: Pantalla - Registro de atención orden
 Fuente: Elaborado por la autora

GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPAL DE SAN MIGUEL DE IBARRA			
ORDEN DE MANTENIMIENTO/ PEDIDOS VARIOS DE SOFTWARE		CÓDIGO	VERSIÓN
			1
		FECHA APROBACIÓN	01/10/2012
ORDEN DE MANTENIMIENTO / PEDIDOS VARIOS DE SOFTWARE			
Nro. Orden:	1	Fecha:	2017-06-01
Solicitante:	REALOZADA ROSA ANDREA	Cargo Solicitante:	SERVIDOR PÚBLICO
Sistema:	SISTEMA RÉPLICA DE BASE DE DATOS ORACLE DELSERVICIO DE RENTAS INTERNAS	Módulos:	RÉPLICA DE BASE DE DATOS
Descripción Corta:	CRUCE DE BASE DE DATOS DEL SRI	Estado:	IMPLEMENTADA/CERRADA
Analista Designado:	REALOZADA ROSA ANDREA		
Descripción Requerimiento:	SE SOLICITA UN CRUCE DE LA BASE DE DATOS DEL SRI PARA IDENTIFICAR SI ESTÁN EN LISTA BLANCA DE LOS CONTRIBUYENTES QUE SE ENCUENTRAN EN LA BASE DE DATOS Y FAVOR NOS AYUDE CON LOS CAMPOS DE NÚMERO TELEFÓNICO Y DIRECCIÓN DE CORREO ELECTRÓNICO PARA ASÍ HACER EFECTIVA LA NOTIFICACIÓN.		
ANÁLISIS DEL REQUERIMIENTO			
Factible:	SI	Fecha:	2017-08-24
Justificación:	SE CUENTA CON LA RÉPLICA DE DATOS DEL SRI		
ATENCIÓN DEL REQUERIMIENTO			
Fecha Inicio:	2017-06-02	Fecha Fin:	2017-06-02
Fecha de entrega:	2017-06-02	Archivo adjunto:	
Descripción trabajo realizado:	SE REALIZÓ EL CRUCE DE LOS DATOS ENTREGADOS POR EL SOLICITANTE CON LOS DATOS DE CONTRIBUYENTES, ESTABLECIMIENTOS Y MEDIOS DE CONTACTO DEL SRI. ÚNICAMENTE SE CONSIDERARON LOS CONTRIBUYENTES CON ESTABLECIMIENTOS ABIERTOS EN IBARRA. EN EL CRUCE ENVIADO SE ANEXARON LOS DATOS DE LISTA BLANCA (S/NI) Y DE CONTACTOS ENCONTRADOS (TELÉFONO FIJO, CELULAR, CORREO ELECTRÓNICO). SE DEBE CONSIDERAR QUE EN EL SRI SE REGISTRAN CONTACTOS POR ESTABLECIMIENTO REGISTRADO, PERO EN LOS DATOS ENVIADOS PARA EL CRUCE NO EXISTÍA EL CAMPO DE ESTABLECIMIENTO, POR ELLO SE ENVIAN TODOS LOS CONTACTOS ENCONTRADOS A NIVEL DE RUC.		
Fecha Inicio Pruebas:		Fecha Fin Pruebas:	
Observaciones de las pruebas:	NINGUNA		
CONSTANCIA DE ENTREGA DEL REQUERIMIENTO			
REALOZADA ROSA ANDREA ANALISTA DE SISTEMAS	REALOZADA ROSA ANDREA SOLICITANTE	Lcda. Sonia Bossano Subia RESPONSABLE DE DESARROLLO DE SOFTWARE	

Figura 48: Pantalla - Impresión de Orden de Mantenimiento
 Fuente: Elaborado por la autora

4.3.5. Descripción del Módulo de Gestión de Proyectos de Desarrollo

El módulo de Gestión de Proyectos de Desarrollo tiene los siguientes sub módulos: Registro de Proyectos, Registro de Stakeholders, Información Relacionada y Glosario de Términos (Figura 49). El Glosario de Términos permite el ingreso y edición de términos y abreviaturas con campo, tipo y descripción (Figura 50) disponibles para todos los proyectos.

El Registro de Información Relacionada permite el ingreso y edición de normativa legal, documentación de procesos y registros organizacionales con título, tipo, autor, fecha de expedición (Figura 51) disponibles para todos los proyectos. El Registro de Stakeholders permite el ingreso y edición de stakeholders disponibles para todos los proyectos con campos de nombres, apellidos, cargo, dirección, tipo, teléfono y correo electrónico (Figura 52).

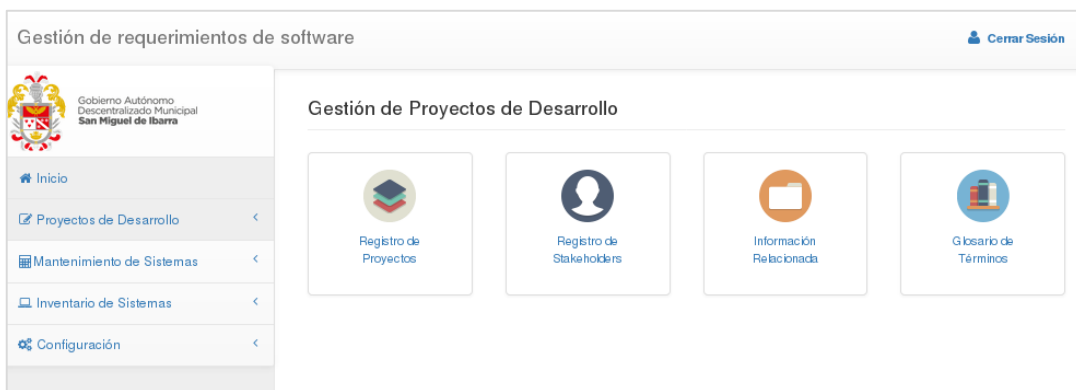


Figura 49: Pantalla - Módulo de Gestión de Proyectos de Desarrollo

Fuente: Elaborado por la autora

Glosario de Términos y Abreviaturas			
Ruta Actual: Administración del Sistema / Gestión de Proyectos de Desarrollo / Glosario de Términos y Abreviaturas			
Glosario de Términos y Abreviaturas			+ Agregar Registro ← Regresar
Campo	Tipo	Descripción	Opciones
ACTIVIDAD ECONÓMICA	TÉRMINO	SE LLAMA ACTIVIDAD ECONÓMICA A CUALQUIER PROCESO DONDE SE GENERAN E INTERCAMBIAN PRODUCTOS, BIENES O SERVICIOS PARA CUBRIR LAS NECESIDADES DEL HOMBRE. LA ACTIVIDAD ECONÓMICA PERMITE LA GENERACIÓN DE RIQUEZA DENTRO DE UNA COMUNIDAD (CIUDAD, REGIÓN, PAÍS) MEDIANTE LA EXTRACCIÓN, TRANSFORMACIÓN Y DISTRIBUCIÓN DE LOS RECURSOS NATURALES O BIEN DE ALGÚN TIPO DE SERVICIO.	Editar Eliminar
CATASTRO	TÉRMINO	ES EL INVENTARIO O CENSO, DEBIDAMENTE ACTUALIZADO Y CLASIFICADO, DE LOS BIENES INMUEBLES PERTENECIENTES AL ESTADO Y A LOS PARTICULARES, CON EL OBJETO DE LOGRAR SU CORRECTA IDENTIFICACIÓN FÍSICA, JURÍDICA, FISCAL Y ECONÓMICA.	Editar Eliminar
CLAVE CATASTRAL	TÉRMINO	CÓDIGO QUE IDENTIFICA AL PREDIO EN FORMA ÚNICA, PARA SU LOCALIZACIÓN, EL CUAL SERÁ HOMOGÉNEO EN TODO EL ESTADO Y DEBERÁ CONTENER LOS DÍGITOS DE IDENTIFICACIÓN DE LOS PREDIOS, RELATIVOS A LA REGIÓN, MANZANA Y LOTE EN QUE SE ENCUENTREN, ASÍ COMO LOS DÍGITOS DE IDENTIFICACIÓN DEL MUNICIPIO Y LA POBLACIÓN AL QUE CORRESPONDAN.	Editar Eliminar

Figura 50: Pantalla - Glosario de Términos y Abreviaturas

Fuente: Elaborado por la autora

Registro de Información Relacionada

Ruta Actual: Administración del Sistema / Gestión de Proyectos de Desarrollo / Registro de Información Relacionada

Registro de Información Relacionada						+ Agregar Registro	← Regresar
Título	Tipo	Autor	Fecha Expedición	Archivo	Opciones		
CÓDIGO ORGÁNICO DE ORGANIZACIÓN TERRITORIAL, AUTONOMÍA Y DESCENTRALIZACIÓN	NORMATIVA LEGAL	MINISTERIO DE COORDINACIÓN DE LA POLÍTICA Y GOBIERNOS AUTÓNOMOS DESCENTRALIZADOS	2011-02-01	Descargar	✎ Editar		
LEY ORGÁNICA DE ORDENAMIENTO TERRITORIAL, USO Y GESTIÓN DE SUELO	NORMATIVA LEGAL	ASAMBLEA NACIONAL DEL ECUADOR	2016-07-05	Descargar	✎ Editar		
ORDENANZA QUE REGLAMENTA EL USO Y LA OCUPACIÓN DE SUELO EN EL CANTÓN IBARRA	NORMATIVA LEGAL	CONCEJO MUNICIPAL DE IBARRA	2016-08-16	Descargar	✎ Editar		
3 Registro(s)							⏪ ⏩ ⏴ ⏵

Figura 51: Pantalla - Registro de Información Relacionada
Fuente: Elaborado por la autora

Registro de Stakeholders

Administración del Sistema / Gestión de Proyectos de Desarrollo / Registro de Stakeholders

Registro de Stakeholders						+ Agregar Stakeholder	← Regresar
Apellidos	Nombres	Cargo	Dirección	Tipo	Opciones		
ALBÁN	DAYANA	DIRECTORA GESTIÓN AMBIENTAL	DIRECCIÓN DE GESTIÓN AMBIENTAL	USUARIO INTERNO	📅 Detalles		
ECHVERRÍA	SONIA	DIRECTORA DE TALENTO HUMANO	DIRECCIÓN DE TALENTO HUMANO	USUARIO INTERNO	📅 Detalles		
GARCÍA	ARTURO	DIRECTOR FINANCIERO	DIRECCIÓN FINANCIERA	CONSULTOR	📅 Detalles		
GUDIÑO	CARLOS ANDRÉS	DIRECTOR DE TECNOLOGÍAS DE LA INFORMACIÓN	DIRECCIÓN DE TECNOLOGÍAS DE LA INFORMACIÓN	USUARIO INTERNO	📅 Detalles		
LUGO	PAÚL	DIRECTOR DE AVALÚOS Y CATASTROS	DIRECCIÓN DE AVALÚOS Y CATASTROS	USUARIO INTERNO	📅 Detalles		
MORÁN	CARLOS	DIRECTOR ADMINISTRATIVO	DIRECCIÓN ADMINISTRATIVA	USUARIO INTERNO	📅 Detalles		

Figura 52: Pantalla - Registro de Stakeholders
Fuente: Elaborado por la autora

El Registro de Proyectos de Desarrollo de Software tiene las funcionalidades mostradas en la **Tabla 70**.

Tabla 70: Funcionalidades Registro de Proyectos

Fuente: Elaborado por la autora

FUNCIONALIDAD	DESCRIPCIÓN
Catálogo de Proyectos	<ul style="list-style-type: none"> • Permite visualizar el listado de proyectos existentes (Figura 53). • Permite agregar y editar un nuevo proyecto con datos de nombre y descripción.
Administrar Proyecto	<p>Una vez creado un proyecto, se crean automáticamente las siguientes secciones (Figura 54):</p> <ul style="list-style-type: none"> • Datos del proyecto: ver nombre y descripción del proyecto. • Glosario de Términos y Abreviaturas: permite seleccionar del repositorio los términos y abreviaturas que se usarán para el proyecto (Figura 55). • Registro de Stakeholders: permite seleccionar del repositorio los stakeholders que se usarán en el proyecto (Figura 56). • Registro de Sistemas Relacionados: permite seleccionar del repositorio los sistemas relacionados con el proyecto (Figura 57). • Registro de Documentación Relacionada: permite seleccionar del repositorio la documentación relacionada con el proyecto (Figura 58). • Administración del proceso de desarrollo: permite visualizar una hoja de ruta de las etapas, actividades y tareas del proceso, así como administrar los artefactos generados en cada una de ellas (Figura 54 y Figura 59).

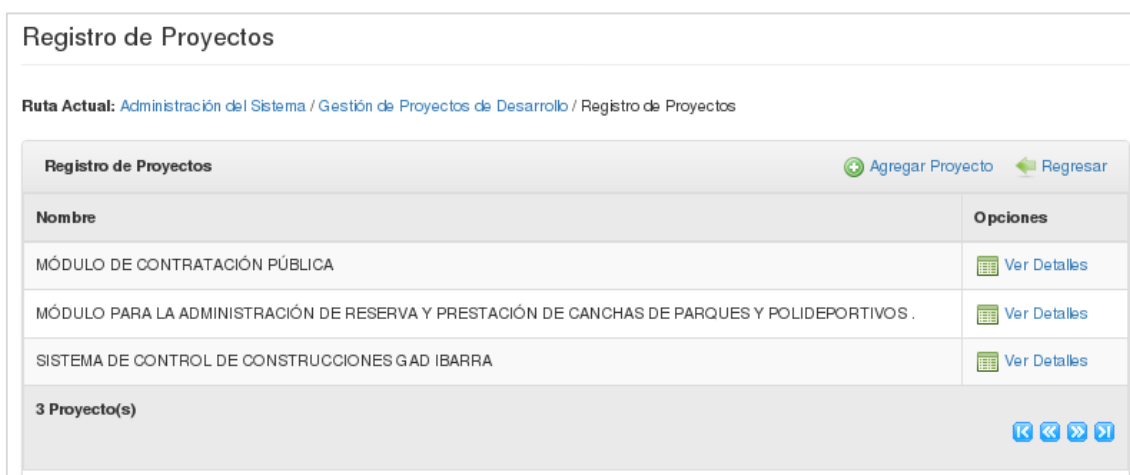


Figura 53: Pantalla - Registro de Proyectos

Fuente: Elaborado por la autora

Registro de Proyecto

Ruta Actual: [Administración del Sistema](#) / [Gestión de Proyectos de Desarrollo](#) / [Registro de Proyectos](#) / Registro de Proyecto

Datos del proyecto

[Editar Proyecto](#) [Regresar](#)

Nombre: DESARROLLO DE UN MÓDULO PARA LA ADMINISTRACIÓN DE RESERVA Y PRESTACIÓN DE CANCHAS DE PARQUES Y POLIDEPORTIVOS .

Descripción: LA ADMINISTRACIÓN DE PARQUES DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO MUNICIPAL DE SAN MIGUEL DE IBARRA GAD-I REQUIERE CONTAR CON UN MÓDULO INFORMÁTICO QUE PERMITA LA ADMINISTRACIÓN DE RESERVAS Y PRESTACIÓN DE LAS CANCHAS DE FUTBOL Y TENIS DEL PARQUE CIUDAD BLANCA Y DE LOS POLIDEPORTIVOS DEL CANTÓN.

Registros

- [Glosario de Términos y Abreviaturas](#)
- [Registro de Stakeholders](#)
- [Registro de Sistemas Relacionados](#)
- [Registro de Información Relacionada](#)
- [Registro de Requerimientos](#)

Proceso de Desarrollo

DESPEGUE DEL PROYECTO [REQUERIMIENTOS](#) [DISEÑO](#) [CONSTRUCCIÓN](#) [PRUEBAS](#) [IMPLANTACIÓN](#)

Nº	Nombre	Tipo	Artefactos
1	PLANIFICAR REUNIONES DE DESPEGUE	TAREA	CONVOCATORIA A REUNIÓN + Cargar Artefacto
2	EJECUTAR REUNIONES DE DESPEGUE	TAREA	ACTA DE REUNIÓN + Cargar Artefacto REGISTRO DE STAKEHOLDERS
3	DOCUMENTAR EL PROYECTO DE SOFTWARE	TAREA	PROYECTO DE DESARROLLO DE SOFTWARE + Cargar Artefacto
4	SOLICITAR APROBACIÓN DEL PROYECTO	TAREA	PROYECTO DE DESARROLLO DE SOFTWARE + Cargar Artefacto

Figura 54: Pantalla - Administración del Proyecto
Fuente: Elaborado por la autora

Glosario de Términos y Abreviaturas del Proyecto

Ruta Actual: [Administración del Sistema](#) / [Gestión de Proyectos de Desarrollo](#) / Glosario de Términos y Abreviaturas del Proyecto

Búsqueda de Términos y Abreviaturas

[Regresar](#)

Término o Abreviatura: [Buscar](#)

Resultado de la Búsqueda

Glosario de Términos y Abreviaturas del Proyecto			
Campo	Tipo	Descripción	Opciones
ACTIVIDAD ECONÓMICA	TÉRMINO	SE LLAMA ACTIVIDAD ECONÓMICA A CUALQUIER PROCESO DONDE SE GENERAN E INTERCAMBIAN PRODUCTOS, BIENES O SERVICIOS PARA CUBRIR LAS NECESIDADES DEL HOMBRE. LA ACTIVIDAD ECONÓMICA PERMITE LA GENERACIÓN DE RIQUEZA DENTRO DE UNA COMUNIDAD (CIUDAD, REGIÓN, PAÍS) MEDIANTE LA EXTRACCIÓN, TRANSFORMACIÓN Y DISTRIBUCIÓN DE LOS RECURSOS NATURALES O BIEN DE ALGÚN TIPO DE SERVICIO.	Quitar
CATASTRO	TÉRMINO	ES EL INVENTARIO O CENSO, DEBIDAMENTE ACTUALIZADO Y CLASIFICADO, DE LOS BIENES INMUEBLES PERTENECIENTES AL ESTADO Y A LOS PARTICULARES, CON EL OBJETO DE LOGRAR SU CORRECTA IDENTIFICACIÓN FÍSICA, JURÍDICA, FISCAL Y ECONÓMICA.	Quitar
CONTRIBUYENTE	TÉRMINO	SE DEFINE CONTRIBUYENTE TRIBUTARIO COMO AQUELLA PERSONA FÍSICA CON DERECHOS Y OBLIGACIONES, FRENTE A UN ENTE PÚBLICO, DERIVADOS DE LOS TRIBUTOS. ES QUIEN ESTÁ OBLIGADO A SOPORTAR PATRIMONIALMENTE EL PAGO DE LOS TRIBUTOS (IMPUESTOS, TASAS O CONTRIBUCIONES ESPECIALES), CON EL FIN DE FINANCIAR AL ESTADO	Quitar

Figura 55: Pantalla - Glosario de Términos y Abreviaturas del Proyecto
Fuente: Elaborado por la autora

Registro de Stakeholders del Proyecto

Ruta Actual: [Administración del Sistema](#) / [Gestión de Proyectos de Desarrollo](#) / Registro de Stakeholders del Proyecto

Búsqueda de Stakeholders [Regresar](#)

Apellidos:

Resultado de la Búsqueda

Registro de Stakeholders del Proyecto							
Apellidos	Nombres	Cargo	Dirección	Tipo	Teléfono	Correo	Opciones
ALBÁN	DAYANA	DIRECTORA GESTIÓN AMBIENTAL	DIRECCIÓN DE GESTIÓN AMBIENTAL	USUARIO INTERNO	0963525221	dalban@ibarra.gob.ec	Quitar
EHEVERRÍA	SONIA	DIRECTORA DE TALENTO HUMANO	DIRECCIÓN DE TALENTO HUMANO	USUARIO INTERNO	0963325225	secheverria@ibarra.gob.ec	Quitar
GARCÍA	ARTURO	DIRECTOR FINANCIERO	DIRECCIÓN FINANCIERA	CONSULTOR	0989652236	agarcia@ibarra.gob.ec	Quitar

Figura 56: Pantalla - Registro de Stakeholders del Proyecto
Fuente: Elaborado por la autora

Sistemas Relacionados con el Proyecto

Ruta Actual: [Administración del Sistema](#) / [Gestión de Proyectos de Desarrollo](#) / Sistemas Relacionados con el Proyecto

Búsqueda de Sistemas [Regresar](#)

Nombre de Sistema:

Resultado de la Búsqueda

Sistemas Relacionados con el Proyecto		
Nombre Corto	Nombre	Opciones
SISMERT	SISTEMA DE ESTACIONAMIENTO ROTATIVO TARIFADO	Quitar
SISINQ	SISTEMA DE INQUILINATO	Quitar

Figura 57: Pantalla - Registro de Sistemas Relacionados con el Proyecto
Fuente: Elaborado por la autora

Registro de documentación relacionada del Proyecto

Ruta Actual: Administración del Sistema / Gestión de Proyectos de Desarrollo / Registro de documentación relacionada del Proyecto

Búsqueda de Sistemas [Regresar](#)

Título: [Buscar](#)

Resultado de la Búsqueda

Registro de documentación relacionada del Proyecto

Título	Tipo	Autor	Archivo	Opciones
CÓDIGO ORGÁNICO DE ORGANIZACIÓN TERRITORIAL, AUTONOMÍA Y DESCENTRALIZACIÓN	NORMATIVA LEGAL	MINISTERIO DE COORDINACIÓN DE LA POLÍTICA Y GOBIERNOS AUTÓNOMOS DESCENTRALIZADOS	Descargar	Quitar
LEY ORGÁNICA DE ORDENAMIENTO TERRITORIAL, USO Y GESTIÓN DE SUELO	NORMATIVA LEGAL	ASAMBLEA NACIONAL DEL ECUADOR	Descargar	Quitar

Figura 58: Pantalla - Registro de Documentación Relacionada del Proyecto
Fuente: Elaborado por la autora

Detalle del Artefacto

Nombre: CONVOCATORIA A REUNIÓN **Plantilla:**

Descripción: LA CONVOCATORIA SE DEBE REALIZAR FORMALMENTE A TRAVÉS DEL SISTEMA DOCUMENTAL QUIPUX, MEDIANTE UN MEMORANDO (INTERNO) U OFICIO (EXTERNO).

Artefactos cargados

N°	Archivo	Fecha Subida	Opciones
1	DTI-2017-00642-M_(1)(21-08-2017_11_00_36).pdf	2017-08-21	Ver archivo Eliminar
2	DTI-2017-00023-O_(1)(21-08-2017_11_03_18).pdf	2017-08-21	Ver archivo Eliminar
3	DTI-2017-00023-O_(1)(21-08-2017_11_13_53).pdf	2017-08-21	Ver archivo Eliminar

Visualizar Archivo

1 of 1 Automatic Zoom

Figura 59: Pantalla - Artefactos del Proyecto
Fuente: Elaborado por la autora

Matriz de Trazabilidad

Ruta Actual: Administración del Sistema / Gestión de Proyectos de Desarrollo / Matriz de Trazabilidad

Matriz de Trazabilidad Regresar		
Código	Requerimiento	Casos de Prueba
RF-001	REGISTRO DE CANCHAS	
RF-002	EDICIÓN DE REGISTROS DE CANCHAS	CP-001 CP-002
RF-003	CONSULTA DE REGISTRO DE CANCHAS	
RF-004	REGISTRO DE MANTENIMIENTO DE CANCHAS	
RF-005	EDICIÓN DE REGISTROS DE MANTENIMIENTO	

5 Requerimiento(s)

Figura 60: Pantalla - Matriz de Trazabilidad del proyecto
Fuente: Elaborado por la autora

5. CAPÍTULO V. RESULTADOS Y DISCUSIÓN

5.1. Introducción

El uso de una guía metodológica adaptada a las necesidades de la Unidad de Desarrollo de Software representa una ventaja competitiva en relación al uso de estándares internacionales ya que la guía soporta la cultura organizacional del GAD-I, las preferencias de los stakeholders, la madurez de la institución, el tipo y complejidad de proyectos realizados, así como la disponibilidad de recursos.

En este capítulo se presentan los resultados obtenidos al implantar la guía metodológica de gestión de requerimientos de software en el Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra. En vista de que la gestión de requerimientos es el proceso de elicitar, recolectar, analizar, documentar, validar, controlar, rastrear y dar seguimiento a requerimientos; la medición del proceso se orientó a tomar datos de esas actividades específicas.

5.2. Proceso de gestión de requerimientos de software

En la **Figura 61** muestra una vista general del proceso de gestión de requerimientos de software definido para la Dirección de TIC, con una referencia a sus entradas, salidas, recursos y responsables. De la misma manera, en la **Anexo 8** se puede observar el detalle de las actividades, tareas y artefactos del proceso, como parte de la guía metodológica desarrollada.

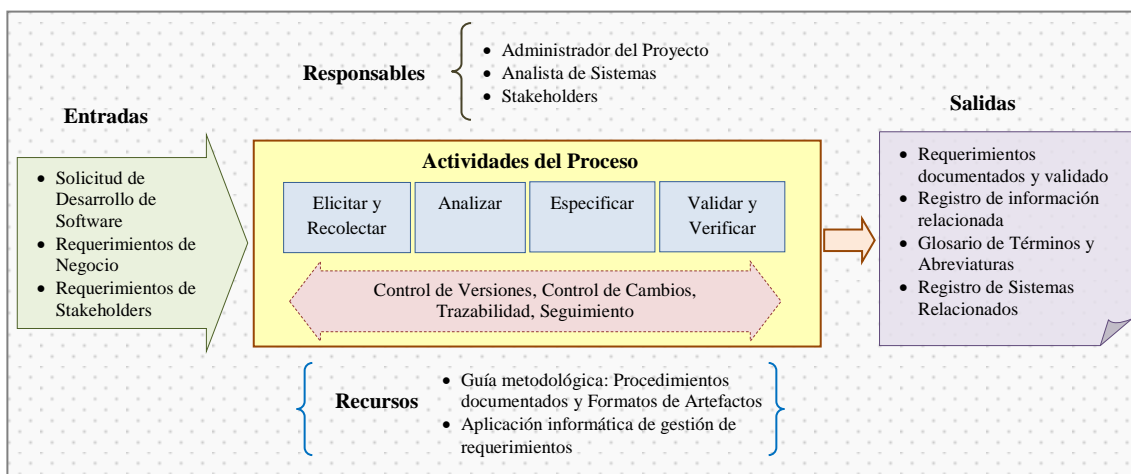


Figura 61: Vista general del proceso de gestión de requerimientos de software

Fuente: Elaborado por la autora

5.3. Características funcionales seleccionadas para la aplicación informática

En la **Figura 62** y **Figura 63** se detallan las características funcionales analizadas de herramientas de gestión de requerimientos existentes en el mercado, y que fueron seleccionadas para la construcción de la aplicación informática de la Dirección de TIC, mismas que están divididas en las actividades de la Ingeniería de Requerimientos: Desarrollo y Administración.

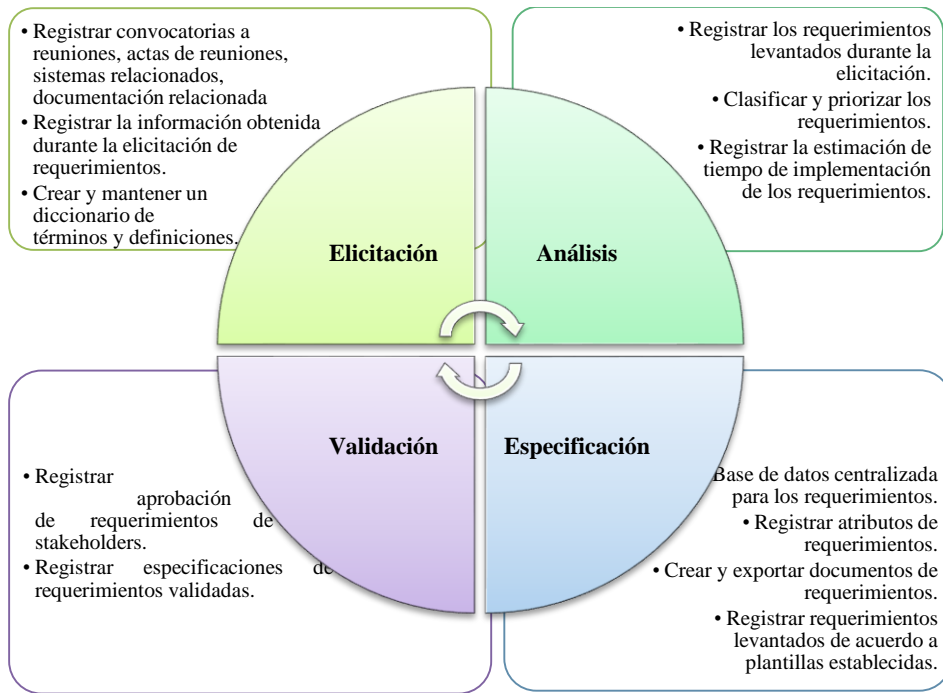


Figura 62: Características aplicación informática - Desarrollo de Requerimientos
Fuente: Elaborado por la autora

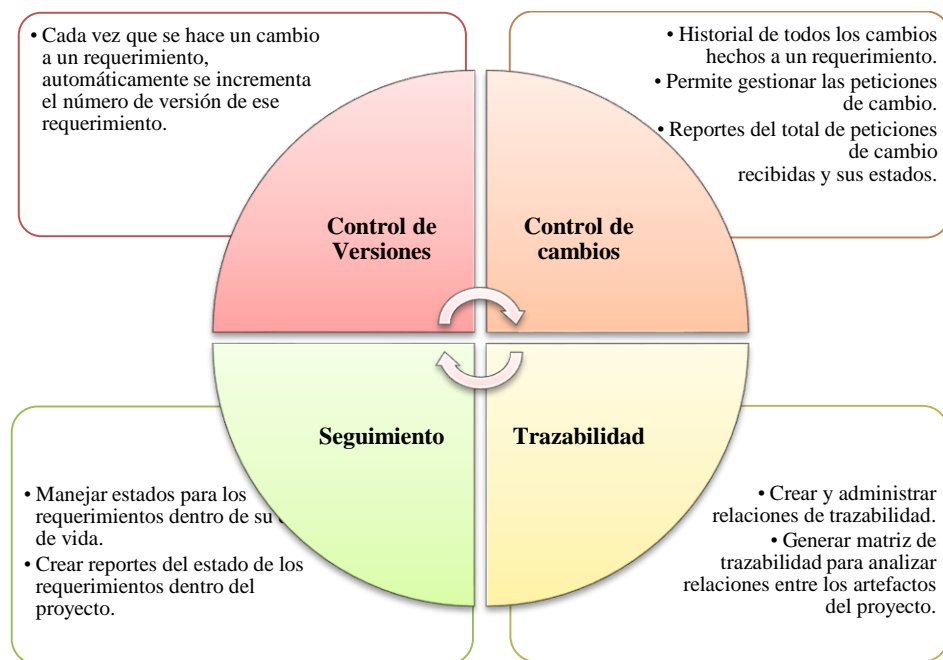


Figura 63: Características aplicación informática - Administración de Requerimientos
Fuente: Elaborado por la autora

5.4. Aplicación informática para la gestión de requerimientos de software

En la **Figura 64** se puede visualizar el diagrama de arquitectura de la aplicación informática desarrollada, que responde a los requerimientos de la Dirección de TIC en cuanto a modelo de capas, despliegue y herramientas tecnológicas. De la misma manera, en la capa de “Controlador” se muestran los módulos de la aplicación informática desarrollada, entre los que figuran: módulo de Gestión de Proyectos de Desarrollo de Software, Mantenimiento de Sistemas, Inventario de Sistemas y Configuración del sistema.

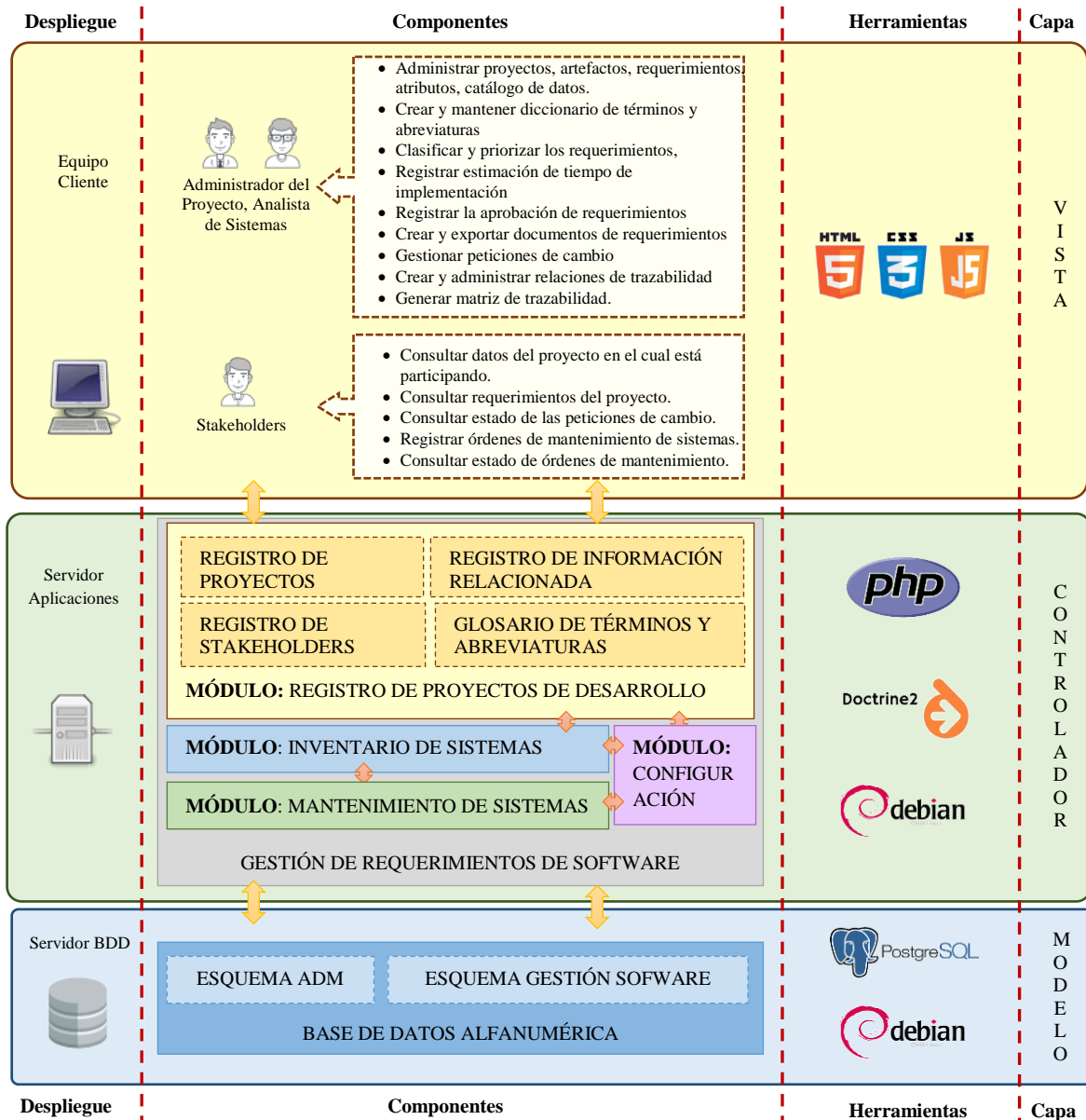


Figura 64: Arquitectura de la aplicación informática

Fuente: Elaborado por la autora

5.5. Guía Metodológica para la gestión de requerimientos de software

El **Anexo 8** contiene la guía metodológica desarrollada para el presente estudio, la misma que contiene la descripción, alcance, políticas, así como el proceso para la gestión de requerimientos de software con procedimientos para realizar el Despegue del Proyecto, Elicitar y Recolectar, Analizar, Especificar, Validar y Verificar, Controlar Versiones, Trazabilidad, Controlar Cambios y Dar seguimiento a requerimientos de software.

Para determinar el nivel de mejora en la gestión de requerimientos de la Dirección de TIC, se realizó una medición del proceso mediante la selección de medidas, levantamiento de información, cálculo de valores y análisis de resultados, tal como se detalla a continuación.

5.5.1. Planeación de la medición del proceso

Para la implantación del proyecto, se capacitó durante 15 horas a los Analistas de Sistemas en el contenido de la guía metodológica y en el uso de la aplicación informática. No se requirió más horas de capacitación en vista de que la guía fue desarrollada en reuniones previas con los Analistas y por esta razón ya estaban familiarizados con la misma.

Para la organización del trabajo se dividió a los Analistas de Sistemas en tres grupos, asignando tareas a cada uno, tal como se muestra en la **Tabla 71**. Al grupo G01 se le entregó la guía metodológica para su uso y cumplimiento obligatorio, por el contrario al grupo G02 se le pidió que realice el trabajo como siempre lo ha hecho en la Unidad.

A cada grupo de trabajo se le asignó dos usuarios: uno con capacidad de toma de decisiones y otro con amplios conocimientos sobre los procesos de la Unidad requirente. El Analista A07 participó como observador en la ejecución de las tareas de los dos grupos, con el fin de realizar la medición del proceso y análisis de resultados.

El tiempo de utilización de la guía para la medición del proceso, fue del 23 de octubre al 10 de noviembre del 2017, tiempo en el que se trabajó con los Analistas de Sistemas en la atención de la solicitud para el desarrollo del módulo “Declaración de Patente y Activos Totales en Línea” de la Dirección de Gestión Tributaria, específicamente en la etapa de levantamiento de requerimientos.

Tabla 71: Organización del trabajo para medición del proceso

Fuente: Elaborado por la autora

Grupo	Analista de Sistemas	Stakeholders	Tarea	Condiciones
G01	A01, A02, A03	U01, U02	Levantamiento de requerimientos del sub módulo: Declaración de Patente y Activos Totales del Cantón Ibarra (Original y Sustitutiva)	Con el uso de la guía
G02	A04, A05, A06	U03, U04	Levantamiento de requerimientos del sub módulo: Declaración de Patente y Activos Totales de Otros Cantones (Original y Sustitutiva)	Sin el uso de la guía
G03	A07	--	Medición del proceso	--

- **Identificar los problemas del proceso**

Los problemas del proceso de gestión de requerimientos de la Unidad de Desarrollo de Software se detallaron en el Capítulo I.

- **Seleccionar y Definir medidas**

Algunos de los problemas detectados en la Unidad de Desarrollo de Software fueron los relacionados con la asignación de tiempo insuficiente durante la etapa de levantamiento y análisis de requerimientos, escasa negociación de compromisos al inicio del proyecto así como el inadecuado involucramiento del usuario.

Otros problemas fueron los fallos en la documentación de requerimientos por parte de los Analistas de Sistemas, ya que debido a la falta de definición de procedimientos y artefactos para esta etapa, se tenía como consecuencia especificaciones de requerimientos incorrectas, incompletas, no validadas por el usuario, e incluso inexistentes.

En ese sentido, se planteó determinar la mejora del proceso en base al incremento del tiempo designado en la etapa de requerimientos así como en la cantidad de entregables finalizados. En la **Tabla 72** se muestran las entidades y atributos para la medición del proceso de gestión de requerimientos de software del presente estudio, mismos que fueron seleccionados de la referencia presentada en la guía PSM Practical Software Measurement - Medición práctica de software.

Tabla 72: Entidades y atributos seleccionados para la medición del proceso

Fuente: Elaborado por la autora

Entidad	Atributo	Unidad de medida
Esfuerzo	# de horas de reuniones de elicitación # de horas de reuniones de análisis # de horas de trabajo de especificación # de horas de reuniones de validación	Horas
Unidad de Trabajo	# de requerimientos analizados # de requerimientos documentados # de requerimientos validados	Unidades
Tiempo	Duración de la actividad de elicitación Duración de la actividad de análisis Duración de la actividad de especificación Duración de la actividad de validación	Días Laborables
Personas	Nivel de satisfacción del servicio (Usuario)	Porcentaje

Para el cálculo del porcentaje de incremento o decremento entre dos valores medidos y para el cálculo del porcentaje promedio, se aplicó la **Fórmula 1** y **Fórmula 2** respectivamente.

Fórmula 1: Cálculo de porcentaje de diferencia entre dos valores

$PIN = ((R_LB - R_IP) / R_IP) * 100$	<ul style="list-style-type: none"> • PIN: porcentaje de incremento [%] • R_LB: respuestas “SI” de la línea base 2016 [# de respuestas] • R_IP: respuestas “SI” luego de la implementación de la propuesta 2017 [# de respuestas]
---	--

Fórmula 2: Cálculo del porcentaje promedio de incremento

$$PPI = PIN / N$$

- **PPI:** porcentaje promedio de incremento [%]
- **PIN:** porcentaje de incremento [%]
- **N:** cantidad de preguntas de la encuesta

5.5.2. Medición del proceso

- **Entidad Esfuerzo**

Para realizar la medición de los atributos de la entidad “Esfuerzo” se elaboró un formato para el registro de las horas de las reuniones de elicitación, análisis, especificación y validación de requerimientos de cada grupo de trabajo. En el **Anexo 6** se muestran los datos levantados de la medición del proceso realizado por grupo G02 y G01.

Posterior a eso, se determinó el incremento de los valores de los atributos de la entidad “Esfuerzo” con el uso y sin el uso de la guía metodológica, tal como se muestra en la **Tabla 73**. Los resultados mostraron un incremento de en el número de horas invertidas para las reuniones de elicitación, análisis, especificación y validación de requerimientos de software, tal como se puede ver en la

Figura 65.

Tabla 73: Incremento de valores de atributos de la entidad Esfuerzo

Fuente: Elaborado por la autora

Atributos	G02 - Sin el uso de la guía [horas]	G01 - Con el uso de la guía [horas]	Incremento [horas]
# horas reuniones de elicitación	09h. 20m.	16h. 15m.	06 h. 55m.
# horas reuniones análisis	03h. 20m.	07h. 25m.	04h. 05m.
# horas trabajo especificación	0h. 0m.	08h. 0m.	08h. 0m.
# horas reuniones de validación	0h. 0m.	04h. 05m.	04h. 05m.
Total	12h. 40m. (12,66 h)	35h. 45m. (35,75 h)	23h. 05m. (23,083 h)

Cálculo de porcentaje de incremento:

$$PIN \text{ Esfuerzo} = ((35,75-12,66)/12,66)*100$$

$$PIN \text{ Esfuerzo} = 182,38\%$$

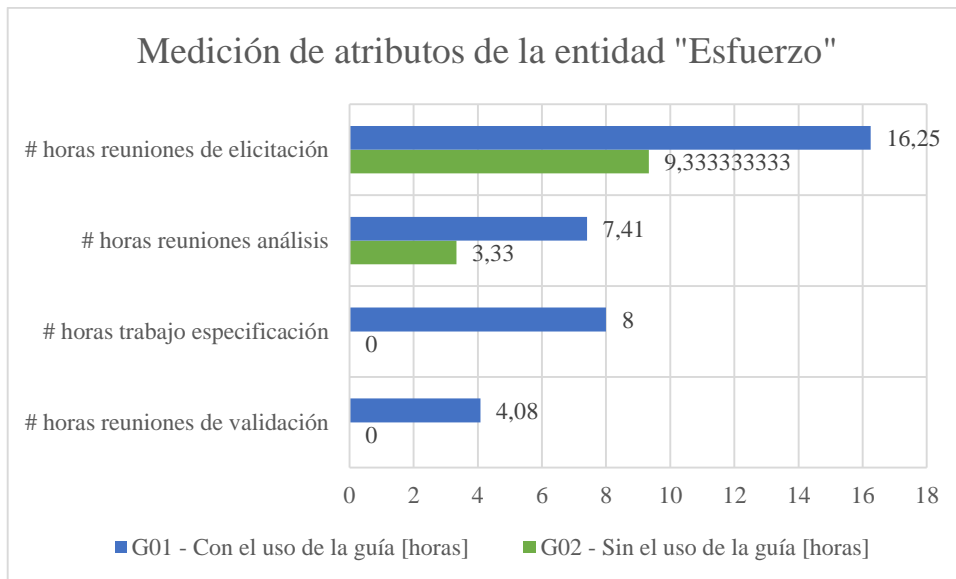


Figura 65: Medición de atributos de la entidad "Esfuerzo" para G01 y G02
Fuente: Elaborado por la autora

- **Entidad Unidad de Trabajo**

Para realizar la medición de los atributos de la entidad “Unidad de Trabajo” se elaboró un formato para el registro del número de requerimientos analizados, documentados y validados de cada grupo. En el **Anexo 7** se muestran los datos levantados de la medición de proceso realizado por grupo G01 y G02.

Posterior a eso, se determinó el incremento de los valores de atributos de la entidad “Unidad de Trabajo” con el uso y sin el uso de la guía metodológica, tal como se muestra en la **Tabla 74**. Los resultados mostraron un incremento en la cantidad de entregables generados en cada actividad por parte de los Analistas de Sistemas, específicamente en el número de requerimientos analizados, documentados y validados, tal como se puede ver en la **Figura 66**.

Tabla 74: Incremento de valores de atributos de la entidad Unidad de Trabajo
Fuente: Elaborado por la autora

Atributos	G02 - Sin el uso de la guía [unidades]	G01 - Con el uso de la guía [unidades]	Incremento [unidades]
# requerimientos analizados	4	10	6
# requerimientos documentados	0	9	9
# requerimientos validados	0	9	9
Total	4	28	24

Cálculo de porcentaje de incremento:

$$\text{PIN Unidad de Trabajo} = ((28 - 4)/4) * 100$$

$$\text{PIN Unidad de Trabajo} = 600\%$$

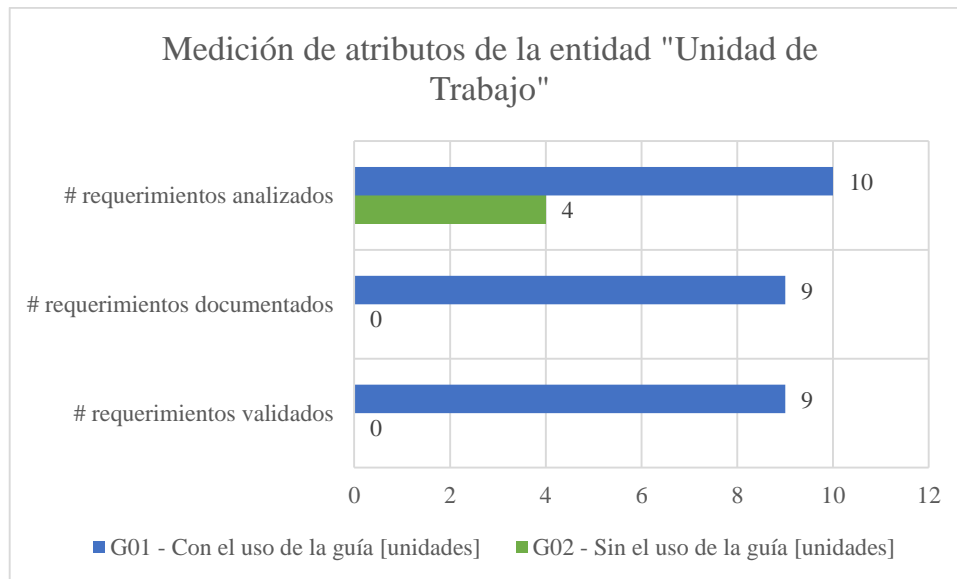


Figura 66: Medición de atributos de la entidad "Unidad de Trabajo" para G01 y G02
Fuente: Elaborado por la autora

- **Entidad Tiempo**

En el caso de la entidad “Tiempo” se levantaron datos de la fecha de inicio y finalización de las actividades de elicitación, análisis, especificación y validación de cada grupo de trabajo. Para la interpretación de resultados, se considerará como una mejora en la gestión del proceso al cumplimiento de las actividades dentro del plazo establecido para la etapa de levantamiento de requerimientos.

Se usaron los mismos datos levantados en la entidad “Esfuerzo”, con la diferencia que en este caso se usaron las fechas de inicio y finalización de cada tarea, tal como se muestra en la **Tabla 75** y **Tabla 76**.

Tabla 75: Datos de medición de la entidad Tiempo para el grupo G02
Fuente: Elaborado por la autora

Atributos	G02 - Sin el uso de la guía		G02 - Duración [días laborables]
	Fecha Inicio	Fecha Fin	
Duración de la actividad de elicitación	23/10/2017	07/11/2017	12
Duración de la actividad de análisis	08/11/2017	10/11/2017	3
Duración de la actividad de especificación	--	--	--
Duración de la actividad de validación	--	--	--

Tabla 76: Datos de medición de la entidad Tiempo para el grupo G01

Fuente: Elaborado por la autora

Atributos	G01 - Con el uso de la guía		G01 - Duración [días laborables]
	Fecha Inicio	Fecha Fin	
Duración de la actividad de elicitación	23/10/2017	30/10/2017	6
Duración de la actividad de análisis	31/10/2017	01/11/2017	2
Duración de la actividad de especificación	06/11/2017	08/11/2017	3
Duración de la actividad de validación	08/11/2017	09/11/2017	2

Posterior a eso, se determinó el porcentaje de incremento de los valores de atributos de la entidad “Tiempo” con el uso y sin el uso de la guía metodológica, tal como se muestra en la **Tabla 77**. El porcentaje se calculó usando únicamente valores de duración de la actividad de elicitación y análisis, ya que en el caso del G02 no lograron realizar las actividades de especificación y validación en el plazo establecido.

Tabla 77: Incremento de valores de atributos de la entidad "Tiempo"

Fuente: Elaborado por la autora

Atributos	G02 - Sin el uso de la guía [días laborables]	G01 - Con el uso de la guía [días laborables]	Decremento [días laborables]
Duración de la actividad de elicitación	12	6	6
Duración de la actividad de análisis	3	2	1
Duración de la actividad de especificación	0	3	N/A
Duración de la actividad de validación	0	2	N/A
Total	15	8	7

Cálculo de porcentaje de incremento:

$$\text{PIN Tiempo} = ((8 - 15)/15) * 100$$

$$\text{PIN Tiempo} = - 46,66\% \text{ (Al ser negativo, es un porcentaje de decremento)}$$

Los resultados mostraron que con el uso de la guía metodológica se lograron cumplir las actividades de elicitación, análisis, especificación y validación de requerimientos en el período de tiempo establecido, y que además se lo redujo en un 46,66%.

Por otro lado, los Analistas de Sistemas que no usaron la guía metodológica no realizaron actividades de especificación y validación, por lo que no lograron cumplir con el plazo establecido y tampoco cumplieron con la generación de artefactos requeridos para las siguientes actividades del proceso de desarrollo de software, tal como se puede ver en la **Figura 67**.

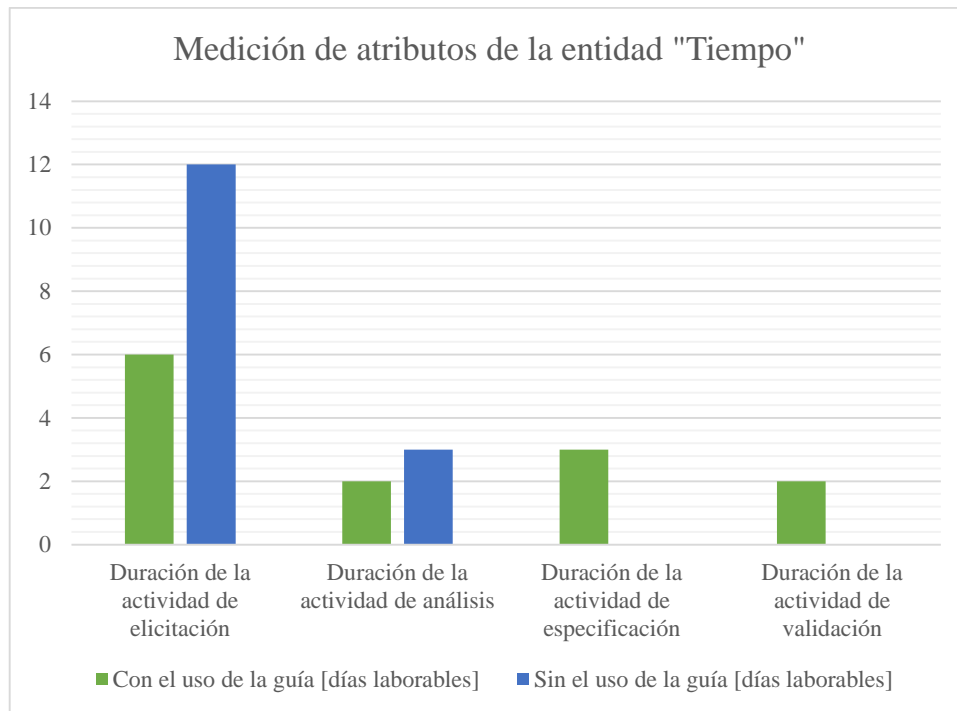


Figura 67: Medición de atributos de la entidad "Tiempo" para G01 y G02
Fuente: Elaborado por la autora

- **Entidad Personas - Usuarios**

Para realizar la medición de los atributos de la entidad “Personas - Usuarios” se elaboró una encuesta para conocer el nivel de satisfacción de los usuarios con respecto al servicio prestado por la Unidad de Desarrollo de Software, cuyos resultados se pueden observar en la **Tabla 78**.

En la encuesta aplicada a 100 usuarios de sistemas informáticos que solicitaron servicios de desarrollo y mantenimiento de sistemas informáticos entre los años 2016 y 2017, se evaluaron varios puntos antes y después de la implantación de la propuesta, tales como: asignación de personal para la atención de pedidos, ejecución de reuniones de trabajo para análisis, período para realizar pruebas, cumplimiento de requerimientos funcionales, satisfacción de servicios.

Una vez obtenidos los resultados de la encuesta aplicada en el 2017, se compararon con los resultados de la misma encuesta aplicada en el 2016 y considerada como línea base. Para la comparación, se obtuvo el valor de porcentaje de incremento de las respuestas “SI” mediante la **Fórmula 1**, y posterior a eso se determinó el porcentaje promedio de incremento mediante la **Fórmula 2**.

En la **Figura 68** se puede observar la diferencia en las respuestas con valor “SI” de los usuarios para cada una de las preguntas planteadas, en el año 2016 y 2017, antes y después de la implantación de la propuesta. Mediante los cálculos realizados, se pudo determinar un incremento del nivel de satisfacción de los usuarios del 20,28%.

Tabla 78: Resultados encuesta usuarios GAD-I

Fuente: Elaborado por la autora

Pregunta	Línea Base (2016)		Resultados (2017)		Diferencia (SI)	Porcentaje incremento (SI)
	SI	NO	SI	NO		
1. En proyectos de desarrollo o mantenimiento de software, ¿Se asignó un profesional de la Unidad para la atención de sus requerimientos?	78	22	96	4	18	23%
2. ¿Fue invitado a reuniones de trabajo para analizar sus requerimientos?	72	28	86	14	14	19%
3. ¿Se le entregó algún documento con la especificación escrita de sus requerimientos solicitando su aprobación?	45	55	61	39	16	36%
4. ¿Tuvo un período en el cual Usted pudo realizar pruebas de sus requerimientos?	69	31	76	24	7	10%
5. ¿El desarrollo o mantenimiento de software recibido, cumplió con sus requerimientos funcionales?	72	28	86	14	14	19%
6. ¿La atención de sus requerimientos se realizó en los tiempos previstos?	58	42	67	33	9	16%
7. ¿Considera que el servicio entregado por la Unidad de Desarrollo de Software es adecuado?	73	27	87	13	14	19%
Porcentaje Promedio del Incremento						20,28%

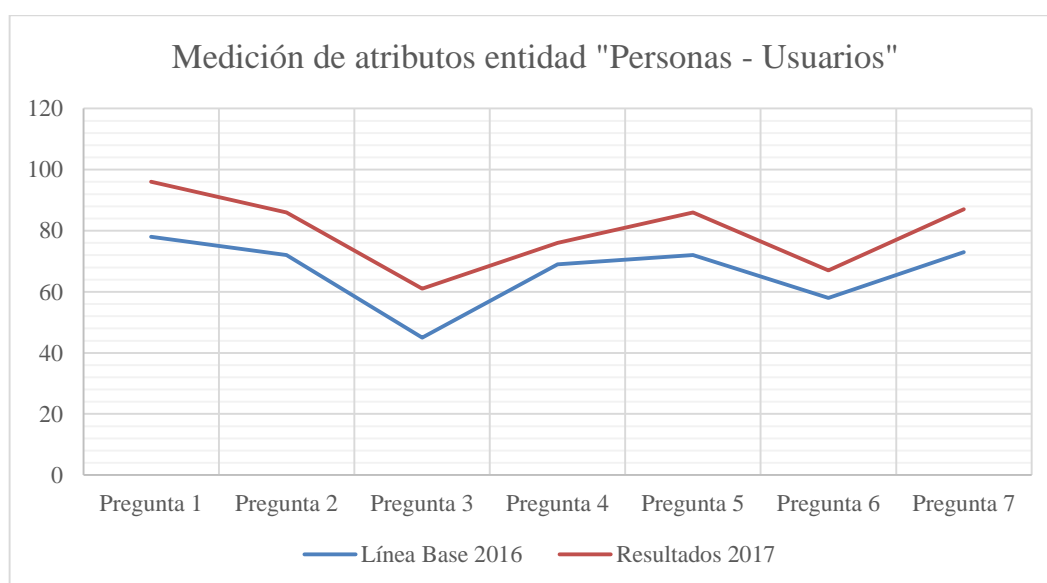


Figura 68: Medición de atributos de la entidad "Personas - Usuarios" en el año 2016 y 2017

Fuente: Elaborado por la autora

5.5.3. Análisis de los Resultados

Una vez que se han medido y analizado los atributos de las entidades seleccionadas, se concluye que la implantación de la guía metodológica incidió positivamente en la gestión de requerimientos de software de la Dirección de TIC; ya que permitió mejorar los procesos para elicitar, analizar, documentar y validar los requerimientos durante el proceso de desarrollo de software, tal como se muestra en la **Figura 69** y se analiza en la **Tabla 79**.

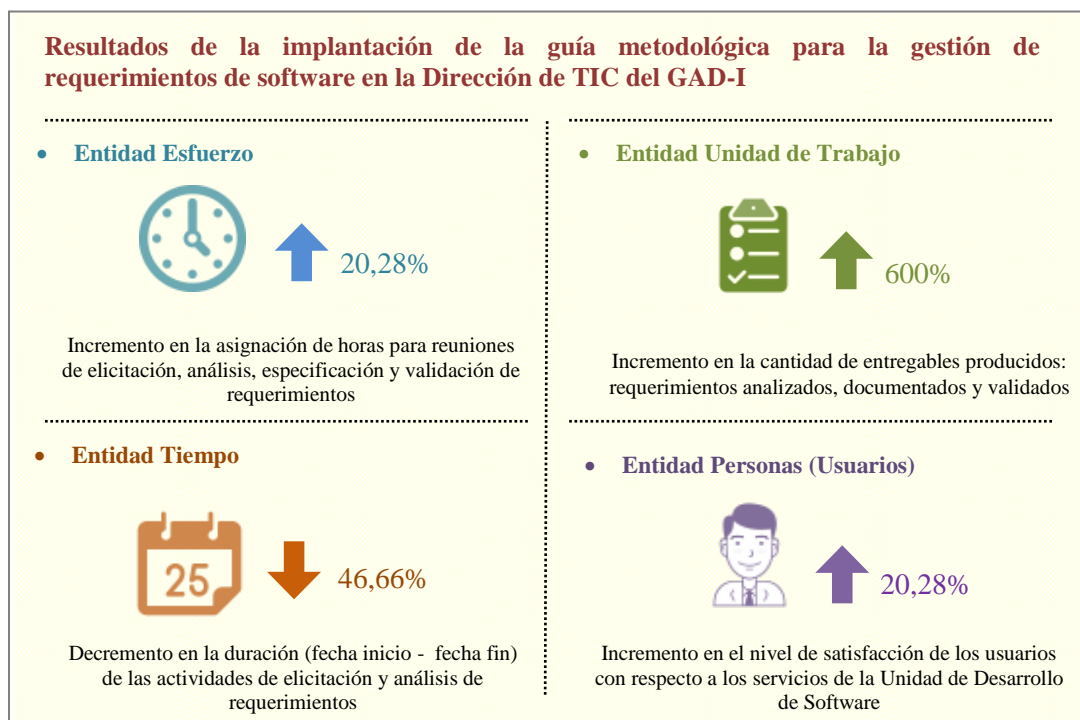


Figura 69: Resultados de la implantación de la guía metodológica

Fuente: Elaborado por la autora

Tabla 79: Análisis de Resultados

Fuente: Elaborado por la autora

Entidad	Porcentaje de incremento/decremento	Análisis y Discusión
Entidad Esfuerzo	Se encontró un incremento total del 182,38% en la asignación de horas para reuniones de trabajo de elicitación, análisis, especificación y validación de requerimientos.	Esto implica que los Analistas de Sistemas están en la capacidad de analizar con mayor detenimiento el problema, comprender el alcance, evitar requerimientos duplicados o mal identificados, y en consecuencia construir software acorde a las necesidades del usuario. Además, ya que en las actividades de elicitación, análisis y validación se requiere la presencia de los stakeholders, se mejora el nivel de involucramiento y participación del mismo, evitando así tener cambios inesperados cuando se haya iniciado el desarrollo del sistema.

Continuación de la **Tabla 79**.

Entidad	Porcentaje de incremento/decremento	Análisis y Discusión
Entidad Unidad de Trabajo	Se encontró un incremento total del 600% en cuanto a la cantidad de entregables generados durante esta etapa (requerimientos analizados, documentados y validados).	<p>A futuro, esto evitará tener documentación dispersa, no estandarizada, inexistente, así como dificultad para demostrar que el software cumple con los requerimientos de usuario.</p> <p>El tener requerimientos analizados, documentados y validados permitirá al Administrador conocer el estado y avance del proyecto, y a los Analistas permitirá realizar casos de prueba acorde a lo especificado para el sistema.</p>
Entidad Tiempo	Se encontró un decremento del 46,66% en cuanto al tiempo de inicio y fin de una tarea de elicitación y análisis de requerimientos.	<p>Con el uso de la guía metodológica se lograron cumplir las actividades de elicitación, análisis, especificación y validación de requerimientos en el período de tiempo establecido (días laborables), y además se lo redujo en un 46,66% en relación a no utilizar la guía.</p> <p>Por otro lado, los Analistas de Sistemas que no usaron la guía metodológica no realizaron actividades de especificación y validación, por lo que no lograron cumplir con el plazo establecido y tampoco cumplieron con la generación de artefactos requeridos para las siguientes actividades del proceso de desarrollo de software</p>
Entidad Personas (Usuarios)	Se encontró un incremento total del 20,28% en cuanto al nivel de satisfacción de los usuarios sobre las actividades realizadas por la Unidad de Desarrollo de Software para la atención de sus requerimientos.	El tiempo de utilización y aplicación de la guía metodológica fue de septiembre a noviembre del 2017; período que fue relativamente pequeño ya que los procesos de desarrollo de software mediano y grande en el lugar de estudio eran de cinco meses en adelante. Por esta razón es posible que las mejoras en las percepciones de los usuarios se vean en un mayor plazo.

6. CAPÍTULO VI. CONCLUSIONES Y RECOMENDACIONES

6.1. CONCLUSIONES

- Uno de los problemas que se encontraron al implantar un proceso de gestión de requerimientos de software en esta institución, fue que los Analistas de Sistemas ya estaban acostumbrados a trabajar de una manera informal en el desarrollo de software; y les resultaba dificultoso el uso de un procedimiento documentado y plantillas de artefactos para realizar su trabajo debido a que consideraban que se invertía mucho tiempo en la ejecución de tareas en las cuales el esfuerzo realizado no era reconocido.
- Los resultados positivos de la implantación de un proceso de gestión de requerimientos se obtendrán a largo plazo debido a la naturaleza del ciclo de vida de desarrollo, en el cual un producto software se puede finalizar en períodos de meses o años. Sin embargo, el hecho de establecer actividades y artefactos ha permitido tener una mejora notable en la gestión de requerimientos de la Unidad, obteniendo beneficios como: centralización de información, estandarización de plantillas, mejoras para la identificación y clasificación de requerimientos, disponibilidad de información, historial del trabajo realizado, control de avances, entre otros.
- La generación de documentación interna y externa, así como la creación de artefactos dentro de la gestión de requerimientos de software, se puede malentender por los usuarios como burocracia en el proceso; sin embargo se ha constatado la importancia de conservar estos registros para mantener un nivel de formalidad, registrar evidencias, así como para involucrar y responsabilizar a todos los stakeholders dentro del proyecto.
- La existencia de gran variedad de herramientas para la administración del ciclo de vida del software y el desconocimiento previo de las mismas, dificultó el proceso de revisión y obtención de características a implementar en la aplicación informática de la Unidad de Desarrollo de Software. Sin embargo, el establecimiento de criterios de agrupamiento de las características funcionales permitió comprender las herramientas desde el punto de vista de la Ingeniería de Requerimientos, así como determinar cuáles herramientas se debían analizar y cuáles no, en base al alcance del presente estudio.
- Implementar una aplicación de gestión de requerimientos de software con características similares a las que tienen las herramientas existentes en el mercado representa una tarea de mejoramiento continuo; esto, debido a la complejidad de implementación de funcionalidades de control de cambios, control de versiones y trazabilidad. En una primera etapa no se podrá alcanzar el mismo nivel que una herramienta madura, sin embargo se ha constatado que la aplicación desarrollada puede cubrir las necesidades de sistematización de la gestión de requerimientos de la Unidad, considerando que anteriormente no se contaba con una aplicación de este tipo.
- Uno de los aspectos más complicados dentro del estudio fue el de definir una metodología con la que se trabajaría para la definición del proceso de gestión de requerimientos de software, ya que en primer lugar era necesario establecer bases conceptuales de Ingeniería de Requerimientos entre los Analistas de Sistemas y en segundo lugar se tuvo que establecer actividades, tareas y salidas a realizar para obtener el proceso requerido.

6.2. RECOMENDACIONES

- Cuando se inicie una implantación de un proceso de gestión de requerimientos en una Unidad de Desarrollo de Software, primero es importante llegar a consensos con los funcionarios con respecto a los términos que se manejarán, cantidad de documentación a generar, número de actividades a realizar, productos que se entregarán, roles y responsabilidades dentro del proyecto; todo esto se puede lograr con la ejecución de reuniones de trabajo en las que se exponen las propuestas y mediante debates se llegan a acuerdos.
- Uno de los principales contribuyentes para el éxito de un proyecto de desarrollo de software es la aplicación adecuada de la Ingeniería de Requerimientos; por ello es recomendable iniciar la implementación de un proceso básico e irlo mejorando a medida que los Analistas de Sistemas vayan ganando experiencia en la aplicación de técnicas, generación de artefactos, documentación de requerimientos, registro de información, organización del trabajo, entre otros.
- Para mejorar el involucramiento del usuario y lograr la aceptación del proceso de gestión de requerimientos de software, es importante concienciar a todas las Unidades Administrativas sobre el trabajo que se realiza en la Unidad de Desarrollo de Software a través de diferentes medios como: aprobación mediante resolución de los procesos y procedimientos, charlas informativas del proceso de desarrollo antes de iniciar un proyecto, mejoras en la comunicación institucional durante la ejecución del proyecto.
- En este estudio se seleccionaron seis herramientas para la revisión de funcionalidades; sin embargo, para trabajos similares se recomienda que se seleccione un máximo de tres herramientas del mismo tipo con el objetivo de que el investigador se enfoque en el alcance y tamaño de la investigación, para ello puede guiarse en la siguiente clasificación de herramientas: Administración de requerimientos, Administración de tareas, Control de defectos, Administración de Casos de Prueba, Administración de Versiones, Modelamiento, Monitoreo y Reportes.
- Si se desea iniciar el desarrollo de una aplicación informática de gestión de requerimientos de software en una institución pública se recomienda definir un proceso de control de cambios básico, no someter todos los artefactos del proceso de desarrollo a control de versiones y no implementar todos los tipos de trazabilidad entre artefactos. Una vez que los usuarios se familiaricen con estos nuevos términos de administración, se pueden ir implementando más funcionalidades y mejoras en la aplicación.
- La realización del presente estudio permitió establecer una metodología de trabajo para la definición de un proceso de gestión de requerimientos de software, misma que puede ser replicada en otra institución con el objeto de obtener productos adaptados a las necesidades y preferencias de cada Unidad de Desarrollo de Software. Se recomienda ampliar la metodología de trabajo para cubrir aspectos de diseño, construcción, pruebas e implantación de software.

BIBLIOGRAFÍA

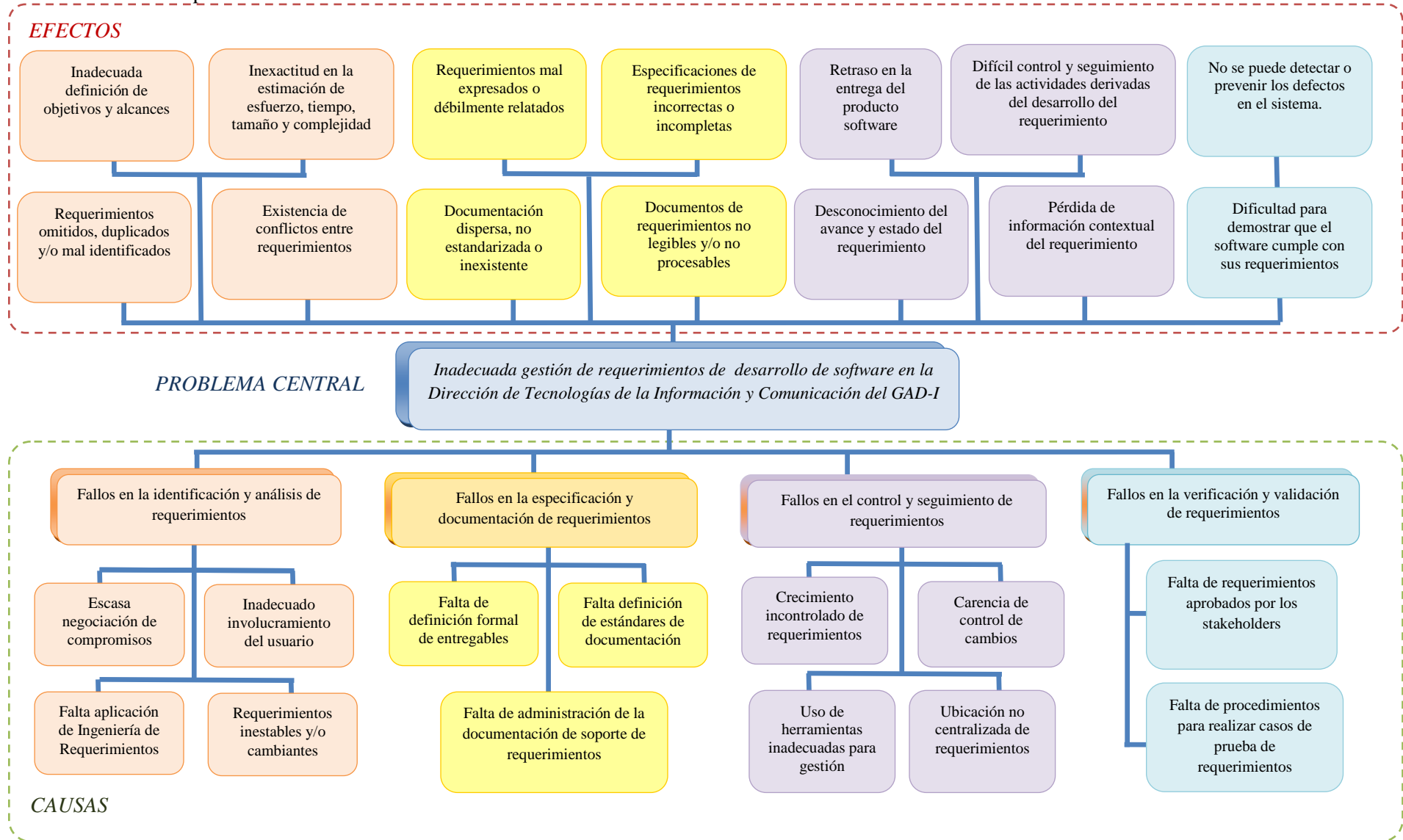
- Accompa, Inc. (2014). *The Definitive Guide to Requirements Traceability*. San Francisco, Estados Unidos.
- Accompa, Inc. (2017). *Accompa*. Retrieved from <http://web.accompa.com>
- ACM Inc. (2017). *Association for Computing Machinery*. Retrieved from <http://www.acm.org/>
- Atlantic Systems Guild. (2016). *Volere Requirements Resources*. Retrieved Agosto 15, 2016, from <http://www.volere.co.uk>
- Atlantic Systems Guild. (2016). *Volere Requirements Resources*. Retrieved from <http://www.volere.co.uk>
- Aurum, A., & Wohlin, C. (2005). *Engineering and Managing Software Requirements*. Berlin: Springer.
- Aurum, A., & Wohlin, C. (2005). *Engineering and Managing Software Requirements*. Berlin, Alemania: Springer.
- Barreno, D. (2009). Propuesta metodológica para especificación de requisitos de software en proyectos pequeños y medianos orientados a objetos. Riobamba, Ecuador: Escuela Superior Politécnica de Chimborazo.
- Berenbach, B., Paulish, D., & Kazmeie, J. (2009). *Software & Systems Requirements Engineering: In practice*. Estados Unidos: McGraw Hill.
- Bourque, P., & Fairley, R. (2014). Guide to the Software Engineering Body of Knowledge - SWEBOK V3. IEEE Computer Society.
- Chemuturi, M. (2013). *Requirements Engineering and Management for Software Development Projects*. Nueva York, Estados Unidos: Springer.
- Club BPM. (2011). *El libro del BPM*. Madrid.
- Contraloría General del Estado. (2009). *Normas de control interno para las entidades, organismos del sector público y de las personas jurídicas de derecho privado que dispongan de recursos públicos*. Quito, Ecuador.
- Eclipse Foundation. (2017). *ProR Requirements Engineering Platform*. Retrieved from <http://www.eclipse.org/rmf/pror>
- Florac, W., Park, R., & Carleton, A. (1997). *Practical Software Measurement: Measuring for Process Management and Improvement*. Software Engineering Institute.
- Foster, E. (2014). *Software Engineering: A Methodical Approach*. Estados Unidos: Apress.
- GAD-I. (2016). *Estatuto Orgánico de Gestión Organizacional por procesos del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra*. Ibarra.

- GAD-I. (2016, Diciembre 23). Estatuto Orgánico de Gestión Organizacional por procesos del Gobierno Autónomo Descentralizado Municipal de San Miguel de Ibarra - Resolución Administrativa N° 286 DAM-2016. Ibarra, Imbabura, Ecuador.
- Gatherspace. (2016). *Gatherspace Requirements Management*. Retrieved from <http://www.gatherspace.com>
- Guerra, L., & Bedini González, A. (2013). *Gestión de Proyectos de Software*. Valparaíso, Chile.
- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. (2010). *Metodología de la Investigación* (Quinta Edición ed.). McGRAW-HILL.
- Hull, E., Jackson, K., & Dick, J. (2011). *Requirements Engineering* (Tercera ed.). Londres, Inglaterra: Springer.
- IEEE. (1998). *IEEE 1012-1998 Standard for software verification and validation*. New York, United States.
- IEEE. (2002). *IEEE Standard Glossary of Software Engineering Terminology*. New York, United States.
- IIBA. (2009). *A Guide to the Business Analysis Body of Knowledge - BABOK V2.0*. Toronto, Canadá.
- Intland Software. (2016). *CodeBeamer Requirements Management*. Retrieved from <https://intland.com/requirements-management>
- ISO/IEC/IEEE. (2011). *International Standard 29148-2011 Systems and software engineering - Life cycle processes --Requirements engineering*. Switzerland.
- Junta de Andalucía de España. (2016). *Marco de Desarrollo de la Junta de Andalucía*. Retrieved from <http://www.juntadeandalucia.es/>
- Kirk, R. (2003). *Software Requirements Engineering: Practices and Techniques*. California: Jet Propulsion Laboratory.
- Lomas, E. (2012). *Sistema de Gestión y Seguimiento de Proyectos de Software*. Ibarra, Ecuador: Universidad Técnica del Norte.
- Ludwig Consulting Services. (2009). *Managing Requirements*. Retrieved from <http://www.jiludwig.com/>
- Manifiesto Ágil*. (2001). Retrieved Diciembre 12, 2016, from <http://agilemanifesto.org/iso/es/manifiesto.html>
- McClure, R. (2001). *Software Engineering - Report on a conference sponsored by the NATO Science Committee*. Arizona.
- McGee, S., & Greer, D. (2009). *A Software Requirements Change Source Taxonomy*. Belfast, United Kingdom: School of Electronics, Electrical Engineering and Computer Science.
- Mette, A., & Hass, J. (2008). *Guide to Advanced Software Testing*. United States: Artech House.

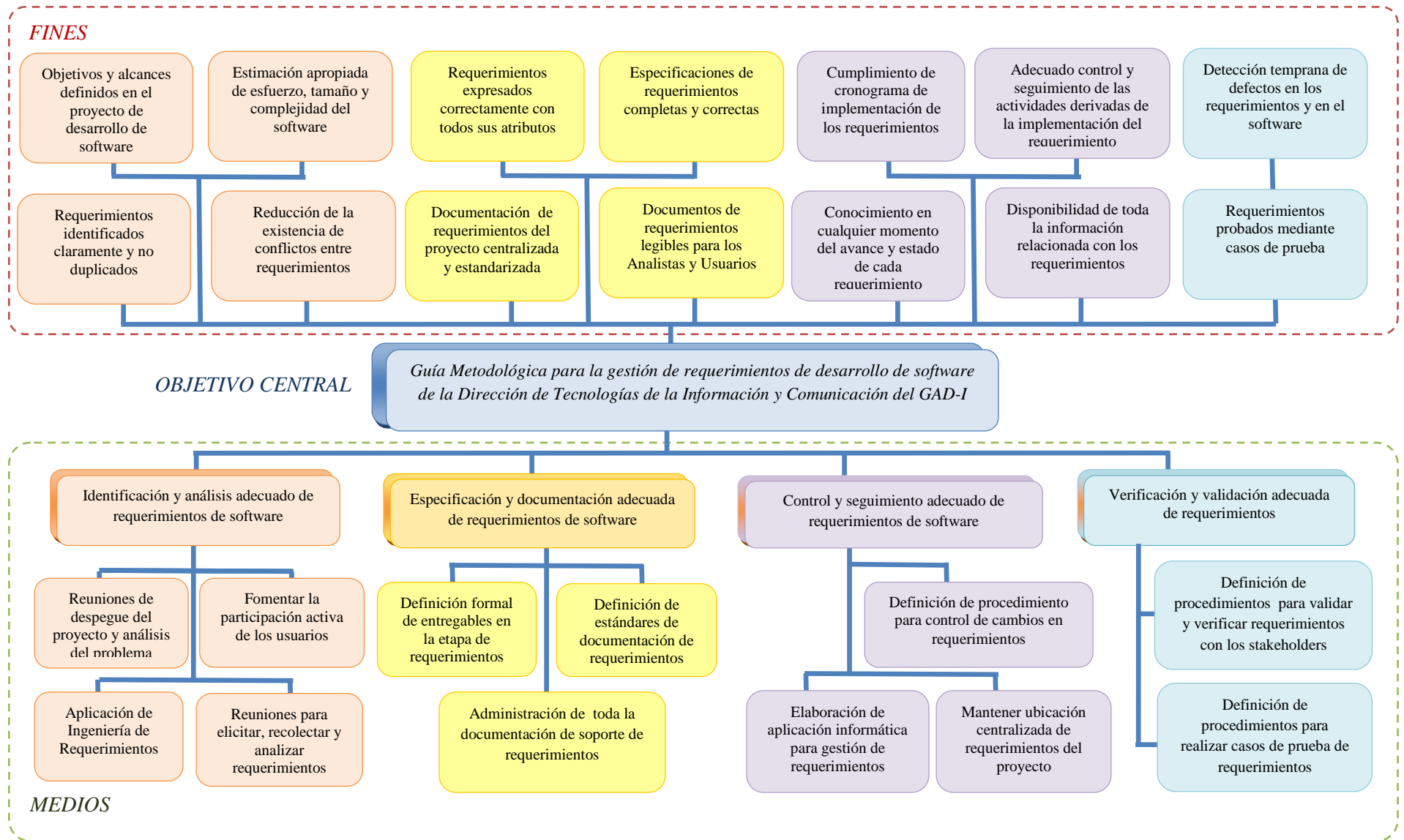
- Ministerio de Coordinación de la Política y Gobiernos Autónomos Descentralizados. (2010, Octubre 19). Código Orgánico de Organización Territorial, Autonomía y Descentralización. Quito, Pichincha, Ecuador.
- Navascués Fernández, J. (2013, Junio 09). *Técnicas avanzadas para la gestión de proyectos software*. Universidad de Sevilla, Escuela Técnica Superior de Ingeniería Informática, Sevilla, España. Retrieved Febrero 10, 2016, from <https://www.lsi.us.es/docs/doctorado/memorias/Memoria-JNavascues.pdf>
- OSSENO Software GmbH. (n.d.). *ReqSuite Requirements Management Tool*. Retrieved from <http://www.osseno.de>
- Palacios, A. (2014, Marzo). *Guía de fundamentos para la dirección de proyectos de desarrollo de software, con enfoque PMI y los métodos ágiles, para la empresa SIACISOLUTIONS S.A.* Escuela Politécnica del Ejército, Sangolquí, Ecuador.
- Pressman, R. (2010). *Ingeniería del software - Un enfoque práctico*. Estados Unidos: McGraw-Hill.
- Rea, A. (2012). *Normas de control interno emitidas por la Contraloría General del Estado aplicadas a la Dirección de Tecnologías de la Información del Ilustre Municipio de Ibarra*. Ibarra, Ecuador.
- Robertson, S., & Robertson, J. (2013). *Mastering the Requirements Process* (Tercera ed.). Washington, Estados Unidos: Microsoft Press.
- Serlio Software. (2017). *CaseComplete*. Retrieved from <http://www.casecomplete.com>
- SGSOFT France. (2016). *REQCHECKER*. Retrieved from <http://reqchecker.eu>
- Sommerville, I. (2005). *Ingeniería del Software*. Addison Wesley.
- Sommerville, I. (2011). *Ingeniería del Software*. Estados Unidos: Addison Wesley.
- Standish Group. (2014). *Caos Report*.
- Varas, M. (2014). *Gestión de Proyectos de Desarrollo de Software*. Cochabamba, Bolivia.
- Wieggers, K., & Beatty, J. (2013). *Software Requirements* (Third Edition ed.). Washington, United States: Microsoft Press.
- Young, R. R. (2004). *The Requirements Engineering Handbook*. Boston, Estados Unidos: Artech House.

ANEXOS

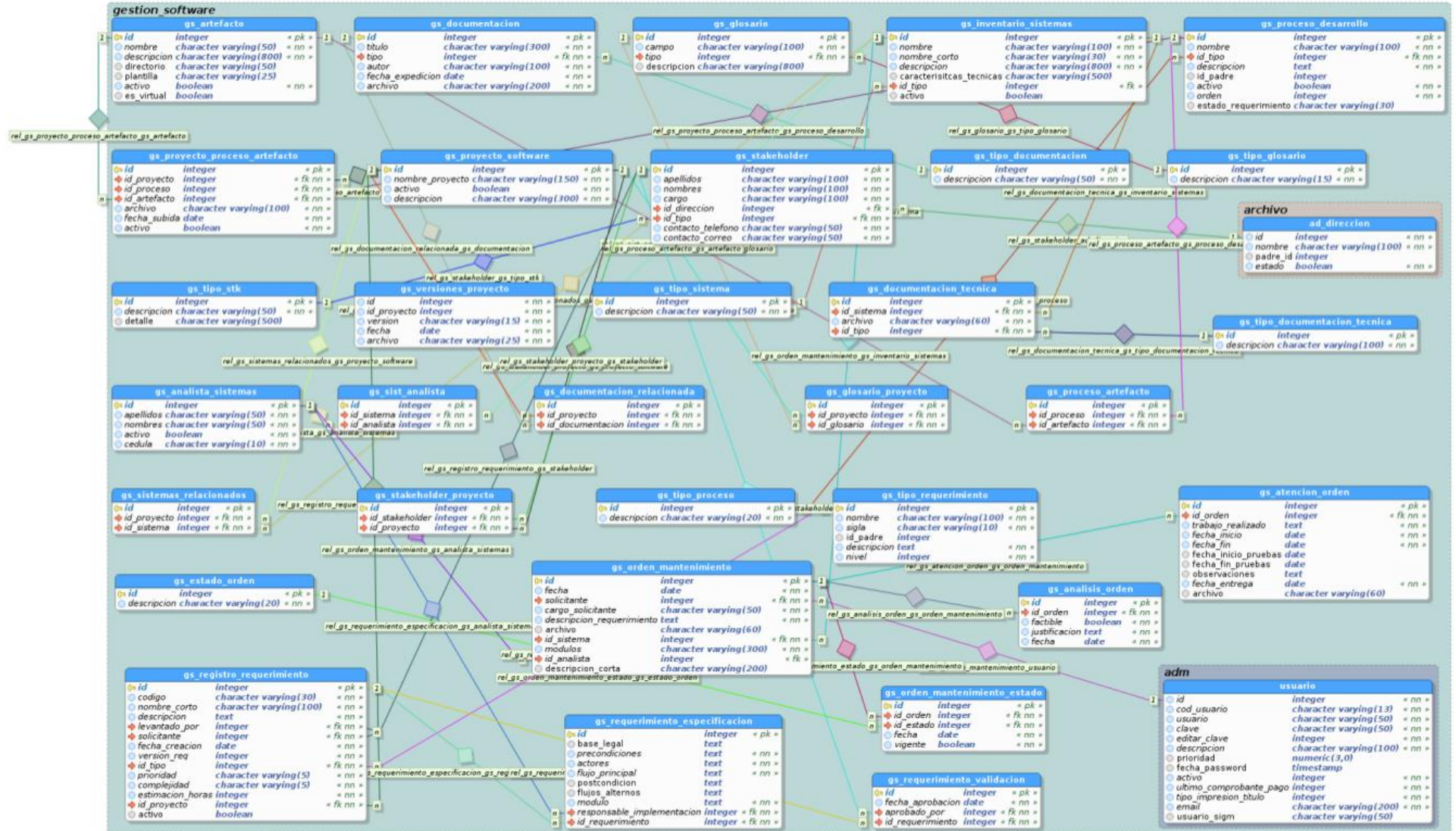
Anexo 1: Árbol de problemas detectados en la Unidad de Desarrollo de Software



Anexo 2: Árbol de Objetivos para la Unidad de Desarrollo de Software



Anexo 3: Modelo Entidad Relación de la aplicación informática desarrollada



Anexo 4: Formato de Acta de Reunión

**GESTIÓN DE TECNOLOGÍAS DE LA INFORMACIÓN Y COMUNICACIÓN
UNIDAD DE DESARROLLO DE SOFTWARE**

ACTA DE REUNION Nro.				
PROYECTO:				
INFORMACIÓN DE LA REUNIÓN				
Fecha:				
Hora de inicio:				
Hora de término:				
Lugar:				
Convocado por:				
Medio de convocatoria:				
Objetivo:				
Tipo de Reunión				
Responsable del Acta				
CONVOCADOS				
Participantes	Asistencia	Dependencia	Área	Función

TEMAS TRATADOS	
Tema tratado N° 1.	Guiado por
Desarrollo	

Se adjuntan firmas de constancia:

Responsable de Desarrollo Software

PROYECTO DE SOFTWARE	
NOMBRE DEL PROYECTO:	<i>Colocar un nombre que identifique al proyecto a ejecutar y que permita identificarlo y diferenciarlo de los demás.</i>
DURACIÓN:	<i>Duración del proyecto</i>
1. DESCRIPCIÓN Y ALCANCE DEL PROYECTO (¿QUÉ?)	
<i>Descripción</i> <i>Se debe detallar lo más claramente posible qué es lo que se quiere realizar, cuál será el resultado final y cómo se va a hacer. Implica explicar qué se va a producir o que servicio va a prestar, cómo lo va a llevar adelante, y quien lo va a ejecutar.</i>	
<i>Alcance</i> <i>El alcance o área de competencia define los límites del proyecto. Decidir qué es lo que está dentro o fuera de los límites del proyecto determinará la cantidad de trabajo que se necesitará realizar.</i>	
2. JUSTIFICACIÓN (¿POR QUÉ?)	
<i>Implica saber, porqué se quiere hacer el proyecto, cuáles son los motivos; no sólo personales sino respecto al impacto que va a tener para la sociedad en su conjunto. Explicar la prioridad y urgencia del problema para el que se busca solución y justificar por qué este proyecto que se formula es la propuesta de solución más adecuada o viable para resolverlo.</i>	
3. OBJETIVOS (¿PARA QUÉ?)	
<i>Para que se quiere hacer el proyecto. Que se quiere lograr. Debe interpretarse adecuadamente la necesidad u oportunidad que tratará de atender.</i>	
4. LUGAR DE EJECUCIÓN (¿DÓNDE?) (EN CASO DE QUE APLIQUE)	
<i>Lugar de ejecución del proyecto.</i>	
5. BENEFICIARIOS DEL PROYECTO (¿PARA QUIENES?)	
<i>Describir los beneficiarios directos o indirectos del proyecto.</i>	
6. DESCRIPCIÓN DE LA SOLUCIÓN TECNOLÓGICA	
<i>En este apartado realizar la descripción de la solución dependiendo del tipo de proyecto: software, hardware, redes, etc. Es libre de colocar aquí toda la parte técnica de la solución como mapas, diagramas y otros.</i>	
7. RESPONSABLES DEL PROYECTO (¿QUIÉNES?)	
<i>Describir el cargo y nombre de los funcionarios responsables de la ejecución del proyecto. En este punto se debe nombrar un servidor responsable con capacidad de decisión y autoridad y administradores o líderes funcionales y tecnológicos con la descripción de sus funciones y responsabilidades de acuerdo al siguiente formato:</i>	

NOMBRE-CARGO RESPONSABLE	DESCRIPCIÓN DE FUNCIONES Y RESPONSABILIDADES

8. RECURSOS (¿CON QUÉ?)

Colocar los recursos materiales, humanos, económicos y técnicos necesarios para llevar a cabo el proyecto.

Humanos	Materiales	Técnicos	Económicos

9. CRONOGRAMA DE ACTIVIDADES (¿CÓMO? Y ¿CUÁNDO?)

La planificación estructura las actividades a realizar dentro del proyecto, definiendo la duración y el orden de ejecución de las mismas. Colocar las actividades de acuerdo a nombre, duración, fecha inicio, fecha fin, actividad predecesora.

N	Actividad	Duración	Fecha Inicio	Fecha Fin	Predecesora

10. RELACIÓN CON OTROS PROYECTOS INSTITUCIONALES

Describir si el proyecto actual tiene relación con otros proyectos ejecutados, en ejecución o por ejecutar.

11. ANÁLISIS DE RIESGOS

N°	DESCRIPCIÓN DEL RIESGO	TIPO DE RIESGO	PROBABILIDAD	RSGR		
				Reducción	Supervisión	Gestión

Donde:

- Descripción del Riesgo:* Describa el riesgo.
- Tipo de Riesgo:* Coloque el tipo de riesgo de acuerdo a: Técnico, Externo, De la Organización, Dirección de Proyectos.
- Probabilidad:* Alta, Media o Baja.
- RSGR*
 - Reducción:** Colocar las estrategias y medidas para evitar que se produzca el riesgo, o reducir y minimizar sus efectos en caso de que se dé.
 - Supervisión:** Colocar las estrategias para controlar y dar seguimiento al proyecto para evitar la ocurrencia del riesgo.
 - Gestión:** Colocar las estrategias para aplicar en caso de que el riesgo se haya materializado y evitar que los efectos se maximicen.

Anexo 6: Datos levantados para la medición de atributos de la entidad Esfuerzo

Medición de atributos de la entidad Esfuerzo			
Grupo:	G02	Sub módulo:	Levantamiento de requerimientos del sub módulo: Declaración de Patente y Activos Totales de Otros Cantones (Original y Sustitutiva)
Atributo:	# de horas de reuniones de elicitación		
Fecha	Hora Inicio	Hora Fin	# Horas
23/10/2017	15:00	17:10	02h. 10m.
27/10/2017	10:00	12:25	02h. 25m.
07/11/2017	15:00	17:15	02h. 15m.
08/11/2017	09:00	11:30	02h. 30m.
Total Horas			09h. 20m.
Atributo:	# de horas de reuniones de análisis		
Fecha	Hora Inicio	Hora Fin	# Horas
08/11/2017	09:00	10:15	01h. 15m.
10/11/2017	10:00	12:05	02h. 05m.
Total Horas			03h. 20m.
Atributo:	# de horas de trabajo de especificación		
Fecha	Hora Inicio	Hora Fin	# Horas
--	--	--	--
Total Horas			0h. 0m.
Atributo:	# de horas de reuniones de validación		
Fecha	Hora Inicio	Hora Fin	# Horas
--	--	--	--
Total Horas			0h. 0m.

Medición de atributos de la entidad Esfuerzo			
Grupo:	G01	Sub módulo:	Levantamiento de requerimientos del sub módulo: Declaración de Patente y Activos Totales del Cantón Ibarra (Original y Sustitutiva)
Atributo:	# de horas de reuniones de elicitación		
Fecha	Hora Inicio	Hora Fin	# Horas
23/10/2017	08:30	12:15	03h. 45m
25/10/2017	14:30	17:30	03h. 0m.
26/10/2017	09:00	12:20	03h. 20m.
27/10/2017	15:00	17:25	02h. 25m.
30/10/2017	08:45	12:30	03h. 45m.
Total Horas			16h. 15m.
Atributo:	# de horas de reuniones de análisis		
Fecha	Hora Inicio	Hora Fin	# Horas
31/10/2017	08:15	12:10	03h. 55m.
01/11/2017	09:00	12:30	03h. 30m.
Total Horas			07h. 25m.
Atributo:	# de horas de trabajo de especificación		
Fecha	Hora Inicio	Hora Fin	# Horas
06/11/2017	10:30	12:30	02h. 0m.
07/11/2017	09:30	12:30	03h. 0m.
08/11/2017	09:00	12:00	03h 0m.
Total Horas			08h. 0m.
Atributo:	# de horas de reuniones de validación		
Fecha	Hora Inicio	Hora Fin	# Horas
08/11/2017	15:00	17:20	02h. 20m.
09/11/2017	09:00	10:45	01h. 45m.
Total Horas			04h. 05m.

Anexo 7: Datos levantados para la medición de atributos de la entidad Unidad de Trabajo

Medición de atributos de la entidad Unidad de Trabajo		
Grupo:	G02	Sub módulo: Levantamiento de requerimientos del sub módulo: Declaración de Patente y Activos Totales de Otros Cantones (Original y Sustitutiva)
Atributo:	# de requerimientos analizados	
Código	Nombre	Tipo
Sin Código	Registro de Declaración Original de Patente de Otros Cantones	Funcional
Sin Código	Registro de Declaración Sustitutiva de Patente Otros cantones	Funcional
Sin Código	Registro de Declaración Original de Activos Totales Otros Cantones	Funcional
Sin Código	Registro de Declaración Sustitutiva de Activos Totales Otros cantones	Funcional
Total Requerimientos		4
Atributo:	# de requerimientos documentados	
Código	Nombre	Tipo
--	--	--
Total Requerimientos		0
Atributo:	# de requerimientos validados	
Código	Nombre	Tipo
--	--	--
Total Requerimientos		0

Medición de atributos de la entidad Unidad de Trabajo		
Grupo:	G01	Sub módulo: Levantamiento de requerimientos del sub módulo: Declaración de Patente y Activos Totales del Cantón Ibarra (Original y Sustitutiva)
Atributo:	# de requerimientos analizados	
Código	Nombre	Tipo
RU-001	Registro de datos de Contador en actividad económica	Funcional
RU-002	Registro de datos de terreno e inmueble (construcción) en actividad económica	Funcional
RU-003	Registro de Declaración ORIGINAL de Patente del cantón Ibarra (En línea)	Funcional
RU-004	Revisión de Declaración de Patente y Activos Totales Cantón Ibarra	Funcional
RU-005	Registro de Declaración SUSTITUTIVA de Patente Cantón Ibarra (En línea)	Funcional
RU-006	Registro de Declaración ORIGINAL Activos Totales del cantón Ibarra (En línea)	Funcional
RU-007	Registro de Declaración SUSTITUTIVA Activos Totales Cantón Ibarra (En línea)	Funcional
RU-008	Emisión de valores por concepto de Declaración de Patente y Activos Totales (Original y Sustitutiva)	Funcional
RU-009	Validación de información de la declaración con datos de la réplica del SRI	Funcional
RU-010	Validación de cuenta de Portal Ciudadano para acceso a Declaraciones en Línea	Funcional
Total Requerimientos		10
Atributo:	# de requerimientos documentados	
Código	Nombre	Tipo
RU-001	Registro de datos de Contador en actividad económica	Funcional
RU-002	Registro de datos de terreno e inmueble (construcción) en actividad económica	Funcional

RU-003	Registro de Declaración ORIGINAL de Patente del cantón Ibarra (En línea)	Funcional
RU-004	Revisión de Declaración de Patente y Activos Totales Cantón Ibarra	Funcional
RU-005	Registro de Declaración SUSTITUTIVA de Patente Cantón Ibarra (En línea)	Funcional
RU-006	Registro de Declaración ORIGINAL Activos Totales del cantón Ibarra (En línea)	Funcional
RU-007	Registro de Declaración SUSTITUTIVA Activos Totales Cantón Ibarra (En línea)	Funcional
RU-008	Emisión de valores por concepto de Declaración de Patente y Activos Totales (Original y Sustitutiva)	Funcional
RU-009	Validación de información de la declaración con datos de la réplica del SRI	Funcional
Total Requerimientos		9
Atributo:	# de requerimientos validados	
Código	Nombre	Tipo
RU-001	Registro de datos de Contador en actividad económica	Funcional
RU-002	Registro de datos de terreno e inmueble (construcción) en actividad económica	Funcional
RU-003	Registro de Declaración ORIGINAL de Patente del cantón Ibarra (En línea)	Funcional
RU-004	Revisión de Declaración de Patente y Activos Totales Cantón Ibarra	Funcional
RU-005	Registro de Declaración SUSTITUTIVA de Patente Cantón Ibarra (En línea)	Funcional
RU-006	Registro de Declaración ORIGINAL Activos Totales del cantón Ibarra (En línea)	Funcional
RU-007	Registro de Declaración SUSTITUTIVA Activos Totales Cantón Ibarra (En línea)	Funcional
RU-008	Emisión de valores por concepto de Declaración de Patente y Activos Totales (Original y Sustitutiva)	Funcional
RU-009	Validación de información de la declaración con datos de la réplica del SRI	Funcional
Total Requerimientos		9

**GOBIERNO AUTÓNOMO DESCENTRALIZADO
MUNICIPAL DE SAN MIGUEL DE IBARRA**



***“GUÍA METODOLÓGICA PARA LA GESTIÓN DE
REQUERIMIENTOS DE DESARROLLO DE SOFTWARE
DEL GOBIERNO AUTÓNOMO DESCENTRALIZADO
MUNICIPAL DE SAN MIGUEL DE IBARRA”***

Fecha: Noviembre del 2017

Autora: Ing. Andrea Rea