

**PROCEDIMIENTO PARA ESPECIFICAR Y VALIDAR REQUISITOS DE SOFTWARE
EN MIPYMES DESARROLLADORAS DE SOFTWARE DE LA CIUDAD DE PEREIRA,
BASADO EN ESTUDIOS PREVIOS EN LA REGIÓN**

ALONSO TORO LAZO

JUAN GUILLERMO GÁLVEZ BOTERO

Marzo 2017

Universidad Autónoma de Manizales

Facultad De Ingenierías

Maestría en Gestión y Desarrollo de Proyectos de Software

Dedicatoria

A Dios, a mis padres Guillermo Gálvez Giraldo y Dora Gilma Botero Chalarca, a Alexandra Piedrahita González mi compañera incondicional, quiero dedicar con todo mi corazón y sentimiento este nuevo logro ya que gracias al amor, apoyo y paciencia recibido; he logrado concluir con éxito un proyecto que en un principio podría parecer una tarea titánica, colosal e interminable. Quisiera dedicar mi tesis a ellos, personas de bien, seres que ofrecen amor, bienestar, confianza y los más finos deleites de la vida.

Juan Guillermo Gálvez Botero

*A Dios, que me dio sabiduría y salud para terminar este proyecto de vida.
A mi madre y herman@s, por apoyar mis sueños...
A Pilar, mi fortaleza y apoyo incondicional, por su amor y paciencia.
A mi compañero de tesis Juan Guillermo, por su entrega y dedicación.*

Alonso Toro Lazo

Agradecimientos

Queremos agradecer sinceramente a nuestra directora de tesis M.Sc. Sandra Victoria Hurtado Gil por su orientación, conocimiento y apoyo durante esta labor.

A los evaluadores de este proyecto Ph.D. Mauricio Fernando Alba Castro de la Universidad Autónoma de Manizales y al M.Sc. Faber Danilo Giraldo Velásquez que sin duda alguna contribuyeron al desarrollo de este trabajo por medio de sus apreciaciones.

A los grupos de investigación Ingeniería de software de la Universidad Autónoma de Manizales y GIII de la Universidad Católica de Pereira, por apoyarnos en este proceso.

A las MiPymes desarrolladoras de software de la ciudad de Pereira por permitirnos conocer su proceso de desarrollo de requisitos y hacer parte de la validación del procedimiento elaborado.

Al M.Sc. Luis Eduardo Peláez Valencia por sus aportes y a todas aquellas personas que compartieron con nosotros sus conocimientos para hacer posible la conclusión de esta tesis.

Alonso Toro Lazo
Juan Guillermo Gálvez Botero

Resumen

El presente documento pretende dar cuenta de un trabajo de investigación que responde a la necesidad de indagar sobre los problemas de calidad del sector del software, y específicamente a lo relacionado con el tratamiento que se les hace a los requisitos como un punto clave para lograr productos y procesos de calidad, con el ánimo de desarrollar propuestas que contribuyan a mejorar dicha calidad, inicialmente en la industria que se representa desde la ciudad de Pereira. Para ello, el proyecto plantea la creación de un procedimiento para especificar y validar requisitos de software basado en estudios previos en la región, que es ejecutado en forma de pilotos de aplicación en Micro, Pequeñas y Medianas empresas desarrolladoras de software de la ciudad de Pereira (MiPymes).

La presente investigación plantea el desarrollo de las siguientes fases:

- Levantamiento del estado del arte de los requisitos de software.
- Descripción y caracterización de la forma en que las MiPymes desarrolladoras de software de la ciudad de Pereira llevan a cabo la especificación y validación de requisitos.
- Diseño del procedimiento para la especificación y validación de requisitos de software.
- Validación del procedimiento de especificación y validación de requisitos de software mediante dos pilotos de aplicación.
- Realización de ajustes al procedimiento propuesto.

Como resultado se obtuvo un procedimiento estructurado a través de una secuencia de actividades puntuales, que establecen el uso de técnicas soportadas desde el referente teórico, el referente contextual y el diagnóstico de las empresas, el cual proporciona además un conjunto de guías, formatos y herramientas que permiten a la persona encargada de especificar los requisitos, conseguir una definición correcta de los mismos en un documento SRS alineado con el procedimiento, basado en estándares mundialmente reconocidos y validado por el usuario.

Finalmente, y como conclusión, la aplicación del procedimiento en las empresas partícipes de la validación logró que los equipos responsables del desarrollo de los requisitos alcanzaran niveles de satisfacción que dan cuenta de los buenos resultados obtenidos en sus proyectos una vez aplicado el procedimiento, quedando satisfechos por los resultados percibidos.

Tabla de contenido

1.	PRESENTACIÓN	1
2.	REFERENTE CONTEXTUAL.....	4
2.1.	Área problemática.....	4
2.2.	Antecedentes	8
2.2.1.	Del orden regional	8
2.2.2.	Del orden nacional.....	12
2.2.3.	Del orden internacional.....	15
2.3.	Problema de investigación	24
3.	JUSTIFICACIÓN.....	25
4.	OBJETIVOS	26
4.1.	Objetivo General	26
4.2.	Objetivos Especificos	26
5.	REFERENTE TEÓRICO	27
5.1.	Requisito Vs. Requerimiento.....	27
5.2.	Ingeniería de Requisitos	28
5.2.1.	Estructura de la Ingeniería de Requisitos	30
5.3.	Comité Internacional de Ingeniería de Requisitos	34
5.4.	Clasificación de los Requisitos	34
5.4.1.	Requisitos de usuario	34
5.4.2.	Requisitos del sistema.....	34
5.4.2.1.	Requisitos funcionales.....	34
5.4.2.2.	Requisitos no funcionales.....	35
5.4.2.3.	Requisitos de dominio	36
5.5.	Niveles de descripción de los requisitos	36
5.5.1.	A nivel de negocio	36
5.5.2.	A nivel de usuario	37
5.5.3.	A nivel de sistema	37
5.6.	Características de un buen requisito	38
5.7.	Especificación de requisitos de software	39
5.7.1.	El Documento de definición del sistema	39
5.7.2.	El Documento de especificación de requisitos de sistema	39

5.7.3.	El Documento de especificación de requisitos de software	39
5.7.4.	Características de una SRS.....	40
5.7.5.	Especificación de requisitos de software en lenguaje natural.....	41
5.8.	Técnicas de Ingeniería de Requisitos tomadas de metodologías ágiles.....	42
5.9.	Las Pequeñas y Medianas Empresas del software	45
5.10.	Software Process Engineering Metamodel (SPEM).....	46
6.	ESTRATEGIA METODOLÓGICA.....	47
7.	RESULTADOS	54
7.1.	Análisis del estado del arte de los requisitos	54
7.1.1.	Análisis de Técnicas para Especificar y Validar Requisitos extraídas del Marco Teórico	54
7.1.2.	Análisis de Técnicas para Especificar y Validar Requisitos extraídas de los Antecedentes	55
7.1.3.	Cruce entre Técnicas del Marco Teórico y los Antecedentes	57
7.2.	Diagnóstico de las Empresas	62
7.2.1.	Análisis de resultados de la encuesta aplicada a las empresas.....	63
7.2.2.	Uso de técnicas por parte de las empresas.....	64
7.3.	Diseño del procedimiento para la Especificación y Validación de requisitos de software.....	65
7.4.	Aplicación del procedimiento propuesto	75
7.4.1.	Piloto de aplicación Empresa No. 1.....	76
7.4.2.	Piloto de Aplicación Empresa No. 2.....	85
8.	DISCUSIÓN DE RESULTADOS.....	99
9.	CONCLUSIONES	103
10.	RECOMENDACIONES	105
11.	REFERENCIAS BIBLIOGRÁFICAS.....	106
12.	ANEXOS.....	111

Lista de tablas

Tabla 1. Especialización de tipo de empresa. Cuadro 15 (FEDESOFTE, 2013, pág. 55).....	5
Tabla 2. Distribución de las empresas por código CIIU. Cuadro 9 (FEDESOFTE, 2013, pág. 51)	5
Tabla 3. Antecedentes regionales sobre especificación y validación de requisitos de software	9
Tabla 4. Antecedentes nacionales sobre especificación de requisitos de software	13
Tabla 5. Antecedentes internacionales sobre especificación de requisitos de software.....	17
Tabla 6. Herramientas que se utilizan en la ingeniería de requisitos (Dávila, 2001, pág. 13)	33
Tabla 7. Caracterización de Pequeñas y Medianas Empresas en Colombia	45
Tabla 8. Antecedentes relevantes	48
Tabla 9. Información de la muestra	49
Tabla 10. Técnicas extraídas del marco teórico.....	50
Tabla 11. Técnicas extraídas de los antecedentes.....	50
Tabla 12. Tabla de cruce de información para análisis.....	50
Tabla 13. Análisis de técnicas usadas por las empresas	50
Tabla 14. Formato propuesto para el procedimiento	51
Tabla 15. Cubrimiento de acuerdo a peso.....	54
Tabla 16. Técnicas para especificación de requisitos de software extraídas del marco teórico.....	54
Tabla 17. Técnicas para validación de requisitos de software extraídas del marco teórico.....	55
Tabla 18. Clasificación de antecedentes y asignación de peso	55
Tabla 19. Técnicas de especificación de requisitos de software extraídas de los antecedentes.....	56
Tabla 20. Técnicas de validación de requisitos de software extraídas de los antecedentes.....	56
Tabla 21. Definición de cubrimiento según rango.....	57
Tabla 22. Tabla de cruce de información para análisis de técnicas de especificación de requisitos de software	57
Tabla 23. Tabla de cruce de información para análisis de técnicas de validación de requisitos de software	59
Tabla 24. Ventajas y desventajas de las técnicas seleccionadas	59
Tabla 25. Clasificación de preguntas por temas	62
Tabla 26. Ficha técnica de la encuesta.....	63
Tabla 27. Técnicas de especificación de requisitos de software según información de las empresas	64
Tabla 28. Técnicas de validación de requisitos de software según información obtenida de las empresas.....	65
Tabla 29. Planeación general del piloto de aplicación.....	75
Tabla 30. Diseño del piloto de aplicación No.1	77
Tabla 31. Identificación del proyecto de referencia.....	78
Tabla 32. Identificación del proyecto a intervenir	78
Tabla 33. Datos recolectados de los proyectos del piloto de aplicación No.1	79
Tabla 34. Situaciones presentadas durante la aplicación del piloto de aplicación No.1	80
Tabla 35. Nivel de satisfacción según rango	85
Tabla 36. Diseño del piloto de aplicación No.2.....	86
Tabla 37. Identificación del proyecto de referencia.....	87
Tabla 38. Identificación del proyecto a intervenir	87
Tabla 39. Datos recolectados de los proyectos del piloto de aplicación No.2	88
Tabla 40. Situaciones presentadas durante la aplicación piloto de aplicación No.2	89
Tabla 41. Nivel de satisfacción según rango	93
Tabla 42. Datos encuesta de expectativas de los dos pilotos de aplicación	94

Tabla 43. Resultado del alfa de Cronbach	94
Tabla 44. Datos encuesta de rendimiento de los dos pilotos de aplicación	95
Tabla 45. Resultado del alfa de cronbach	95
Tabla 46. Ajustes realizados al procedimiento de especificación y validación de requisitos	96

Lista de Ilustraciones

Ilustración 1. Participación por tamaño año 2014 (SUPERSOCIEDADES, 2015).....	5
Ilustración 2. Proporción de errores en cada una de la fases de desarrollo. (Torres de Paz, 2009)	6
Ilustración 3. Factores de falla o cancelación en los proyectos (Cristiá, 2011)	7
Ilustración 4. Requisito Vs Requerimiento (Pagani, 2013)	28
Ilustración 5. Estructura de la ingeniería de requisitos (Wiegers & Beaty, 2013, pág. 15)	30
Ilustración 6. Proceso iterativo de desarrollo de requisitos (Wiegers & Beaty, 2013, pág. 45).....	30
Ilustración 7. Tipos de Requisitos no funcionales (Sommerville, 2011, pág. 88).....	35
Ilustración 8. Relación entre niveles de descripción (Wiegers & Beaty, 2013, pág. 8).....	37
Ilustración 9. Niveles de descripción en el proceso de requisitos (Camacho Zambrano, 2005, pág. 68)	38
Ilustración 10. Características de un buen requisito. (elaboración propia a partir de Wiegers & Beaty, 2013).....	38
Ilustración 11. Cualidades de una especificación de requisitos de calidad (elaboración propia a partir de Camacho Zambrano, 2005)	40
Ilustración 12. Mejores prácticas de AM (Ambler, 2014)	42
Ilustración 13. Técnicas y artefactos de AM (elaboración propia a partir de (Burgos Pintos & Garbarino de la Rosa, 2010).....	43
Ilustración 14. Estructura del Ciclo PHVA (Fuente: http://www.pdcahome.com/5202/ciclo-pdca/)	47
Ilustración 15. PEVReS frente a Metodología propuesta por (De la Cruz Londoño & Castro Guevara, 2014)	66
Ilustración 16. Actividades que componen el procedimiento de especificación de requisitos	67
Ilustración 17. Actividades que componen el procedimiento de validación de requisitos.....	72
Ilustración 18. Comparación de tiempos del piloto de aplicación No.1	81
Ilustración 19. Comparación de revisiones, modelos y documentos del piloto de aplicación No.1	81
Ilustración 20. Defectos encontrados en ambos proyectos	82
Ilustración 21. Expectativas del equipo en el piloto de aplicación No.1	83
Ilustración 22. Rendimiento del procedimiento en el piloto de aplicación No.1	84
Ilustración 23. Comparación de tiempos del piloto de aplicación No.2	89
Ilustración 24. Comparación de revisiones, modelos y documentos del piloto de aplicación No.2	90
Ilustración 25. Comparación de datos obtenidos piloto de aplicación No.2	91
Ilustración 26. Expectativas del equipo en el piloto de aplicación No.2	92
Ilustración 27. Rendimiento del procedimiento piloto de aplicación No.2.....	92
Ilustración 28. Nivel de consistencia alfa de cronbach (BIOESTADISTICO, 2014)	93
Ilustración 28. Información general sobre PEVReS	97
Ilustración 29. Actividades de especificación del procedimiento	97
Ilustración 30. Despliegue de tareas en PEVReS	98

1. PRESENTACIÓN

Cada día los expertos de la industria y la academia en Ingeniería del Software transmiten al mundo la necesidad de desarrollar software de calidad; en diferentes cursos, congresos, charlas, debates y foros, entre otros, que traten sobre temas relacionados con el desarrollo de software se refuerza la idea de que la calidad es de vital importancia. En la industria del software ya sea nacional o internacional el objetivo primordial que se debería alcanzar es obtener productos software de calidad aplicando diferentes métodos y modelos bajo procesos estandarizados y maduros de gestión y desarrollo de software, junto con técnicas, estrategias y todos aquellos aspectos que contribuyan a la búsqueda de conseguir calidad en los productos intangibles.

Es de notar que los sistemas de software son cada vez más complejos, por lo que las exigencias de calidad sobre su desarrollo son mayores. Es necesario que el desarrollo de software sea más riguroso para obtener un producto de alta calidad. Esto busca reducir el número de errores presentes en su construcción. La identificación y tratamiento de estos errores en etapas tempranas del proyecto de software es crucial para disminuir los costos de su operación y evitar errores en etapas posteriores del desarrollo.

Una de las principales etapas en el desarrollo de software tiene que ver con el desarrollo de los requisitos, la cual a su vez se divide en cuatro etapas entre las cuales cobra importancia la especificación de requisitos, ésta se refiere típicamente a la producción de un documento, o a su equivalente electrónico, que puede estar sistemáticamente repasado, evaluado, y aprobado (IEEE, 2014, pág. 41)¹. Para lograr un producto de software de alta calidad, los requisitos deben cumplir con ciertas propiedades (Wiegiers & Beaty, 2013) citado por (Salazar Osorio, Rengifo Romero, Machuca Villegas, & Aranda Bueno, 2012), tales como completo, correcto, realizable y verificable, entre otras. Sin embargo, en algunas ocasiones los requisitos no satisfacen dichas propiedades, lo que se traduce en errores y en mala calidad del producto que se está desarrollando.

Si bien durante el trabajo con los requisitos es donde se interactúa por primera vez con el cliente que carece de una solución a la necesidad que posee, también es fundamental para conseguir que el producto desarrollado cumpla con las expectativas trazadas por el usuario con el fin de que se satisfaga sus necesidades.

Por esto, la gestión de requisitos en la Ingeniería de Software se ha convertido en una de las principales estrategias para garantizar desde las primeras etapas del proceso de desarrollo de software, la calidad de las aplicaciones resultantes (García Ramírez & Puello Marrugo, 2010).

Así mismo, a través del tiempo se ha podido evidenciar que los requisitos son de vital importancia en un proyecto de desarrollo de software, ya que se convierten en el elemento fundamental que permite entender lo que el cliente requiere, analizar sus necesidades, constatarlas, y administrar estos requisitos conforme se avanza en el desarrollo del proyecto. Además, marcan el punto de partida para actividades como la planeación, en lo que se refiere a las estimaciones de tiempos y costos, así como la definición de recursos necesarios, la elaboración de cronogramas que será uno de los principales mecanismos de control con los que se

¹ The Computer Society of the Institute of Electrical and Electronics Engineers (IEEE)

contará durante la etapa de desarrollo (Arias Chaves, 2006) y verificar si se alcanzaron o no los objetivos establecidos en el proyecto.

Al respecto, menciona también (Arias Chaves, 2006, pág. 1) que es muy frecuente escuchar entre los conocedores del desarrollo de software que “*un gran número de los proyectos de software fracasan por no realizar una adecuada definición, especificación, y administración de los requisitos.*”. Así mismo, su inadecuada gestión puede conllevar a la presentación de problemas relacionados con el poco involucramiento del usuario, incompletitud de requisitos y mala trazabilidad de los mismos.

En la misma línea, la especificación y validación de requisitos cumplen un papel fundamental en el proceso de desarrollo de software, pues se centran en la adecuada definición de lo que se requiere producir. Llevarlas a cabo permite conseguir la definición de especificaciones correctas, claras y no ambigüas de las necesidades de los clientes, evitando problemas causados por un deficiente desarrollo de los requisitos.

Lo anterior permite vislumbrar la importancia que tiene para la calidad del software el implementar en cada proyecto de desarrollo de software una correcta ingeniería de requisitos. Por ello, el presente proyecto de investigación pretende definir un procedimiento para especificar y validar requisitos de software que pueda ser utilizado por las MiPymes² desarrolladoras de software de la ciudad de Pereira, con la intención de contribuir en el mejoramiento de la calidad de sus procesos y productos. Lo anterior, a través de la revisión de estudios previos que alrededor de los requisitos se han realizado en la región.

Para lograr lo anterior, el presente proyecto se realizó siguiendo las etapas propuestas por el Ciclo PHVA (Planear-Hacer-Verificar-Actuar)³. Esto se hace considerando que se desea elaborar un procedimiento como un elemento de mejora para uno de los procesos de desarrollo de software, mediante una investigación que constó inicialmente de una exploración que se realizó con respecto a la especificación y validación de requisitos, teniendo en cuenta tanto el estado del arte como una descripción y caracterización de una muestra por conveniencia representativa de la población de empresas desarrolladoras de software de la ciudad de Pereira. De igual forma, se elaboró un procedimiento para la especificación y validación de requisitos de software, así como la validación del mismo a través de su aplicación mediante dos pilotos de aplicación en dos unidades de análisis representadas por dos MiPymes desarrolladoras de software.

Como resultados principales se logró el diseño y aplicación de un procedimiento para especificar y validar requisitos de software; el levantamiento del estado de arte de los requisitos en el ámbito regional, nacional e internacional; así mismo una descripción y caracterización de la forma en que las MiPymes desarrolladoras de software de la ciudad de Pereira llevan a cabo la especificación y validación de requisitos.

El presente proyecto se logra enmarcar en el contexto de dos iniciativas que promueven la intención de abordar el tema de requisitos para la región a nivel de investigación: la primera, la

² Hace referencia a Micro, Pequeñas y Medianas Empresas (MiPymes)

³ También conocido como Círculo de Deming, ciclo PDCA (Plan-Do-Check-Act) por sus siglas en inglés o Espiral de mejora continua, propuesto por Edwards Deming.

marcada necesidad de la industria local, específicamente a las MiPymes dedicadas al proceso de desarrollo de software; la segunda, la consonancia que se encuentra al complementar la etapa de requisitos de un antecedente inmediato llevado a cabo en la ciudad de Pereira por la Universidad Católica de Pereira mediante el proyecto de investigación denominado “*Propuesta metodológica para estandarizar el proceso de construcción y evaluación del producto software que permita a las Pymes Colombianas medir la calidad del software*”, elaborado por (Peláez Valencia, 2011).

Sobre la primera iniciativa, el presente proyecto ha resaltado la necesidad de orientar apuestas alrededor de la mejora de la calidad del software de tal manera que las MiPymes desarrolladoras logren hacerse más competitivas desde el orden local y con fines de ubicarse en un escenario global y exigente.

Respecto a los avances logrados en la segunda iniciativa, y propuestos como futuras líneas de investigación, postulado adoptado por este proyecto, donde se concluye que “las MiPymes de la ciudad de Pereira requieren adoptar mejores prácticas en todas las fases del proceso de desarrollo de tal manera que éstas puedan redundar en un producto software de más y mejor calidad; de tal forma que sus clientes directos, locales la mayoría, aumenten el nivel de satisfacción frente a los entregables comprometidos y ellas mismas aumenten sus expectativas de competitividad a nivel regional y nacional a medida que vayan logrando mejores productos con procesos de distribución menos dependientes de soporte y mantenimiento” (Peláez Valencia, 2011).

En la línea de lo anterior, trabajar en un proyecto que tenga como propósito entregar mejores prácticas para la especificación y validación de requisitos, continuaría el trabajo conducente a mejorar los niveles de calidad en la industria del software.

Este documento está dividido en 5 partes. La primera consiste en una revisión teórica de los requisitos de software a partir de antecedentes regionales, nacionales e internacionales relacionados con MiPymes, así como autores ampliamente reconocidos en la materia por sus citaciones en diferentes publicaciones, con la intención de establecer un estado del arte de los requisitos. La segunda, presenta una descripción y caracterización de la forma en que las MiPymes desarrolladoras de software de la ciudad de Pereira llevan a cabo la especificación de requisitos. La tercera y cuarta parte contemplan el diseño y la validación del procedimiento para especificar y validar requisitos de software, respectivamente. Finalmente, la quinta parte hace referencia a la realización de ajustes al procedimiento una vez realizada la validación mediante pilotos de aplicación.

2. REFERENTE CONTEXTUAL

2.1. Área problemática

El software como parte intangible

El software consiste, de acuerdo a (IEEE Computer Society, 1990), en “*un conjunto de los programas de cómputo, procedimientos, reglas, documentación y datos asociados, que forman parte de las operaciones de un sistema de computación*”. El mismo, permite a los distintos dispositivos (computadoras, teléfonos móviles, teléfonos inteligentes y tabletas, entre otros) realizar las operaciones necesarias. De esta manera, según la Conferencia de las Naciones Unidas sobre comercio y Desarrollo (UNCTAD, 2012) el software puede ayudar a las empresas -según el mismo organismo-, “*a gestionar mejor sus recursos, encontrar la información que requieren, reducir los costos de sus operaciones y agilizar la comercialización*”.

Es por esto, que el software se ha convertido en uno de los ejes de la sociedad de la información. Al ser una tecnología transversal con efectos multiplicadores en otros sectores, su aplicación tiene repercusiones para las empresas de todos los tamaños, para los gobiernos y para las personas. Por consiguiente, las capacidades nacionales en materia de software son cada vez más importantes para que los países puedan crear una sociedad de la información inclusiva. (UNCTAD, 2012).

La industria del software, como parte tangible

Como se mencionó en la introducción, una de las principales problemáticas de la industria del software es el proceso de requisitos que se aplica y su efecto en el producto y su calidad. Es así como la industria del software servirá como objeto de descripción en este proyecto en el cual es importante introducir al lector en diferentes aspectos que se relacionan a continuación.

La industria del Software, como una de las tantas clases de industrias que existen en Colombia y en el mundo entero, juega un papel muy importante para el desarrollo integral de todo país; esto se debe a que el software como producto intangible de esta industria educa, genera empleo, brinda bienestar, satisface necesidades y sobre todo contribuye al desarrollo tecnológico e integral de toda sociedad; por ende al ser una industria que ofrece tanto potencial para la sociedad tiene, como lo indica el Ministerio de Comercio, Industria y Turismo de Colombia (2008), citado en (Lopera Carmona, 2012, pág. 3), “*un gran componente de conocimiento, por lo que requiere un alto desempeño en investigación, desarrollo tecnológico y formación de personas capaces de producir conocimiento y soluciones acordes a las necesidades de todo tipo de organizaciones que lo requieran*”.

Por otro lado, es importante mencionar que la industria del software en Colombia está dominada por pequeñas y medianas empresas (Pymes) (SUPERSOCIEDADES, 2015), según la *Ilustración 1*, las mismas, están dedicadas especialmente al desarrollo de software a medida (CEPAL, 2009, pág. 153).

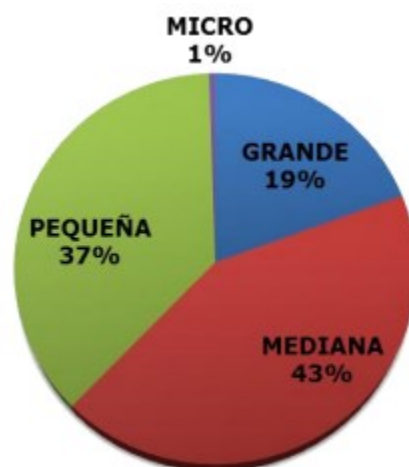


ILUSTRACIÓN 1. PARTICIPACIÓN POR TAMAÑO AÑO 2014 (SUPERSOCIEDADES, 2015)

También menciona (FEDESOFTE, 2013, pág. 55), que de acuerdo a la especialización por tipo de empresa según el porcentaje de su facturación, durante el año 2012, el tamaño de las mismas se puede clasificar de la siguiente manera:

TABLA 1. ESPECIALIZACIÓN DE TIPO DE EMPRESA. CUADRO 15 (FEDESOFTE, 2013, PÁG. 55)

Tamaño de Empresa	%
Grande empresa (más de 500 empleados - activos mayores a los \$28.335.000.000)	36%
Mediana empresa (entre 51 y 500 empleados - activos entre \$2.834.066.700 y \$28.335.000.00)	28%
Pequeña empresa (entre 11 y 50 empleados - activos entre \$283.916.700 y \$2.833.500.000)	20%
Microempresa (menos de 10 empleados - activos hasta \$283.350.000)	12%
Personas naturales	4%

Para el año 2009 según (CEPAL, 2009, pág. 153), las empresas estaban distribuidas en cuatro tipos de empresas bien definidas: “empresas desarrolladoras de software, empresas distribuidoras y comercializadoras de productos informáticos, empresas proveedoras de acceso y servicios de Internet y empresas productoras de hardware”.

De acuerdo a cifras más actualizadas (FEDESOFTE, 2013), la distribución de las empresas de acuerdo con la Clasificación Industrial Internacional Uniforme de todas las Actividades Económicas – CIIU-, corresponde a la siguiente:

TABLA 2. DISTRIBUCIÓN DE LAS EMPRESAS POR CÓDIGO CIIU. CUADRO 9 (FEDESOFTE, 2013, PÁG. 51)

CIIU	DESCRIPCIÓN CIIU	%
6201	Actividades de desarrollo de sistemas informáticos	36%
6202	Actividades de consultoría informática y de instalaciones informáticas	26%
6209	Otras actividades de tecnologías de información	3%
5820	Edición de programas de informática (software)	3%
4741	Comercio al por menor de computadores, equipos periféricos, programas de informática	3%
	Otros códigos CIIU	29%

Lo anterior, sumado al hecho de que la industria del software es una industria intensiva en conocimiento, que posibilita la modernización de los procesos productivos, propicia el uso de habilidades laborales sofisticadas y conduce a la producción de bienes con mayor valor agregado (Lopera Carmona, 2012); se infiere que si dicha industria desea continuar con su crecimiento y consolidación, es necesario caracterizarla para determinar los factores causantes de baja calidad en el software; refinarla a través de la adopción de buenas prácticas de ingeniería e implementación de estándares, modelos, metodologías y demás; especializarla a través de profesionales altamente calificados y enriquecerla a partir de una discusión fina que se dé con los requisitos como elemento fundamental en el aseguramiento de la calidad del producto software desde el inicio del proceso de desarrollo.

Al respecto, (Merchan, Urrea, & Rebollar, 2008) mencionan que el 58.3% de las 306 empresas emergentes de desarrollo de software del sur-occidente colombiano encuestadas no establecen criterios para la aceptación de proveedores de requisitos, lo que puede traer falencias para el desarrollo del proyecto por cuanto las fuentes de los requisitos no tienen dominio de la problemática del sistema ni de sus procesos. Esto también puede causar que se obtengan requisitos mal interpretados e inconsistentes, o incluso la omisión de requisitos que pueden ser necesarios para la solución.

Así mismo, indica el autor que un alto porcentaje de las empresas (48.71%) no establece criterios para la aceptación de requisitos, factor que puede ocasionar la presencia de errores en el desarrollo de los mismos que redunden en retrasos en los cronogramas de trabajo, trabajo repetitivo o hasta rechazo del cliente (Merchan, Urrea, & Rebollar, 2008).

También se sabe, según (Torres de Paz, 2009, pág. 5) que el 56% del total de errores encontrados en los productos software son generados en la fase de Requisitos, como lo muestra la *Ilustración 2*. Además, el porcentaje total del esfuerzo del proyecto que se destina en la fase de Requisitos no llega al 20% independientemente del modelo de desarrollo (iterativo, en cascada, etc.) o del tipo de proyecto (nuevo, de mantenimiento). (Yang, He, Li, Wang, & Barry, 2008).

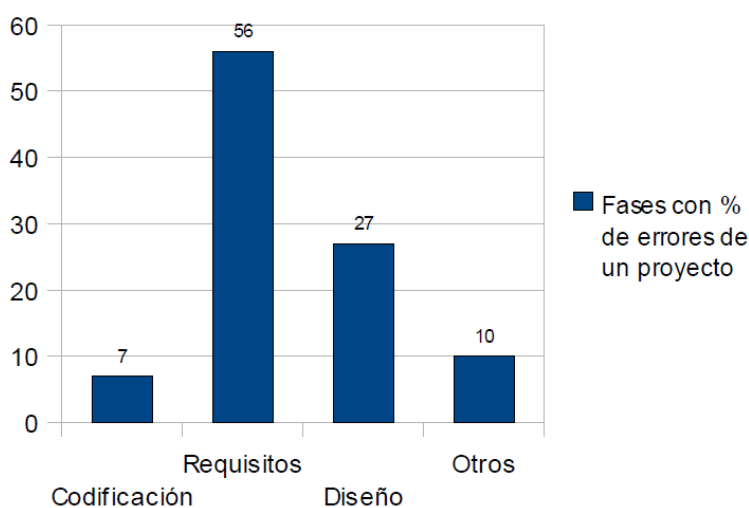


ILUSTRACIÓN 2. PROPORCIÓN DE ERRORES EN CADA UNA DE LA FASES DE DESARROLLO. (TORRES DE PAZ, 2009)

Del mismo modo, (Cristiá, 2011) en la *Ilustración 3*, relaciona las principales causas de fracaso en los proyectos:

Factores de Daño o cancelación	%
Requerimientos incompletos	13.1
Deficiencia en el involucramiento del usuario	12.4
Deficiencia de recursos	10.6
Expectativas no realistas	9.9
Deficiencia en soporte ejecutivo	9.3
Cambios en los requerimientos y especificaciones	8.7
Deficiencia en la planeación	8.1
Ya no se necesita más	7.5
Deficiencia en administración de TI	6.2
Desconocimiento en tecnología	4.3
Otros	9.9

ILUSTRACIÓN 3. FACTORES DE FALLA O CANCELACIÓN EN LOS PROYECTOS (CRISTIÁ, 2011)

Al ser la parte inicial de un proyecto de desarrollo de software, los requisitos tienden a ser la parte más incomprendida de la Ingeniería de Software y, sin embargo, es la más crucial. Gran parte de las fallas en los proyectos de software se deben a una pobre definición de requisitos, encontrándose dentro de las principales razones más comunes de fracaso de dichos proyectos.

Según (INTECO, 2008), el costo de una buena recogida de requisitos y el análisis del sistema a desarrollar es menor comparado con el costo resultante de tener requisitos pobres, es decir, el costo de reparar productos deficientes o de poca calidad.

Por lo anterior y de acuerdo al mismo autor, *“el fundamento básico de cualquier software recae sobre su proceso de ingeniería de requisitos. El éxito o fallo del software depende casi siempre de cómo de bien se hayan capturado, entendido y usado los requisitos como base para el desarrollo.”*

Así pues, al entender que la ingeniería de requisitos es la fase de un proyecto software donde se definen las propiedades del mismo; y que a la vez comprende el desarrollo y gestión de requisitos, se entiende también que debe realizarse de manera adecuada, pues omitir información en esta actividad ha provocado que muchos proyectos de software fracasen.

Adicionalmente, debido a que no se realiza un estudio previo de los requisitos del usuario, no se hace una definición completa del alcance del proyecto y no se realiza el modelado del negocio antes de desarrollar el software; el equipo desarrollador o analista no se involucran en el problema y aunque se tiene claro que el sistema debe desarrollarse para dar soporte a los procesos de la organización, si el equipo no se involucra en la problemática se corre el riesgo de que los requisitos identificados no correspondan a las necesidades para lo que se debe crear (Maita, 2012).

En el caso de Colombia, según estudios de la Asociación Colombiana de Ingenieros de Sistemas (ACIS) citado en (Varela Galvis & Arango Sterling, 2012), en la industria del software cerca del 70% de los proyectos no son exitosos, siendo uno de los principales problemas la ejecución de los procesos relacionados con la ingeniería de requisitos. Entre sus causas más

frecuentes esta la falta de comprensión de los requisitos iniciales y la inhabilidad para gestionar los cambios durante el desarrollo del mismo.

Además, es pertinente mencionar los resultados obtenidos por (De la Cruz Londoño & Castro Guevara, 2014) en su trabajo de investigación “*Metodología para la adquisición y gestión de requerimientos en el desarrollo de software para pequeñas y medianas empresas (PyMes) del Departamento de Risaralda*”, en el cual se identificaron los principales inconvenientes que presentan las pequeñas empresas de la región al trabajar con los requisitos, los cuales se relacionan a continuación:

- En ocasiones los clientes no conocen sus necesidades o presentan inconvenientes para expresarlas.
- En general falta conocimiento por parte del equipo de desarrollo del proyecto en el dominio del problema del cliente.
- No hacen uso de metodologías, sin embargo, realizan adecuaciones a plantillas ad-hoc para realizar el proceso de “*elicitación*”.
- Es complejo realizar la estimación de tiempos.
- En la mayoría de empresas no se identifica con claridad los *stakeholders* involucrados en el proyecto.
- En algunos casos no realizan identificación de requisitos no funcionales.
- En la mayoría de empresas no se realiza gestión de requisitos.
- Tan solo hacen uso de entrevistas, prototipos y plantillas ad-hoc como técnicas de “*elicitación*” de requisitos

2.2. Antecedentes

Como se pudo notar en apartados anteriores los requisitos de software representan una parte fundamental en la consecución de la calidad del software.

Sin embargo, se detecta -a manera de hipótesis- un problema específicamente en la industria del software relacionado con una deficiente especificación y validación de los requisitos, lo que conlleva al desarrollo de un producto software que no cumple con las características requeridas para satisfacer las necesidades de la industria. Se hace necesario entonces reconocer orígenes de ese problema a partir de unos antecedentes regionales, nacionales e internacionales, así como las iniciativas que se han propuesto para contribuir a su solución. A continuación, algunos trabajos que permiten dar cuenta de los logros que se han alcanzado en investigaciones similares y que tienen como objeto de estudio la especificación y validación de los requisitos de software:

2.2.1. Del orden regional

En la *Tabla 3* que se encuentra a continuación, se relaciona una descripción de algunas de las investigaciones sobre especificación de requisitos de software realizadas en la región y que sirven como referencia para este trabajo. En ella, se presentan los nombres de los autores, el título del trabajo o investigación, la universidad a la cual pertenecen, las técnicas de especificación y validación sugeridas o trabajadas por los autores, las principales falencias identificadas en el trabajo y, finalmente, el aporte específico al presente proyecto.

TABLA 3. ANTECEDENTES REGIONALES SOBRE ESPECIFICACIÓN Y VALIDACIÓN DE REQUISITOS DE SOFTWARE

AUTOR	TÍTULO	UNIVERSIDAD	TÉCNICAS ESPECIFICACIÓN	TÉCNICAS VALIDACIÓN	FALENCIAS DEL PROYECTO	APORTE ESPECÍFICO
(Correa Botero & González Bedoya, 2009)	Guía para implementar buenas prácticas en las áreas de procesos de gestión de requerimientos y planeación del proyecto para las microempresas desarrolladoras de software, basada en CMMI	Universidad Tecnológica de Pereira	<ul style="list-style-type: none"> • Documento de la definición del sistema • Especificación de requisitos del sistema • Especificación de requisitos del software • Diagramas de casos de uso • Plantillas y patrones 	<ul style="list-style-type: none"> • Prototipado • Modelo de validación • Tablas para validación de requisitos y evaluación de su impacto. • Walkthroughs o recorridos. 	<p>La guía ilustra a las empresas como se pueden llevar a cabo buenas prácticas en el área de Gestión de Requisitos, Planeación del Proyecto y Desarrollo de Requisitos.</p> <p>Adicionalmente, se presenta a las microempresas desarrolladoras de software como una herramienta sencilla para la medición de los procesos de Gestión de Requisitos y Planeación del Proyecto, pero no se validó la guía.</p>	<p>Técnicas para la especificación y validación de requisitos. Plantillas para especificación de requisitos.</p>
(De la Cruz Londoño & Castro Guevara, 2014)	Metodología para la adquisición y gestión de requerimientos en el desarrollo de software para pequeñas y medianas empresas (PYMES) del departamento de Risaralda	Universidad Tecnológica de Pereira	<ul style="list-style-type: none"> • Historias de usuario • Documento de Especificación de Requisitos de Software (SRS por sus siglas en inglés) • Especificación en lenguaje Z • Modelo de flujo de datos • Modelo de contexto • Modelo de objetos • Diagrama de 	<ul style="list-style-type: none"> • Prototipos no funcionales • (Walkthroughs o recordos) • Formatos y herramientas 	<p>El trabajo es netamente teórico, evidenciando la falta de una aplicación realmente práctica que corrobore lo expuesto teóricamente.</p> <p>La metodología requiere de personas expertas en el área de requisitos al momento de su aplicación.</p> <p>También, se debe contar con conocimiento específico en lenguajes</p>	<p>Indicar la manera en que las empresas de desarrollo de software de la región están llevando a cabo la adquisición y gestión de requisitos, como también entregando una metodología que permite el abordaje de los mismos.</p> <p>La metodología aporta además formatos, métodos y herramientas para llevar a cabo los procesos de</p>

			<p>secuencia</p> <ul style="list-style-type: none"> • Modelo con redes Petri • Prototipos (maquetas digitales, storyboard navegación) • Formatos y herramientas 		<p>formales para especificar requisitos de software.</p>	<p>especificación y validación.</p>
--	--	--	--	--	--	-------------------------------------

Otros trabajos relacionados con requisitos de software desarrollados por investigadores de la región se describen a continuación, obteniendo de ellos también los posibles aportes al presente proyecto.

La Universidad Tecnológica de Pereira ha mostrado interés en explorar a manera de investigación o de proyectos de desarrollo el área de requisitos. Es así como (Lema Becerra & Rodas Valencia, 2012) elaboran el análisis y diseño de una herramienta gráfica para los procesos de la ingeniería de requisitos que tiene como objetivo proporcionar un mecanismo visual de comunicación para identificar las necesidades que dan lugar al desarrollo de un proyecto de software, como también dar sencillez a la monitorización de la información que se genera en los procesos de la ingeniería de requisitos. El conocimiento y descripción de estos procesos, así como el tener una herramienta que facilita la identificación de los requisitos del usuario son aporte importante que el mencionado antecedente entrega al presente proyecto, pues complementa la adquisición de requisitos -una fase previa a la especificación- mediante la implementación por parte de un equipo de desarrollo, de la teoría de los sistemas blandos, para que, a través de la construcción de imágenes enriquecidas se minimicen los problemas relacionados a la especificación de requisitos de naturaleza cognitivos, antropológicos, sociales y lingüísticos en la comunicación, tales como aquellos relacionados con problemas de articulación, problemas de limitaciones cognitivas del ser humano, problemas de comunicación entre los interesados, analistas y equipo de desarrollo; problemas técnicos y problemas de estructuración de la información. Todo ello, mediante la implementación de la teoría de los mapas conceptuales para configurar el plan de administración de requisitos y también realizar la administración de los mismo durante un proyecto de software, obteniendo finalmente los documentos formales del plan de administración de requisitos y la especificación de requisitos a partir del mapa conceptual y las plantillas que se hayan construido.

El proyecto anterior aporta herramientas que podrían llegar a ser útiles para este trabajo, tales como las plantillas “Atributos de los requisitos”, “Tipos de requisitos” y “Documento especificación de requisitos”, las cuales ayudan a la generación de documentos. Sin embargo, el proyecto posee una seria dificultad: se desarrolló el análisis y diseño de un gestor gráfico de requisitos, pero no se implementó ni se validó con la industria real, lo cual no permite concluir respecto a su contribución al mejoramiento de la calidad del software.

En la misma línea, se han realizado diferentes investigaciones con el objetivo de encontrar mejores prácticas, herramientas y métodos que faciliten el trabajo con los requisitos. (Rangel Grisales & Peláez Betancur, 2013) por ejemplo, han mostrado preocupación en estudiar una parte específica de la ingeniería de requisitos concerniente a la recolección de los mismos, planteando como objetivo de su investigación la elaboración de un documento que analiza los métodos de obtención de requisitos tales como entrevistas, lluvias de ideas, grupos focales, cuestionarios, entre otros y sus enfoques de selección, con miras de minimizar problemas relacionados con la mala gestión de los requisitos en el desarrollo de proyectos de software; esto se debe a que la obtención de requisitos por ser una actividad que es llevada a cabo por personas, cada quien expone su punto de vista subjetivo haciendo que se haga más compleja la comunicación y el entendimiento por parte de los analistas, quienes más tarde usarán los detalles de la adquisición para especificar lo que el software debe satisfacer. La falencia principal de este proyecto es que, si bien logró comprender los métodos de obtención de requisitos, recopilar la información más relevante y presentarla de manera estructurada, no evidencia ningún proceso de aplicación o

validación que pudiera entregar aportes reales sobre su contribución en el aseguramiento de la calidad de los requisitos y del software mismo, convirtiéndose el proyecto en un análisis meramente teórico.

Los antecedentes anteriores, sin embargo, a pesar de las importantes contribuciones y aportes que pueden hacer al presente proyecto, poseen en términos generales una serie de falencias y desventajas que ameritan ser tenidas en cuenta para el desarrollo del nuevo procedimiento. Una de ellas, quizá la principal para la perspectiva de esta investigación, es la relacionada con la ausencia de aplicación de las diferentes metodologías, guías o técnicas propuestas en el sector real, pues es allí donde finalmente harán su contribución al mejoramiento de la calidad del software. Se evidencia entonces la existencia de aspectos a mejorar y la oportunidad que tiene el presente proyecto de retomarlos para contribuir con soluciones más cercanas a las empresas, que no se queden en la teoría y puedan ser realmente aplicadas en los procesos de ingeniería de requisitos.

2.2.2. Del orden nacional

Se determina de igual manera que en el ámbito nacional se han llevado a cabo investigaciones importantes en el tema de especificación y validación de requisitos de software. La *Tabla 4* muestra algunas de las investigaciones relacionadas con la especificación de requisitos de software desde el ámbito nacional.

TABLA 4. ANTECEDENTES NACIONALES SOBRE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

AUTOR	TÍTULO	UNIVERSIDAD	TÉCNICAS ESPECIFICACIÓN	TÉCNICAS VALIDACIÓN	FALENCIAS DEL PROYECTO	APORTE ESPECÍFICO
(Valencia, 1989)	Guía para la preparación de un documento de requerimiento	Universidad del Valle	<ul style="list-style-type: none"> • Diagrama jerárquico o de bloques • Diagrama de flujo de datos • Formato para informes 	<ul style="list-style-type: none"> • No refiere 	Se indican los aspectos básicos para el desarrollo del documento de especificación de requisitos, pero no se muestran ejemplos desarrollados ni se mencionan experiencias de aplicación. Tampoco está enfocado a un tamaño de empresa en particular.	Aspectos básicos fundamentales para que el documento de especificación de requisitos sea completo. Formatos para la especificación y validación de requisitos.
(Camacho Zambrano, 2005)	Herramienta para el análisis de requerimientos dentro de la pequeña empresa desarrolladora de software en Bogotá	Pontificia Universidad Javeriana	<ul style="list-style-type: none"> • No refiere 	<ul style="list-style-type: none"> • Prototipo funcional • Formato para análisis de riesgos 	<p>La herramienta fue desarrollada para el análisis de requisitos en pequeñas empresas desarrolladoras de la ciudad de Bogotá, de manera aislada para las fases de recolección y análisis de requisitos. Por esto como lo menciona el autor, existen interrogantes alrededor de su aplicabilidad y extensibilidad a las demás fases de la ingeniería de requisitos.</p> <p>Los requisitos no funcionales no son tenidos en cuenta.</p> <p>La herramienta no satisface todas las actividades del análisis de requisitos.</p>	Directrices y recomendaciones a la construcción de la nueva propuesta. Documentación de su aplicación mediante un caso de estudio en la empresa SWONE Ltda.

<p>(Merchan, Urrea, & Rebolgar, 2008)</p>	<p>Definición de una metodología ágil de ingeniería de requerimientos para empresas emergentes de desarrollo de software del suroccidente colombiano</p>	<p>Universidad de San Buenaventura, Cali</p>	<ul style="list-style-type: none"> • Historias de usuario • Casos de uso • Prototipos • Documento de definición de requisitos • Documento de priorización de requisitos 	<ul style="list-style-type: none"> • Prototipos 	<p>La metodología ágil de ingeniería de requisitos para las pequeñas empresas emergentes del Valle del Cauca se definió y documento para dos tipos de proyectos: desarrollo a la medida e ideas de negocio. Por lo cual, se asume que otros tipos de proyectos no contemplados por la metodología como por ejemplo software de aplicación o software de programación, podrían tener dificultades en su aplicación.</p> <p>No se evidencian resultados de aplicación real en las empresas, aunque se menciona que la metodología se encuentra en etapa de aplicación en empresas piloto.</p> <p>Se menciona que la herramienta no soporta en su totalidad la etapa de requisitos de software, por lo cual recomienda el desarrollo de otros trabajos que puedan llegar a complementarla.</p>	<p>La identificación de los factores problema en la ingeniería de requisitos, que será base para la definición del procedimiento. La recomendación de técnicas para identificación y especificación de requisitos de software.</p>
---	--	--	--	--	---	--

Los investigadores (Zapata J & Arango Isaza, 2004) resaltan la importancia de la alineación entre las metas organizacionales y la elicitación o captura de requisitos de software al mencionar que *“la elicitación de los requisitos del software debe contribuir a la obtención de una primera descripción del problema que permita establecer las motivaciones para su solución mediante sistemas informáticos”*. Proponen además que los métodos y herramientas convencionales para la elicitación de requisitos pueden dejar de lado aspectos fundamentales de la organización al enfocarse en situaciones relacionadas con problemas puntuales del entorno del software, razón por la cual sugieren la utilización de diagramas causa-efecto para registrar las ligaduras que deben existir entre los problemas o metas de la organización que de alguna manera justifican la aparición de una solución informática, para lograr un análisis cuantitativo del grado en que se logran alinear las metas organizacionales con los requisitos correspondientes a una pieza de software. Indudablemente este proyecto entregará al nuevo lineamiento bases importantes para el análisis de las metas organizacionales, reglas de negocio y requisitos de interesado o usuario con la *“elicitación”*, de tal manera que las directrices o lineamientos dados por el instrumento elaborado para la especificación, propicien la alineación mencionada. La propuesta es aplicada mediante un caso de estudio al software para la *“Gestión de préstamos y reservas de equipos y salones”* de la Universidad Nacional de Colombia.

En la misma línea, (Martínez Vaca & Ospina Arredondo, 2011) proponen un nuevo enfoque simplificado y fácil de implementar basado en herramientas de análisis causal mediante el uso de árboles jerárquicos que permita formular sistemáticamente las iniciativas de requisitos de software, que servirán como punto de partida para el proceso de adquisición de requisitos. De esta manera se podrán definir requisitos que realmente apoyen al negocio, evitando aquellos que van en contravía de los lineamientos organizacionales o que simplemente son innecesarios. Para el proyecto este antecedente puede ser importante para garantizar que los objetivos de negocio estén presentes en el documento de especificación logrado a través de la aplicación de la nueva propuesta, pues es fundamental que la misma no solo propicie el trabajo con requisitos funcionales y no funcionales, sino también los requisitos de negocio, de usuario y reglas de negocio. La propuesta no evidencia aplicación en casos reales.

En los trabajos del orden nacional, puede evidenciarse también una serie de falencias y puntos a mejorar que podrían considerarse en el nuevo procedimiento, algunos de ellos tienen que ver con el hecho de haber sido elaborados para una única fase del desarrollo de los requisitos o una parte específica de las mismas, dejando vacíos y aspectos incompletos en cada una de las propuestas producto de las diferentes investigaciones. Otra grande falencia, evidenciada también en el ámbito regional, es la falta de aplicación práctica de las propuestas en el sector real, aspecto fundamental para validar con las empresas los resultados de los proyectos de investigación analizados en los antecedentes. Finalmente, puede decirse que la mayoría de los trabajos estudiados indica la necesidad de desarrollar, a manera de trabajo futuro, herramientas, artefactos o investigaciones adicionales tendientes a complementar el trabajo realizado.

2.2.3. Del orden internacional

Respecto al ámbito internacional, se han encontrado investigaciones relevantes respecto al campo de los requisitos de software que aportan significativamente a la construcción de este trabajo, pues hacen alusión a aspectos importantes a tener en cuenta a la hora de determinar los factores de éxito y fracaso en el trabajo con los requisitos de software, tales como aquellos

relacionados con: la captura, análisis y especificación de requisitos, así como la identificación de los *stakeholders* o interesados, las técnicas y herramientas utilizadas en el proceso, así como la complejidad y efectividad de las mismas. En la *Tabla 5* que se presenta a continuación, se relacionan aquellos trabajos que desde el orden internacional aportan de manera específica con el presente trabajo.

TABLA 5. ANTECEDENTES INTERNACIONALES SOBRE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

AUTOR	TÍTULO	UNIVERSIDAD	TÉCNICAS ESPECIFICACIÓN	TÉCNICAS VALIDACIÓN	APORTE ESPECÍFICO
(Durán Toro & Bernárdez Jiménez, 2001)	Metodología para la elicitación de requisitos de sistemas software	Universidad de Sevilla (España)	<ul style="list-style-type: none"> • Modelo de casos de uso • Formatos y plantillas 	<ul style="list-style-type: none"> • No refiere 	Entrega una serie de actividades recomendadas, productos entregables y técnicas a emplear durante el proceso de elicitación de requisitos. Presenta un ejemplo de aplicación suponiendo el caso de la gestión de un pequeño video club, pero no indica el tamaño de empresa a la que está orientada.
(Báez & Barba Brunner, 2001)	Metodología DoRCU para la Ingeniería de Requerimientos	Instituto Superior Politécnico José Antonio Echeverría, La Habana (Cuba) y Universidad Autónoma de Entre Ríos (Argentina)	<ul style="list-style-type: none"> • Uso de Herramientas y Modelos formales o semi-formales (no especificados) 	<ul style="list-style-type: none"> • No refiere 	DoRCU es una metodología producto del estudio de diferentes enfoques, que se apoya en diversos métodos, técnicas y herramientas ampliamente utilizados y difundidos, lo cual la convierte en un sólido referente. Sin embargo, no especifica cuáles modelos y herramientas formales o semi-formales utilizar. No refiere a qué tamaño de empresa se puede aplicar ni se encontró evidencia de su aplicación.
(Sumano López, 2001)	Áncora: Metodología para el Análisis de Requerimientos de Software conducente al Reuso	Universidad Veracruzana (México)	<ul style="list-style-type: none"> • Guiones y Diálogos para la elaboración de casos de uso. • Modelo de Datos • Formatos 	<ul style="list-style-type: none"> • Técnica de Grupo Nominal (TGN). • Reunión de Reflexión y Diseño. • Prototipos 	Presenta modelos que pueden ser desarrollados en el análisis de requisitos conducente a obtener el documento de especificación. Entrega ejemplos de aplicación y puesta en práctica que serán fundamentales para

					<p>el entendimiento de la guía y para verificar su utilidad.</p> <p>Formatos para clasificación de requisitos.</p> <p>No refiere a qué tamaño de empresa se puede aplicar ni se encontró evidencia de su aplicación en casos reales.</p>
(Davyt Dávila, 2001)	Una guía para extraer, analizar, especificar y validar los requisitos de un proyecto	Universidad ORT (Uruguay)	<ul style="list-style-type: none"> • Prototipo • Modelo de clases conceptual • DCO (Plantilla Documento de Concepto de Operaciones) • Diagrama de actividad • ESRE (Documento de Especificación de requisitos) • Casos de uso 	<ul style="list-style-type: none"> • Prototipo 	<p>Orientar sobre la manera de llegar a una especificación de requisitos de calidad indicando con ejemplos las técnicas y herramientas que se pueden utilizar y el paso a paso de cómo utilizarlas durante el desarrollo de los requisitos.</p> <p>Ejemplos de uso de todas las técnicas en un proyecto aplicado.</p> <p>No refiere a qué tamaño de empresa se puede aplicar ni se encontró evidencia de su aplicación en casos reales.</p>
(Chikh & Alajmi, 2014)	Towards a dynamic software requirements specification	Universidad King Saud (Arabia Saudí)	<ul style="list-style-type: none"> • Representación XML del documento de SRS en el que está contenida la estructura lógica del documento SRS y la información básica. • Diagrama entidad-relación • Diagrama de casos de uso • Modelo de datos • DTD <i>language</i> (parte textual del modelado) 	No refiere	<p>Conjunto de problemas identificados y errores que generalmente se presentan en la especificación de requisitos de software.</p> <p>Conjunto de problemas que ocurren en la plantilla actual de Especificación de Requisitos o SRS (Software Requirements Specification) del estándar IEEE 830-1998.</p>

					<p>Modelo de documento dinámico para representar la especificación, en el que la mayor parte se alimenta automáticamente de una base de datos de requisitos.</p> <p>Se desarrolló un primer prototipo de un sistema de reservación de hotel para ilustrar cómo el modelo SRS propuesto permitirá a los ingenieros de requisitos ser más eficientes en la redacción del SRS. No refiere a qué tamaño de empresa se puede aplicar ni se encontró evidencia de su aplicación en casos reales.</p>
(Dorigan & De Barros, 2014)	A Process Model for Standardization and Increase in the Requirements Quality.	Universidade Estadual de Maringá y Universidade Estadual de Londrina (Brasil)	<ul style="list-style-type: none"> • Lenguaje natural detallado • Procesamiento del lenguaje natural (PLN) a través de software. 	<ul style="list-style-type: none"> • Análisis de escritura del lenguaje natural. 	<p>Presenta aspectos fundamentales a considerar para el nuevo procedimiento relacionados con la estandarización y el uso adecuado del lenguaje natural para la construcción del documento de especificación de requisitos.</p> <p>Presenta también una serie de trabajos relacionados con las técnicas para especificar y validar requisitos en lenguaje natural.</p> <p>No refiere a qué tamaño de empresa se puede aplicar ni se encontró evidencia de su aplicación en casos reales.</p>

(Carrizo Moreno, 2012) de la Universidad de Atacama en Chile, presenta una comparación de efectividad de las técnicas de análisis de requisitos software desde la visión que tienen los ingenieros de requisitos noveles acerca de la efectividad de las técnicas de educación y compararlas con la visión de expertos en requisitos. Esta comparación se realizó usando la técnica de emparrillado que permite conocer indirectamente la opinión de los sujetos respecto de las técnicas, en este caso: entrevista no estructurada, entrevista estructurada, observación de tareas habituales, clasificación de conceptos, cuestionario, análisis de protocolos, emparrillado, tormenta de ideas, método Delphi, grupos focales, *laddering* y casos de uso. El contexto en el que se realizó la comparación de las técnicas fue en un grupo de ingenieros de requisitos, noveles y expertos, representados por un grupo de estudiantes de master en ingeniería de software los primeros y por tres profesores encargados del módulo de requisitos del master en ingeniería de software con vasta experiencia tanto práctica como teórica en desarrollo de software y, en particular, en educación de requisitos los segundos. Además de los resultados obtenidos del estudio en el que se evidencia una sustancial diferencia entre ambas visiones, lo que facilita reconocer diferencias de efectividad de las técnicas y poder decidir con mayor certeza sobre las técnicas más adecuadas a utilizar; es importante destacar los hallazgos relacionados con el contexto del análisis de los requisitos, tales como la experiencia, la familiaridad con el dominio, la disponibilidad de tiempo y el interés del informante, entre otros, aportando al presente trabajo claridades respecto a que la calidad de una especificación de requisitos depende, en gran medida, de los hallazgos encontrados con relación al contexto de la educación.

Otro estudio relacionado con las prácticas usadas en la Ingeniería de Requisitos es el presentado por (Basharat, Fatima, Nisa, Hashim, & Khanum, 2013) en su artículo denominado *Requirements engineering practices in small and medium software companies: An empirical study*. En él los autores afirman que “la literatura reporta una variedad de prácticas de ingeniería de requisitos” y su estudio se centra en examinar dichas prácticas y determinar en qué medida estas prácticas se están siguiendo en la industria de software de la pequeña y mediana empresa de Pakistán. El estudio realizado mediante la aplicación de un cuestionario a 15 pequeñas, medianas y grandes empresas desarrolladoras de software también intenta identificar las limitaciones potenciales para adoptar prácticas de ingeniería de requisitos apropiadas y en presentar algunas soluciones posibles a las mismas. Del documento se pudo identificar que las empresas de software objetivo del estudio utilizan *Prototipos* y *Encuestas* como técnicas para identificar los requisitos, así como *Documentos de Requisitos* y *Tarjetas de Historias* para documentarlos, además de videoconferencia o grabación de vídeo del cliente. Para la validación se manifiesta usar *Casos de prueba automatizados* y los ejecutan antes de cada lanzamiento. En algunos lugares se utilizan *Casos de Uso* y *Prototipos* para la resolución de conflictos. No hay evidencia de aplicación de las soluciones posibles propuestas para contrarrestar las limitaciones identificadas al usar las prácticas relacionadas con la Ingeniería de Requisitos ni proponen trabajo futuro diferente a la aplicación de nuevas encuestas.

Ahondando un poco más en la temática de adquisición de requisitos, (Burnay, Jureta, & Faulkner, 2014) contribuyen en investigar de una manera empírica y teórica la adquisición de los requisitos y afirman que la dificultad al hacer las entrevistas es que los ingenieros y los interesados tienen requisitos diferentes, antecedentes y experiencias de los sistemas existentes como también expectativas del nuevo sistema. Durante las entrevistas cada quien tendrá diferentes supuestos sobre el medio ambiente, los requisitos y el futuro sistema. En sí mismo, no es un problema que las diferentes partes interesadas tengan diferentes supuestos. Sin embargo, se

convierte en un problema si algunos de esos supuestos permanecen implícitos en las entrevistas de adquisición. Además consideran que una parte en el tema de requisitos que no ha recibido la atención suficiente consiste en tratar de entender sobre qué categorías independientes de dominio de información los interesados tienden a hablar espontáneamente durante las entrevistas de obtención, y que otras categorías tienden a permanecer implícitas, es por ello que los autores proponen un Mapa de Tema de Elicitación (ETM) por sus siglas en inglés *Elicitation Topic Map*, que tiene como objetivo ayudar a los ingenieros de requisitos a preparar las entrevistas donde se muestra por medio de un diagrama los temas que deberían ser discutidos durante las entrevistas y también qué tanta probabilidad existe de que dichos temas sean discutidos de manera espontánea por los interesados. Para desarrollar una propuesta que incluya de manera explícita la gran mayoría de los temas relacionados con la especificación de requisitos de software, es fundamental conocer los entregables del proceso de adquisición, de tal manera que no se queden temas a tratar por fuera del documento logrado. Es allí donde recobra importancia el presente antecedente, pues presenta una técnica mediante la cual se pueden evidenciar las temáticas principales a ser discutidas entre los involucrados durante el proceso de requisitos, de tal manera que no se dejen por fuera temas importantes a ser definidos en el documento de especificación. La aplicación de la técnica incluyó entrevistas con un grupo de ingenieros profesionales en requisitos para identificar el conjunto de temas de una encuesta a ser aplicada a empresarios y estudiantes con el propósito de evaluar la manera en que cada grupo comparte información sobre los temas establecidos a cerca de las redes sociales en internet.

En Tailandia se identifica un estudio denominado “*A review of practice and problems in requirements engineering in small and medium software enterprises in Thailand*” llevado a cabo por (Khankaew & Riddle, 2014) de la School of Computing Science Newcastle University en Newcastle Upon Tyne, UK. En este se investiga el estado actual de los problemas de ingeniería de requisitos y la práctica entre las pequeñas y medianas empresas de software en Tailandia. El objetivo principal del estudio fue determinar las áreas a mejorar en los procesos de ingeniería de requisitos. Los datos fueron recogidos a través de entrevistas semiestructuradas con once pequeñas y medianas empresas (PYME). Los resultados muestran que las empresas de software en Tailandia encuentran problemas comunes como claridad, corrección, integridad, gestión del cambio y comunicación con el cliente. El resultado también muestra las necesidades de desarrollo en PYME tales como mejora de procesos de software, conocimiento de Ingeniería de Requisitos, herramientas de gestión de requisitos, capacitación y transferencia de conocimiento. Finalmente, se plantea como trabajo futuro llevar a cabo un estudio de caso industrial con los contactos de las PYMES tailandesas para identificar mejoras en la práctica de ingeniería de requisitos.

Por su parte (Quispe, Marquez, Silvestre, Ochoa, & Robbes, 2010) de la Universidad de Chile, en su trabajo de investigación denominado “*Requirements Engineering Practices in Very Small Software Enterprises: A Diagnostic Study*” presentan los resultados de un estudio de diagnóstico realizado en las pequeñas empresas de software de Chile, específicamente las ubicadas en la ciudad de Santiago. El estudio trata de identificar el estado de la práctica de la ingeniería de requisitos en este nicho de mercado y también las posibles limitaciones que tienen para adoptar prácticas de ingeniería de requisitos apropiadas. Los principales hallazgos del estudio se pueden resumir en los siguientes:

1. Las especificaciones del proyecto son por lo general conocidas, pero el cliente se encuentra a menudo con una solución insatisfactoria.

2. Lo anterior, lleva a la conclusión de que los problemas de comunicación con los clientes hacen que las especificaciones queden incompletas.
3. El alcance del proyecto se expande a medida que los clientes requieren cambios adicionales, a menudo con cambios inadecuados.
4. La Especificación de requisitos en pequeñas empresas de software es sobre todo un proceso *ad-hoc*.
5. Este proceso *ad hoc* conduce a problemas de gestión de requisitos tales como la pérdida de los mismos.
6. Cuando la incertidumbre surge, los desarrolladores tienden a resolver el problema sin hacer contacto con los clientes.
7. Las pequeñas empresas de software son conscientes de los beneficios de las prácticas de Ingeniería de Requisitos, pero no están seguras de que se aplican en su contexto.

Con el mismo sentido del trabajo anterior y dado que la unidad de análisis a la cual está enfocado este proyecto son las MiPymes, se hace importante mencionar como antecedente el estándar ISO/IEC 29110 también conocido como VSE (de la abreviatura en inglés de Very Small Entities). Según el trabajo presentado por (Abarca Romero, Arisaca Mamani, & Dávila Ramón, 2015) denominado “*Implementación del Perfil Básico de la ISO/IEC 29110 de una Pequeña Empresa Desarrolladora de Software: Lecciones Aprendidas*”, el mismo es un conjunto de documentos (estándares y reportes técnicos) que comenzaron a publicarse en el 2011 (Laporte, April, & Renault, 2006) y que propone un conjunto de perfiles de procesos (entrada, básico, intermedio y avanzado). La ISO/IEC 29110 es un modelo que se basa en la ISO/IEC 12207, la ISO/IEC 15289 y toma de MoProSoft aspectos fundamentales en su organización en detalle (Abarca Romero, Arisaca Mamani, & Dávila Ramón, 2015). A nivel de perfiles, se define uno denominado Perfil de Entrada que corresponde con lo esperado por una organización pequeña que inicia sus operaciones (ISO, 2012). Para el Perfil Básico se ha establecido dos procesos principales: Gestión de Proyecto e Implementación de Software (ISO, 2012).

La ISO/IEC 29110-5-1-2, es una guía de gestión e ingeniería en dónde se describe el proceso en detalle (actividades, roles, productos de entrada y salida) (ISO, 2012). Con dicho nivel de detalle se requiere pocos ajustes para su implementación, por lo que resulta más fácil de adoptar que otros modelos como ISO/IEC 12207 o CMMI.

Entre los principales beneficios esperados al aplicar la ISO/IEC 29110, según lo establecido en el propio estándar se tienen: (i) mejorar la calidad del software producido por la organización que adopta el modelo; (ii) elevar la capacidad de las organizaciones para ofrecer servicios con calidad y alcanzar niveles internacionales de competitividad; (iii) permitir la fácil adopción del modelo pues se brinda mayor detalle en la información; (iv) facilitar la migración de organizaciones que han implementado modelos como MoProSoft o ISO 9000; y (v) permitir el posicionamiento de las empresas que lo implementan destacando sobre sus competidores y en beneficio de sus clientes.

De acuerdo al análisis realizado de los antecedentes y explícitamente sobre la especificación de requisitos de software, se encuentra que de las 24 empresas que participaron de este estudio realizado en Arequipa (Perú), la mayoría no tienen una forma estructurada de hacer la especificación de requisitos, en su lugar adoptan un proceso *ad-hoc*, a pesar de la literatura

disponible sobre el tema, o la experiencia de otras empresas. Sienten además que los cursos y los libros no se aplican al contexto de las pequeñas empresas.

Respecto a la validación de requisitos de software, encuentran que el desarrollo comienza una vez que se validan los requisitos. Sin embargo, es común que las incertidumbres aparezcan en el momento en que los desarrolladores están construyendo la solución real, cuando los requisitos no fueron especificados lo suficientemente bien. Lo anterior, de acuerdo al autor, hace pensar que el proceso de validación de requisitos se llevó a cabo a toda prisa y que en general, estas pequeñas empresas no tienen un proceso de validación compatible con sus necesidades.

Una vez estudiados los antecedentes de los ámbitos regional, nacional e internacional, se establece como base principal para el proyecto el trabajo desarrollado por (De la Cruz Londoño & Castro Guevara, La Ingeniería de Requerimientos en las Pequeñas Empresas del Departamento de Risaralda, 2014), pues se considera que aporta de manera significativa al desarrollo del mismo, pues además de indicar la manera en que las empresas de desarrollo de software de la región están llevando a cabo la adquisición y gestión de requisitos, también entrega una metodología que permite el abordaje de los mismos incluyendo todas las etapas para el desarrollo y gestión de los requisitos, a través del establecimiento de las entradas y salidas de cada una de estas etapas. También aporta formatos para llevar a cabo los procesos de especificación y validación, pero no especifica en detalle las actividades y secuencia de pasos requeridos para realizarlas y lograr conseguir un documento de especificación de requisitos estructurado y validado por el usuario de acuerdo a lo que propone la metodología. A su vez la metodología no suministra una relación precisa entre las entradas y salidas de cada una de las etapas propuestas para el desarrollo de los requisitos.

Como falencia, se encuentra que para la validación de la metodología se usó como instrumento el juicio de expertos en donde se evidenció que la industria del software podrá disminuir los errores, el costo y el tiempo de desarrollo de sistemas de software al aplicar esta metodología. Sin embargo, la misma no se aplicó de manera real en empresas desarrolladoras de software de la región.

Así mismo, la metodología requiere de personas expertas en el área de requisitos al momento de su aplicación, pues se presenta de manera muy técnica y se requiere de conocimientos específicos en lenguajes formales tales como el Lenguaje Z para especificar requisitos de software en sistemas complejos, situación desfavorable para las MiPymes desarrolladoras pues las obliga a contar con este tipo de profesionales para aplicar de manera adecuada la metodología.

Los demás proyectos de investigación apoyarán de manera indirecta al procedimiento para la especificación y validación de requisitos de software, pues trabajan temas relacionados con la captura y análisis de requisitos, aportando buenas prácticas para el proceso inicial del desarrollo de los requisitos, de tal manera que las entradas requeridas para las fases de especificación y validación sean las adecuadas para el procedimiento elaborado en el presente proyecto.

Finalmente, en el *ANEXO I - Artículo publicado Entre Ciencia e Ingeniería No. 19* elaborado por los autores del presente proyecto y publicado en la Revista Entre Ciencia e Ingeniería, se presenta a manera de resumen una revisión teórica de los antecedentes relacionados

con la especificación y validación de requisitos de software realizada como parte del proyecto de investigación y presentada anteriormente.

2.3. Problema de investigación

De acuerdo al referente contextual y a una de las principales problemáticas de la industria del software que es la gestión de requisitos, surge un cuestionamiento que será el punto de partida para conseguir el resultado que se pretende con la investigación. Así pues, resulta importante indagar al respecto del siguiente interrogante y tratar de darle respuesta mediante el proceso de investigación:

¿De qué forma se debe llevar a cabo la especificación y validación de requisitos en un proyecto software en MiPymes desarrolladoras de software de la ciudad de Pereira?

3. JUSTIFICACIÓN

Para hacer frente a las causas de fracaso relacionadas en el apartado **2.1. Área Problemática**, generalmente se aplican procesos de mejoramiento de desarrollo de software e iniciativas aceptadas internacionalmente, por ejemplo: CMMI, SPICE y PSP/TSP. Sin embargo, la implementación de estas iniciativas en micro y pequeñas empresas implica mayor complejidad, mayor cantidad de prácticas, rigidez y costos, como lo expresan (Casallas, 2007) y (Arboleda & Casallas, 2004), y son ajenas a condiciones y/o características propias de la organización como lo indica (Merchan, Urrea, & Rebollar, 2008) y lo confirma (Laporte C. , 2010). *“No existe un instrumento que sugiera un proceso de Desarrollo de Software con base en características de la organización y en buenas prácticas, sobre el cual la organización pueda iniciar su proceso de mejoramiento”* (Varela Galvis & Arango Sterling, 2012).

De acuerdo a las fuentes antes citadas, existen grandes dificultades respecto a la calidad del software relacionadas con las actividades que se llevan a cabo durante el desarrollo de los requisitos, pues como lo afirma (IEEE, 2014, pág. 32) *“está extensamente reconocido dentro de la industria del software que los proyectos de la ingeniería de software son críticamente vulnerables cuando estas actividades se realizan mal”*. Tal razón soporta el realizar la presente investigación, de tal manera que los resultados de la misma contribuyan al mejoramiento de la calidad del software a través del proceso de desarrollo de requisitos, teniendo en cuenta el impacto que ellos generan en el éxito de los proyectos software. Así mismo, ayudará a elaborar un procedimiento que permita llevar a cabo la especificación y validación de requisitos en un proyecto software a cargo de las MiPymes desarrolladoras de software de la ciudad de Pereira.

El diseño del procedimiento que se propone tendrá aportes diferenciadores tales como el no requerir de un ingeniero de requisitos al momento de su aplicación ni tampoco la necesidad de contar con conocimiento específico en lenguajes formales para especificar requisitos de software, como sucede el proyecto de (De la Cruz Londoño & Castro Guevara, 2014). Esto, porque el procedimiento suministrará en detalle las actividades que lo comprenden, con la ayuda de guías explícitas para llevarlas a cabo y con la aplicación de dos pruebas piloto con dos unidades de análisis para que las MiPymes desarrolladoras de software lo puedan apropiar. De esta manera no solo se da continuidad a proyectos de investigación desarrollados en la región, sino que se actualizan y se validan, contribuyendo así a la difusión y apropiación de este conocimiento por parte del sector empresarial.

Finalmente, es indispensable aclarar que este proyecto tiene dos elementos importantes que requieren de más de un maestrante para llevarlos a cabo: el primero, tiene que ver con la caracterización de las empresas que comprende el trabajo de campo, con el objetivo de abarcar una población mayor; el segundo, con la validación del procedimiento elaborado teniendo en cuenta un proyecto base por cada estudiante, (proyectos desarrollados sin aplicar el procedimiento propuesto) y la aplicación del procedimiento en dos proyectos nuevos (uno por cada estudiante).

4. OBJETIVOS

4.1. Objetivo General

Elaborar un procedimiento para especificar y validar requisitos de software basado en estudios previos en la región, que pueda ser utilizado por las MiPymes desarrolladoras de software de la ciudad de Pereira.

4.2. Objetivos Específicos

- Caracterizar la manera como las MiPymes desarrolladoras de software de la ciudad de Pereira están llevando a cabo la especificación y validación de requisitos.
- Identificar los elementos de proceso (actividades, técnicas, roles, guías, entre otros) requeridos para el procedimiento de especificación y validación de requisitos de software, a partir de los antecedentes, marco teórico y MiPymes desarrolladoras de software.
- Diseñar la propuesta inicial del procedimiento para la especificación y validación de requisitos de software en las MiPymes desarrolladoras de software de la ciudad de Pereira, a partir de las técnicas y recomendaciones analizadas.
- Validar mediante una prueba piloto el procedimiento propuesto en MiPymes desarrolladoras de software de la ciudad de Pereira.

5. REFERENTE TEÓRICO

5.1. Requisito Vs. Requerimiento

Existen diferentes conceptos relacionados con el objeto de estudio de este trabajo, uno de ellos es el término *requisito*. El Instituto de Ingenieros Eléctricos y Electrónicos (IEEE), en su estándar 610.12-1990 lo define como “*una condición o necesidad de un usuario para resolver un problema o alcanzar un objetivo*” (IEEE, 1993, pág. 62). De igual forma, menciona que es “*una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal*” (IEEE, 1993).

El término *requisito* es definido por la guía al Cuerpo de Conocimiento de la Ingeniería del Software (SWEBOK) por sus siglas en inglés, como “*una propiedad que debe exhibir el software para resolver algún problema del mundo real*”. (IEEE, 2004, pág. 24). De igual forma, en su versión 2014 indica que “*los requisitos de software expresan las necesidades y limitaciones impuestas a un producto de software que contribuyen a la solución de algún problema en el mundo real*” (IEEE, 2014, pág. 32).

Para (Sommerville, 2005, pág. 108),

“Los requisitos para un sistema son la descripción de los servicios proporcionados por el sistema y sus restricciones operativas. Estos requisitos reflejan las necesidades de los clientes de un sistema que ayude a resolver algún problema como el control de un dispositivo, hacer un pedido o encontrar información”.

En la misma línea el autor afirma que “*un requisito es simplemente una declaración abstracta de alto nivel de un servicio que debe proporcionar el sistema o una restricción de éste. En el otro extremo, es una definición detallada y formal de una función del sistema*” (Ídem).

El termino *requerimiento*, por su parte se refiere al “*acto judicial por el que se intima que se haga o se deje de ejecutar algo (petición)*” (RAE, 2014).

Según la RAE, *requerimiento* es un sinónimo de *necesidad*, es decir, un concepto orientado hacia la *carencia* o *falta* de algo. *Requisito* es, en cambio, *una circunstancia o condición necesaria para algo*.

Así pues, por ser la especificación y validación de requisitos de software el tema central de este proyecto, entender la diferencia entre requisito y requerimiento es de vital importancia, ya que en muchas ocasiones se hace uso indistinto de ambos términos. El origen de este problema es debido a la mala traducción del idioma inglés ya que la palabra *requirement* (requisito en inglés) puede fácilmente ser mal traducida como requerimiento (*request* en inglés).

Analizando el uso de estos términos en otros idiomas, se nota que los mismos son ajenos a esta situación, como se muestra en la siguiente ilustración:

Español	requerimiento	requisito
Inglés	request	requirement
Alemán	beantragen	Anforderung
Italiano	richiedere	requisiti
Francés	demander	exigence
Latín	requirere	requisitus

ILUSTRACIÓN 4. REQUISITO VS REQUERIMIENTO (PAGANI, 2013)

Por tanto, para efectos de tener claridad en este documento, requerimiento serán todas las necesidades y deseos pedidos por el cliente y las personas involucradas en el software y requisito todas las funcionalidades, características y restricciones que debería tener el software.

5.2. Ingeniería de Requisitos

La *Ingeniería de Requisitos* es tratada como una disciplina según (Boehm, 1981), y tiene como propósito “*desarrollar una especificación de requisitos completa, consistente y no ambigua, la cual servirá como base para acuerdos comunes entre todas las partes involucradas y en dónde se describen las funciones que realizará el sistema*”. El procedimiento propuesto para la especificación y validación de requisitos de software buscar satisfacer las características antes mencionadas, alineándose con el propósito de la Ingeniería de Requisitos, según lo expuesto por el autor.

Para (Sommerville, 2005, pág. 108) “*el proceso de descubrir, analizar, documentar y verificar estos servicios y restricciones se denomina ingeniería de requisitos (RE)*”. También menciona que “*la Ingeniería de Requisitos es el proceso de desarrollar una especificación de software. Las especificaciones pretenden comunicar las necesidades del sistema del cliente a los desarrolladores del sistema*”. (Ibídum, Pág. 82). Es evidente entonces la relevancia que tiene la especificación de requisitos dentro de la Ingeniería de requisitos, pues permite que el desarrollador entienda y de solución a las necesidades expresadas por el cliente.

(Pressman, 2006, pág. 115), por su parte, indica que la Ingeniería de Requisitos:

“Ayuda a los ingenieros de software a entender mejor el problema en cuya solución trabajarán. Incluye el conjunto de tareas que conducen a comprender cuál será el impacto del software sobre el negocio, qué es lo que el cliente quiere y cómo interactuarán los usuarios finales con el software.”

Así mismo, (Pressman, 2010, pág. 102) manifiesta que “*el espectro amplio de tareas y técnicas que llevan a entender los requisitos se denomina Ingeniería de Requisitos.*”

En la misma línea (Thayer & Dorfman, 1997) citados en (Pressman, 2010, pág. 102) definen que:

“La ingeniería de requisitos proporciona el mecanismo apropiado para entender lo que desea el cliente, analizar las necesidades, evaluar la factibilidad, negociar una solución razonable, especificar la solución sin ambigüedades, validar la especificación y administrar los requisitos a medida que se transforman en un sistema funcional.”.

Esta preocupación que expresa el autor por entender lo que el cliente realmente necesita y determinar en qué medida el software desarrollado puede contribuir al cumplimiento de los objetivos del negocio, deberían ser aspectos a tener en cuenta en el desarrollo de todo proyecto software y, por ende, en el documento de definición de requisitos.

Por su parte (Oberg, Probasco, & Ericsson, 2003) indican que la Ingeniería de requisitos es un enfoque sistémico para recolectar, organizar y documentar los requisitos del sistema; es también el proceso que establece y mantiene acuerdos sobre los cambios de requisitos, entre los clientes y el equipo del proyecto. De la misma manera, (Thayer & Dorfam, 2000, pág. 1) mencionan que “*la Ingeniería de Requisitos es la ciencia y disciplina a la cual le concierne el establecer y documentar los requisitos*”.

También (Boehm, 1981), menciona una disciplina denominada ingeniería de requisitos para desarrollar una especificación completa, consistente y no ambigua, la cual servirá como base para acuerdos comunes entre todas las partes involucradas y en dónde se describen las funciones que realizará el sistema. Así mismo, autores como (Wieggers & Beaty, 2013, pág. 15) consideran que todo lo concerniente a requisitos hace parte de un dominio entero definido Ingeniería de Requisitos, que a la vez es dividida tanto en Desarrollo como en Gestión de Requisitos. Como disciplina, establece el proceso de definición de requisitos en una sucesión de actividades mediante las cuales lo que debe hacerse se elicit, se modela y se analiza (Choque, 2001) citado en (Merchan, Urrea, & Rebolgar, 2008).

Por otro lado, la guía al cuerpo de conocimientos para la ingeniería del software SWEBOK® (Software Engineering Body of Knowledge) propuesto por la IEEE, en su versión 2014 trata el tema como la primera de sus áreas del conocimiento (KA por sus siglas en inglés *Knowledge Areas*), en la que se refiere a la captura, el análisis, la especificación y la validación de los requisitos del software y contempla una serie de aspectos y conceptos que llevan al software a ser objeto de aplicación de la ingeniería. Como producto de salida al proceso correspondiente de aplicar los conocimientos del área se logra un documento que permita sistematizar, revisar, evaluar y aprobar todo lo relacionado con los requisitos del software (Peláez Valencia, 2010).

De igual forma, el SEI⁴ a través del Modelo de Madurez de Capacidad Integrado CMMI® (Capability Maturity Model Integration) el cual contiene un conjunto de buenas prácticas de la industria para el desarrollo, mantenimiento, adquisición y operación de productos y servicios; contempla el trabajo con los requisitos donde se incluye tanto el desarrollo como la gestión de los mismos. Para ello se cuenta con las áreas de proceso “*Gestión de Requisitos*” o *Requirements Management (REQM)* por sus siglas en inglés y “*Desarrollo de Requisitos*” o *Requirements Development (RD)* ubicadas en los niveles 2 y 3 en la representación por etapas, estando ubicadas en la categoría de Proceso de Ingeniería para la representación continua y escalonada. Tienen como propósito producir, analizar y administrar los requisitos del cliente, del usuario, del

⁴ Software Engineering Institute (SEI) | Carnegie Mellon University

producto y de los componentes del producto. Las prácticas definidas en esta área de proceso permiten determinar todos los requisitos del proyecto, ya sea para el desarrollo o mantenimiento. Parte de los requisitos del cliente que son derivados en requisitos del producto hasta refinarlos al nivel de requisitos de los componentes del producto, todo esto durante el ciclo de vida del producto.

Si bien se puede considerar que la especificación de requisitos puede hacer parte de la documentación de un proyecto de software, es importante resaltar, como lo indican los autores, que dicho documento no solo permitirá plasmar las necesidades del cliente respecto al sistema a desarrollar, sino que también posibilitará llevar a cabo una adecuada gestión de los mismos.

5.2.1. Estructura de la Ingeniería de Requisitos

Algunos autores como (Wieggers & Beaty, 2013, pág. 15), (IEEE, 2014) proponen que la Ingeniería de Requisitos puede estructurarse como se ilustra a continuación:



ILUSTRACIÓN 5. ESTRUCTURA DE LA INGENIERÍA DE REQUISITOS (WIEGERS & BEATY, 2013, PÁG. 15)

Teniendo en cuenta que el desarrollo de requisitos como se muestra en la *Ilustración 6* comprende diferentes fases, el autor propone abordar todo el desarrollo de requisitos como un proceso iterativo con el objetivo de adquirir una mayor comprensión y expresión de los requisitos. En la siguiente ilustración se da cuenta de ello:

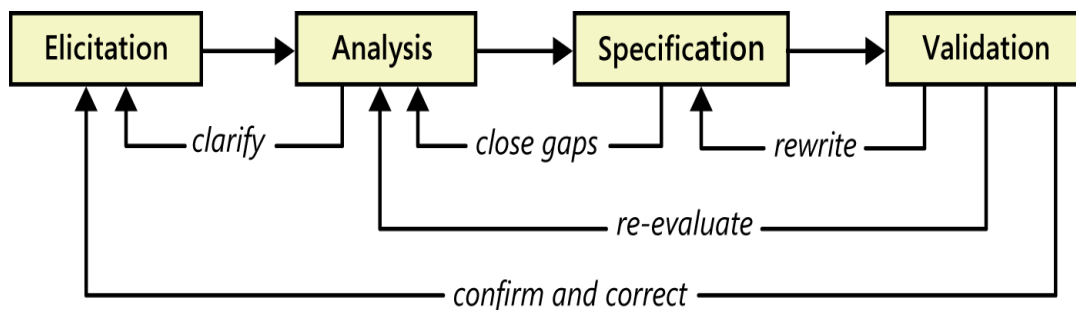


ILUSTRACIÓN 6. PROCESO ITERATIVO DE DESARROLLO DE REQUISITOS (WIEGERS & BEATY, 2013, PÁG. 45)

Cada una de estas fases de la ingeniería de requisitos, si se toma como referencia lo propuesto por el (IEEE, 2014) con respecto al desarrollo de requisitos, se explican con mayor detalle a continuación.

“Elicitación”: La elicitación o captura de requisitos hace referencia a los orígenes de los requisitos y como el ingeniero de software puede recogerlos. Es la primera etapa en la construcción de una comprensión del problema que el software requerido va a resolver. Es fundamentalmente una actividad humana y es donde los interesados se identifican y se establecen las relaciones entre el equipo de desarrollo y el cliente. Se denomina diversamente "Captura de Requisitos", "Descubrimiento Requisitos" y "Adquisición de Requisitos" (Idem, pág. 36).

Por su parte (Wieggers & Beaty, 2013, pág. 119) afirman que la elicitación de requisitos es “el proceso de identificación de las necesidades y limitaciones de los interesados para un sistema de software. “*Elicitación*” no es lo mismo que ‘el levamiento de requisitos.’ Tampoco es una simple cuestión de transcribir exactamente lo que dicen los usuarios. La “*elicitación*” es un proceso colaborativo y analítico que incluye actividades para recolectar, descubrir, extraer y definir los requisitos”.

Análisis: Según (IEEE, 2014, pág. 38), el análisis se refiere al proceso de analizar requisitos para:

- Detectar y resolver los conflictos entre los requisitos.
- Descubrir los límites del software y cómo debe obrar recíprocamente con su ambiente.
- Elaborar los requisitos del sistema para derivar requisitos software

Comprende las siguientes actividades:

- Clasificación de los requisitos
- Modelamiento conceptual.
- Diseño de arquitectura y asignación de requisitos.
- Negociación de Requisitos.
- Análisis formal.

Desde el punto de vista de (Pressman, 2002, pág. 212) en el análisis los requisitos se agrupan por categorías y se organizan en subconjuntos, se estudia cada requisito en relación con el resto, se examinan los requisitos en su consistencia, completitud y ambigüedad, y se clasifican en base a las necesidades de los clientes/usuarios.

Tanto la “*elicitación*” como el análisis de requisitos comprenden actividades fundamentales para identificar las necesidades del cliente y entender muy bien el software que se debe desarrollar para satisfacer dichas necesidades. Ambas son indispensables para elaborar una correcta definición de los requisitos y por ellos es importante mencionarlas en este documento.

Especificación: según la (IEEE, 2014, pág. 41),

“Para la mayoría de las carreras de ingeniería, el término ‘*especificación*’ se refiere a la asignación numérica de valores o límites a los objetivos de diseño de un producto. La

“Especificación de requisitos de Software se refiere típicamente a la producción de un documento, o a su equivalente electrónico, que puede estar sistemáticamente repasado, evaluado, y aprobado”.

Es notorio que la especificación, según el proceso iterativo propuesto por (Wiegers & Beaty, 2013) comienza cuando se ha realizado un trabajo en las etapas de “*elicitación*” y análisis. Es por ello que la especificación consolida los resultados de las etapas anteriores donde se hace fundamental que dicha consolidación sea lo suficientemente clara para convertir las necesidades del cliente en una solución software que las satisfaga.

El documento resultante en esta etapa conjuga entonces todos los aspectos necesarios, que relacionados entre sí proveen de forma transparente y clara las necesidades del cliente, como también la información necesaria para que el desarrollador del sistema entienda de forma correcta y exacta qué es lo que se debe realizar. Además, la especificación de requisitos es la fuente de lectura primaria donde todos los involucrados en el proyecto pueden conocer el alcance del mismo, en síntesis, es el acuerdo formal entre el cliente y el equipo desarrollador del sistema donde las necesidades del usuario son entendidas y donde se plantea una posible solución. Por tal razón un documento de especificación de requisitos influye en gran medida en la calidad del producto esperado.

Validación: (IEEE, 2014, pág. 42) considera la validación como posterior ejercicio a la especificación de requisitos donde el documento de requisitos de software resultante de una especificación debería ser sujeto de procedimientos de validación y verificación. Los requisitos deberían ser validados para asegurar de que el ingeniero software ha entendido los requisitos, es también importante verificar que un documento de requisitos se ajusta a las normas de la empresa y que es comprensible, coherente y completo. En casos donde la documentación de los estándares de la compañía o terminología no es compatible con los estándares ampliamente aceptados se debería realizar un mapeo o una correspondencia que será acordada y anexada al documento de requisitos.

Desde la mirada de (Wiegers & Beaty, 2013, pág. 331), los requisitos pueden ser sometidos a procesos tanto de verificación como de validación. El término “*verificación*” determina si se han escrito los requisitos bien, es decir, que cada requisito cumple con las características de un buen requisito. La “*validación*” evalúa si se han escrito los requisitos adecuados: están alineados a los objetivos de negocio. Estos dos conceptos están estrechamente entrelazados.

La validación de requisitos permite a los equipos construir una solución correcta que cumpla con los objetivos de negocio establecidos. Las actividades de validación requisitos tratan de asegurar que:

- Los requisitos de software describen con precisión las capacidades y propiedades previstas de los sistemas que van a satisfacer las necesidades de los interesados.
- Los requisitos de software correctamente se derivan de los requisitos del negocio, requisitos del sistema, reglas de negocio, y otras fuentes.
- Los requisitos están completos, factibles y verificables.
- Todos los requisitos son necesarios, y todo el conjunto es suficiente para cumplir con los objetivos de negocio.

- Todas las representaciones de los requisitos son coherentes entre sí.
- Los requisitos proporcionan una base adecuada para proceder con el diseño y la construcción.

De forma similar (Pressman, 2010, pág. 105) considera que:

“La calidad de los productos del trabajo que se generan como consecuencia de la ingeniería de los requisitos se evalúa durante el paso de validación. La validación de los requisitos analiza la especificación a fin de garantizar que todos ellos han sido enunciados sin ambigüedades; que se detectaron y corrigieron las inconsistencias, las omisiones y los errores, y que los productos del trabajo se presentan conforme a los estándares establecidos para el proceso, el proyecto y el producto”.

Para el interés particular del proyecto, se comprenderá entonces la validación como la etapa final dentro del desarrollo de requisitos, en ella los equipos involucrados en el proyecto se aseguran de que lo redactado en el documento de especificación de requisitos tenga la conformidad necesaria para continuar con los pasos siguientes, además en esta etapa se identifica si hay que realizar una retroalimentación desde el inicio del proceso de requisitos o en alguna etapa en particular, buscando una total certeza de que la plasmada en el papel es realmente la solución que el cliente necesita.

Como complemento al proceso de desarrollo de requisitos presentado anteriormente, resulta pertinente mencionar que existe una gran variedad de técnicas y herramientas utilizadas en las diferentes fases de dicho proceso, como se puede notar a manera de resumen en la *Tabla No. 6*. Aquellas relacionadas con la especificación de requisitos fueron ya presentadas en la sección 2.2 Antecedentes:

TABLA 6. HERRAMIENTAS QUE SE UTILIZAN EN LA INGENIERÍA DE REQUISITOS (DÁVILA, 2001, PÁG. 13)

Herramientas	Extracción	Análisis	Especificación	Validación
Entrevistas y cuestionarios	X			
Sistemas existentes	X	X		
Grabaciones de video y de audio	X	X		
Brainstorming (lluvia de ideas)	X	X		
Arqueología de Documentos	X	X		
Observación	X			
Prototipo Throw Away (no funcional)	X	X	X	
Prototipo Evolutionary (funcional)	X		X	X
Análisis DOFA		X		
Cadena de Valor		X		
Modelo Conceptual		X	X	
Diagrama de Pescado	X	X	X	
Glosario	X	X	X	X
Diagrama de Actividad		X	X	
Casos de uso	X	X	X	X
Casa de Calidad o QFD				X
Checklist	X		X	

5.3. Comité Internacional de Ingeniería de Requisitos

En el año 2006 se crea el Comité Internacional de Ingeniería de Requisitos (IREB por sus siglas en inglés: *International Requirements Engineering Board*), el cual busca “crear una base internacionalmente aceptada y profesional para la Ingeniería de Requisitos” (IREB, 2015). Este comité, conformado por varios expertos en el área, ha definido un glosario básico y un conjunto de certificaciones para profesionales en Ingeniería de Requisitos, que se han vuelto referentes importantes en la industria del software.

Algunos de los conceptos y lineamientos definidos por este comité fueron tenidos en cuenta para el desarrollo de este proyecto, tal como se menciona en la sección **7.3 Diseño del procedimiento para la Especificación y Validación de requisitos de software**, dada su importancia como referente para la disciplina de Ingeniería de Requisitos.

5.4. Clasificación de los Requisitos

Durante un proceso de desarrollo de requisitos se identifican generalmente diferentes tipos de ellos, algunos se relacionan a continuación de acuerdo a (Sommerville, 2005).

5.4.1. Requisitos de usuario

Los Requisitos de usuario “*son declaraciones, en lenguaje natural y en diagramas, de los servicios que se espera que el sistema proporcione y de las restricciones bajo las cuales debe funcionar*” (Sommerville, 2005, pág. 108).

5.4.2. Requisitos del sistema

Los Requisitos del sistema son aquellos que:

“Establecen con detalle las funciones, servicios y restricciones operativas del sistema. El documento de requisitos del sistema (algunas veces denominado especificación funcional) debe ser preciso. Debe definir exactamente qué es lo que se va a implementar. Puede ser parte del contrato entre el comprador del sistema y los desarrolladores del software” (Sommerville, 2005, pág. 108).

A su vez “*los requisitos de sistemas software a menudo se clasifican en funcionales y no funcionales, o como requisitos del dominio*”, como lo expresa (Ibídem, Pág. 109).

5.4.2.1. Requisitos funcionales

Los Requisitos funcionales, según el mismo autor:

“Son declaraciones de los servicios que debe proporcionar el sistema, de la manera en que éste debe reaccionar a entradas particulares y de cómo se debe comportar en situaciones particulares. En algunos casos, los requisitos funcionales de los sistemas también pueden declarar explícitamente lo que el sistema no debe hacer”.

De igual forma, “los requisitos funcionales describen las funciones que el software va a ejecutar; por ejemplo, ajustarse a un formato de texto o modular una señal. Se conocen también como capacidades” (IEEE, 2014, pág. 34).

5.4.2.2. Requisitos no funcionales

Los Requisitos no funcionales de acuerdo a (IEEE, 2014, pág. 34),

“Son los que actúan para obligar la solución. Los requisitos no funcionales se conocen a veces como apremios o requisitos de calidad. Pueden ser clasificados más a fondo según si son requisitos de funcionamiento, requisitos de capacidad de mantenimiento, requisitos de seguridad, requisitos de confiabilidad, o uno de muchos otros tipos de requisitos del software”.

Para (Sommerville, 2005),

“Son restricciones de los servicios o funciones ofrecidos por el sistema. Incluyen restricciones de tiempo, sobre el proceso de desarrollo y estándares. Los requisitos no funcionales a menudo se aplican al sistema en su totalidad. Normalmente apenas se aplican a características o servicios individuales del sistema” (Pág. 109).

Como se muestra en la *Ilustración 7*, los requisitos no funcionales se clasifican a su vez de la siguiente manera:

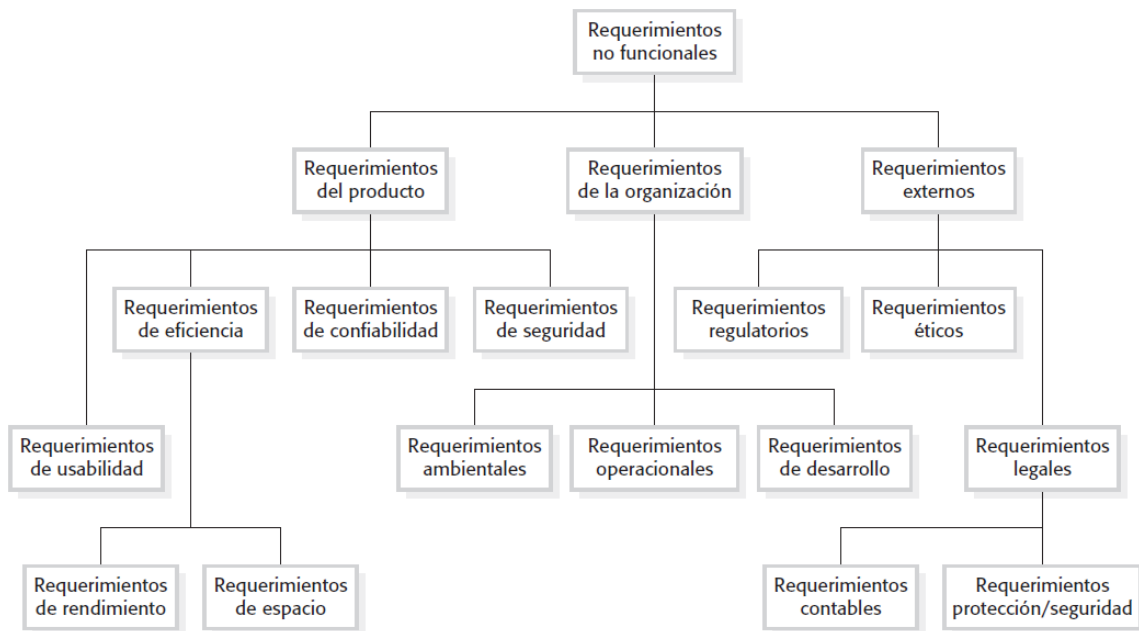


ILUSTRACIÓN 7. TIPOS DE REQUISITOS NO FUNCIONALES (SOMMERVILLE, 2011, PÁG. 88)

Los **requisitos del producto** especifican el comportamiento del producto. Algunos ejemplos son los requisitos de rendimiento en la rapidez de ejecución del sistema y cuánta memoria se requiere; los requisitos de fiabilidad que fijan la tasa de fallos para que el sistema sea aceptable;

los requisitos de portabilidad, y los requisitos de usabilidad.

Por su parte los **requisitos organizacionales** se derivan de políticas y procedimientos existentes en la organización del cliente y en la del desarrollador. Algunos ejemplos son los estándares en los procesos que deben utilizarse; los requisitos de implementación, como los lenguajes de programación o el método de diseño a utilizar, y los requisitos de entrega que especifican cuándo se entregará el producto y su documentación.

Los **requisitos externos** incluyen todos los requisitos que se derivan de los factores externos al sistema y de su proceso de desarrollo. Éstos pueden incluir los requisitos de interoperabilidad que definen la manera en que el sistema interactúa con sistemas de otras organizaciones; los requisitos legislativos que deben seguirse para asegurar que el sistema funcione dentro de la ley, y los requisitos éticos. Estos últimos son puestos en un sistema para asegurar que será aceptado por sus usuarios y por el público en general.

5.4.2.3. Requisitos de dominio

Al mismo nivel de los requisitos funcionales y no funcionales se encuentran los Requisitos del dominio, que según (Sommerville, 2011, pág. 110) son aquellos que provienen del dominio de aplicación del sistema y que reflejan las características y restricciones de ese dominio. Pueden ser funcionales o no funcionales. Es importante tener en cuenta que, en realidad, la distinción entre diferentes tipos de requisitos no es tan clara como sugieren estas definiciones. Por ejemplo, un requisito del usuario sobre seguridad podría parecer un requisito no funcional. Sin embargo, cuando se desarrolla en detalle, puede generar otros requisitos que son claramente funcionales, como la necesidad de incluir en el sistema recursos para la autenticación del usuario.

Es de vital importancia tener presente la anterior clasificación de los requisitos puesto que deberá quedar consignada con claridad en el documento de especificación. Dicha clasificación proporciona una perspectiva global del sistema, lo que permite a los involucrados precisar factores claves que conlleven al éxito del proyecto.

5.5. Niveles de descripción de los requisitos

Además de las clasificaciones anteriormente descritas, también existen diferentes niveles de descripción de un requisito, los cuales según (Camacho Zambrano, 2005, pág. 25):

“Permiten hacer una clara separación entre los diferentes tipos de requisitos que se pueden concebir en un documento de requisitos. Son necesarios para evitar errores y mejorar la descripción de los mismos. El clasificar los requisitos en estos niveles facilita su entendimiento y su descripción”.

5.5.1. A nivel de negocio

A nivel del Negocio se encuentran aquellos requisitos que representan objetivos de alto nivel para la organización o el cliente que requiere el producto. Estos requisitos son la necesidad principal por la cual se empieza la construcción o mejora del producto. Estos requisitos se

caracterizan por ser descritos de manera muy generalizada en términos de beneficios o necesidades de la organización; y se expresan en un lenguaje natural. En ocasiones son llamados los objetivos del software.

5.5.2. A nivel de usuario

A *nivel del Usuario* los requisitos describen tareas que los usuarios deben estar en capacidad de cumplir con el producto de software que se está describiendo, son conocidos como requisitos del usuario. Estos requisitos son descritos con frases usando lenguaje natural complementado con diagramas, describiendo las expectativas acerca de lo que el sistema debe proveer y las restricciones sobre las cuales debe operar.

5.5.3. A nivel de sistema

A *nivel del Sistema* los requisitos hacen referencia a la funcionalidad que debe ser construida para permitir al producto realizar sus tareas, en términos de las necesidades del sistema. Los requisitos del sistema se enfocan en las funciones del sistema, los servicios y las restricciones de operabilidad en detalle. El documento que contenga los requisitos del sistema debe ser sumamente preciso y definir de manera exacta lo que va a ser implementado. Debe ser parte del contrato entre el comprador o cliente del sistema y desarrollador del mismo.

A continuación, se muestra gráficamente la relación entre los diferentes niveles de descripción de requisitos con el objetivo de conseguir una especificación de requisitos de software:

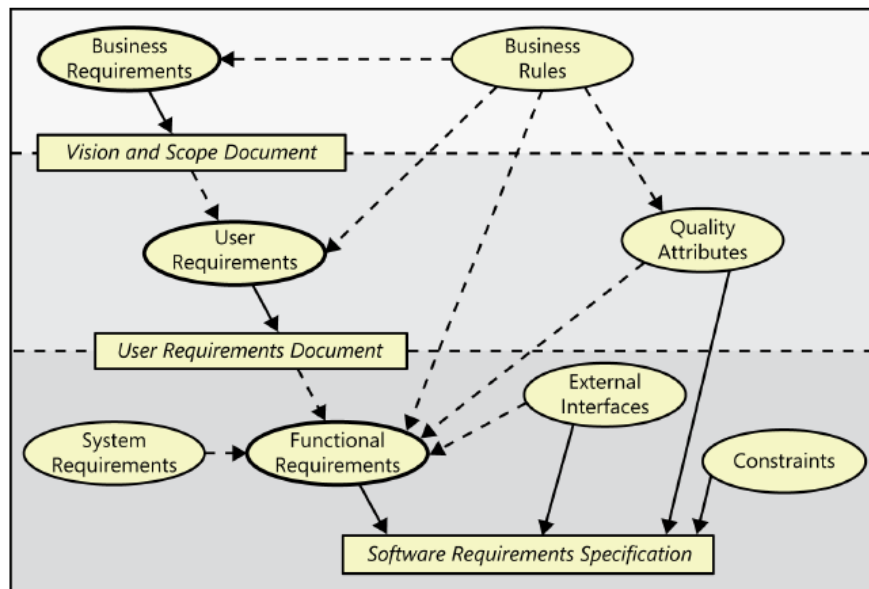


ILUSTRACIÓN 8. RELACIÓN ENTRE NIVELES DE DESCRIPCIÓN (WIEGERS & BEATY, 2013, PÁG. 8)

Por consiguiente, establecer dichos niveles de descripción constituye una tarea fundamental para la especificación de requisitos, puesto que de cada uno de ellos se derivan insumos importantes para adquirir un dominio total del problema y, por consiguiente, proponer una solución alineada a los objetivos del negocio y a las expectativas del usuario.

Además de identificar los niveles de descripción de requisitos, es sustancial comprender la relación de cada uno de estos niveles en las diferentes fases del proceso de desarrollo de requisitos, como se muestra en la *Ilustración 10*.

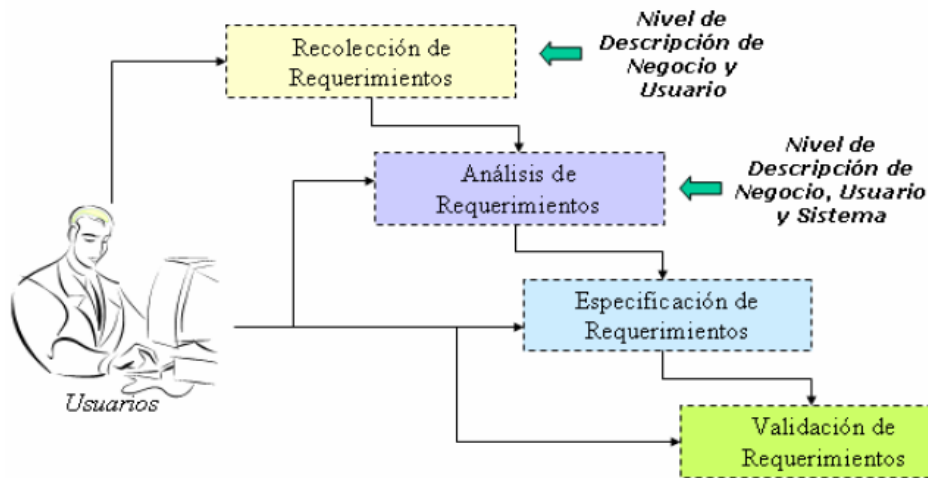


ILUSTRACIÓN 9. NIVELES DE DESCRIPCIÓN EN EL PROCESO DE REQUISITOS (CAMACHO ZAMBRANO, 2005, PÁG. 68)

5.6. Características de un buen requisito

Otro aspecto fundamental en la especificación de requisitos es conocer las diferentes características que tienen los requisitos bien formulados, puesto que autores como (Wiegiers & Beaty, 2013, pág. 206) mencionan que los requisitos deben satisfacer varias características y cumplir ciertos criterios. Si no lo hacen, deben ser reformulados hasta conseguirlo. La *Ilustración 11* propone una idea general de estas características:

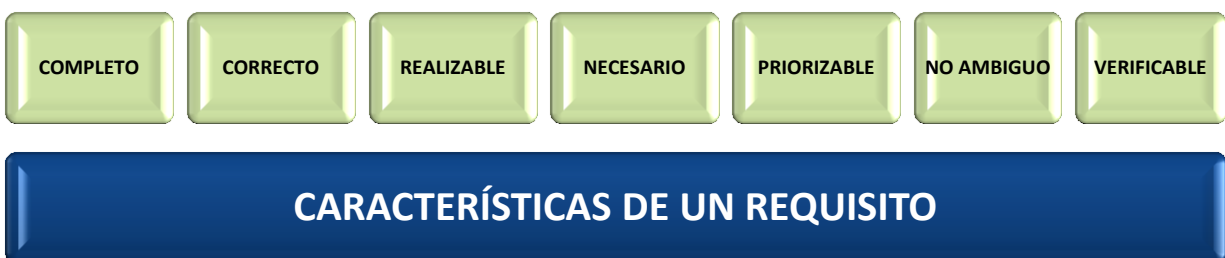


ILUSTRACIÓN 10. CARACTERÍSTICAS DE UN BUEN REQUISITO. (ELABORACIÓN PROPIA A PARTIR DE WIEGERS & BEATY, 2013)

El cumplimiento de cada una de las características de un requisito bien formulado contribuye a un mejor entendimiento de las expectativas de los interesados en el proyecto y a la producción de un documento de especificación de requisitos que cumpla con criterios de calidad.

5.7. Especificación de requisitos de software

Respecto a la Especificación de requisitos de software concretamente, se citan diferentes autores con el propósito de tener un amplio bagaje en esta fase del desarrollo de requisitos.

De acuerdo con (IREB, 2015), toda forma de representación más o menos formal de los requisitos, desde escritura en prosa hasta diagramas con semántica formal, se denomina “técnica de documentación”. La (IEEE, 2014), indica que, para sistemas complejos, se elaboran tres tipos de documentos: definición del sistema, requisitos del sistema, y requisitos del software. Para sistemas simples, solamente se requiere el tercer documento. Dichos documentos se describen a continuación.

5.7.1. El Documento de definición del sistema

A veces conocido como documento de requisitos de usuario o documento de concepto de operaciones, registra los requisitos del sistema. Este documento define los requisitos del sistema de alto nivel desde la perspectiva de dominio. Sus lectores son representantes del sistema como lo son usuarios/clientes, por lo cual su contenido debe ser redactado en términos de dominio. Además, el documento lista los requisitos del sistema, junto con la información sobre los objetivos generales para el sistema, su entorno, una declaración de las limitaciones, supuestos, y los requisitos no funcionales. Puede incluir modelos conceptuales diseñados para ilustrar el contexto del sistema, el uso de escenarios y las principales entidades de dominio, como también flujos de trabajo.

5.7.2. El Documento de especificación de requisitos de sistema

Trata de separar la descripción de los requisitos del sistema de la descripción de los requisitos de software. Desde este punto de vista, los requisitos del sistema se especifican y los requisitos de software se derivan de los requisitos del sistema (IEEE, 2014, pág. 11).

5.7.3. El Documento de especificación de requisitos de software

Establece la base para el acuerdo entre los clientes y los contratistas o los proveedores, donde se establece que es lo que el producto de software va hacer, así como lo que no se espera que haga.

Para (Sommerville, 2005, pág. 124) el documento de requisitos del software, también denominado especificación de requisitos del software o SRS,

“Es la declaración oficial de qué deben implementar los desarrolladores del sistema. Debe incluir tanto los requisitos del usuario para el sistema como una especificación detallada de los requisitos del sistema. En algunos casos, los dos tipos de requisitos se pueden integrar en una única descripción. En otros, los requisitos del usuario se definen en una introducción a la especificación de los requisitos del sistema. Si existe un gran número de requisitos, los detalles de los requisitos del sistema se pueden presentaren un documento separado”.

Por su parte, (Wiegiers & Beaty, 2013, pág. 183) menciona que “*la especificación de requisitos de software tiene muchos nombres en diferentes organizaciones, aunque las organizaciones no utilizan estos términos de la misma manera.*” Algunas por ejemplo lo llaman documento de requisitos del negocio (BRD), otras, especificación funcional, especificación de producto, especificación del sistema, o simplemente documento de requisitos.

La SRS establece las funciones y capacidades que un sistema de software debe proporcionar, sus características y las limitaciones que debe respetar. Debe describir de forma tan completa como sea necesaria los comportamientos del sistema en distintas condiciones, así como las cualidades del sistema deseadas tales como el rendimiento, la seguridad y facilidad de uso. La SRS es la base para la planificación, diseño y codificación del proyecto, así como también para las pruebas del sistema y la documentación del usuario. Sin embargo, no debe contener el diseño, la construcción, las pruebas, o los detalles de gestión del proyecto u otras limitaciones de diseño e implementación que sean conocidas.

Por otro lado, el mismo autor menciona que, para la especificación de requisitos de software en proyectos ágiles, los equipos podrían comenzar la especificación escribiendo la información suficiente para cada historia de usuario, de modo que las partes interesadas tengan una comprensión general de lo que trata la historia y así puedan priorizar su relación con otras historias. Esto permite al equipo planificar la asignación de historias de usuario específicas a las iteraciones (Wiegiers & Beaty, 2013, pág. 199).

5.7.4. Características de una SRS

Además de lo anterior, es importante considerar que la Especificación de Requisitos debe satisfacer un conjunto de características con el ánimo de que el proceso de desarrollo de software inicie con calidad desde sus primeras etapas, es así cómo (Camacho Zambrano, 2005, pág. 30) relaciona las diferentes cualidades que se deben conseguir en una buena especificación. Estas cualidades se pueden medir, por ejemplo, de forma similar mediante la incorporación de variables de calidad establecidas en normas ISO como la 9126-1:4 de 2001 y la 25000 que la reemplaza, en la que se miden características internas y externas de calidad. La *Ilustración 12* da cuenta de las cualidades que se deben conseguir en una buena especificación de requisitos:



ILUSTRACIÓN 11. CUALIDADES DE UNA ESPECIFICACIÓN DE REQUISITOS DE CALIDAD (ELABORACIÓN PROPIA A PARTIR DE CAMACHO ZAMBRANO, 2005)

Si bien existen características que ayudan a determinar si los requisitos están bien formulados, no se debe descartar la posibilidad de que se presenten inconvenientes durante la etapa de especificación de requisitos, que deben ser identificados y prevenidos. A continuación, se presenta un listado con los problemas y dificultades más comunes en este proceso, según (Pérez Huebe, 2005, pág. 12):

- Los requisitos no son obvios y vienen de muchas fuentes.
- Son difíciles de expresar en palabras (el lenguaje es ambiguo).
- Existen muchos tipos de requisitos y diferentes niveles de detalle.
- La cantidad de requisitos en un proyecto puede ser difícil de manejar.
- Nunca son iguales. Algunos son más difíciles, más riesgosos, más importantes o más estables que otros.
- Los requisitos están relacionados unos con otros, y a su vez se relacionan con otras partes del proceso.
- Cada requisito tiene propiedades únicas y abarcan áreas funcionales específicas.
- Un requisito puede cambiar a lo largo del ciclo de desarrollo.

5.7.5. Especificación de requisitos de software en lenguaje natural

Con el fin de reducir al máximo la ocurrencia de los inconvenientes listados anteriormente, se han propuesto e implementado diversas maneras de llevar a cabo la especificación de requisitos, algunas de ellas según (Camacho Zambrano, 2005, pág. 26) consisten en especificar requisitos utilizando un *lenguaje natural estructurado*. *“Se considera lenguaje natural al lenguaje utilizado a diario entre los integrantes de la organización. Se caracteriza por estar orientado a una descripción más humana y generalizada, y no regido por consideraciones técnicas específicas.”*

El lenguaje natural estructurado es una forma de redactar los requisitos del sistema donde la libertad del redactor de los requisitos está limitada y donde todos los requisitos se redactan de una forma estándar. La ventaja de este enfoque es que mantiene la mayor parte de la expresividad y comprensión del lenguaje natural, pero asegura que se imponga cierto grado de uniformidad en la especificación. Las notaciones del lenguaje estructurado limitan la terminología que se puede utilizar y emplean plantillas para especificar los requisitos del sistema. Pueden incorporar construcciones de control derivadas de los lenguajes de programación y manifestaciones gráficas para dividir la especificación.

En la misma línea (Heninger, 1980 citado en Sommerville, 2005, pág. 134)

“Describe uno de los primeros proyectos que utilizó el lenguaje natural estructurado para especificar los requisitos del sistema. Se diseñaron formularios de propósito especial para describir la entrada, la salida y las funciones de un sistema software para un avión. Estos requisitos se especificaron utilizando dichos formularios”.

Así, la utilización de un enfoque basado en formularios para especificar los requisitos del sistema, se resume en definir formularios o plantillas que sirvan como estándar para expresar los requisitos.

Se puede estructurar además la especificación de los requisitos alrededor de los objetos que manipula el sistema, las funciones ejecutadas por el sistema o los eventos procesados por éste.

En disciplinas de ingeniería tales como la ingeniería civil y la eléctrica, se ha logrado con el tiempo el desarrollo de mejores técnicas matemáticas. La industria de la ingeniería no ha tenido dificultad en aceptar la necesidad del análisis matemático e incorporarlo en sus diferentes procesos. El análisis matemático es una parte rutinaria del proceso de desarrollo y validación del diseño de un producto.

5.8. Técnicas de Ingeniería de Requisitos tomadas de metodologías ágiles

Con la intención de no dejar de lado las metodologías ágiles, muy posiblemente utilizadas por las MiPymes desarrolladoras de software, se presentan a continuación algunas técnicas que tratan el desarrollo de los requisitos propuestas por algunas metodologías ágiles como *Agile Modeling (AM)* y *Lean Software Development (LSD)*, metodologías seleccionadas como representativas de los procesos ágiles, pues son quizá las que tienen mayores elementos para la especificación y validación de requisitos de software.

En AM se encuentra una recopilación de las mejores prácticas para una documentación y modelado efectivo de sistemas software (Burgos Pintos & Garbarino de la Rosa, 2010), como se muestra en la *Ilustración 12* presentada a continuación.

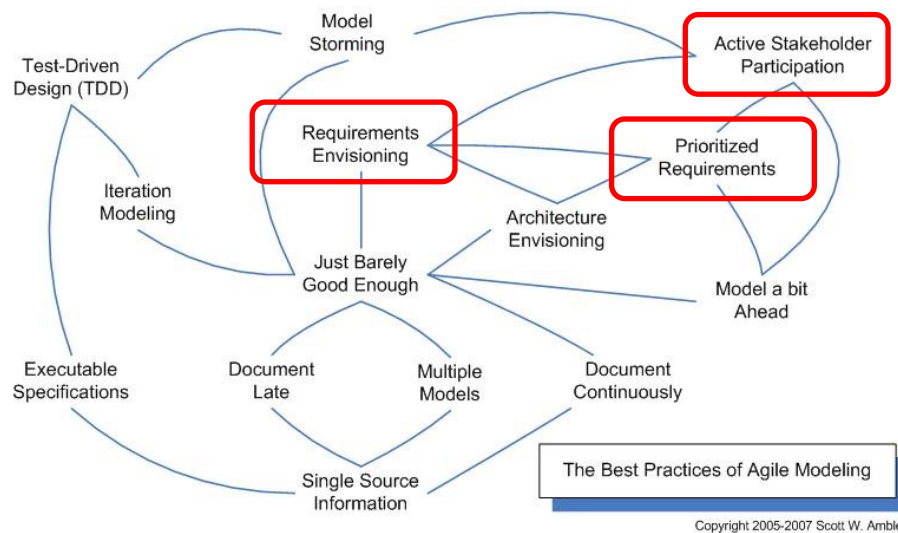


ILUSTRACIÓN 12. MEJORES PRÁCTICAS DE AM (AMBLER, 2014)

Propone también AM que existen varios tipos de requisitos, los cuales se pueden separar principalmente en dos categorías:

Comportamiento: un requisito de comportamiento describe cómo un usuario interactúa con un sistema, cómo alguien va a utilizar un sistema, o cómo un sistema cumple una función de negocio (reglas de negocio). Estos se refieren a menudo a los requisitos funcionales.

No comportamiento: describe las características técnicas de un sistema, relacionadas con la disponibilidad, seguridad, rendimiento, interoperabilidad, fiabilidad y confiabilidad.

Para lograr los modelos y documentos relacionados en la *Ilustración 13*, AM propone el uso de las siguientes técnicas y artefactos durante las diferentes etapas del proceso:



ILUSTRACIÓN 13. TÉCNICAS Y ARTEFACTOS DE AM (ELABORACIÓN PROPIA A PARTIR DE (BURGOS PINTOS & GARBARINO DE LA ROSA, 2010))

Las técnicas marcadas en color rojo fueron las tenidas en cuenta por los autores para la elaboración del procedimiento para especificar y validar requisitos de software, además de aquellas analizadas en el Capítulo 7. Resultados.

Por su parte y de acuerdo a (Letelier Torres & Sánchez López, 2006), *Lean Software Development (LSD)* indica respecto a los requisitos que el enfoque ágil asume que los requisitos no pueden ser establecidos con seguridad desde el comienzo del proyecto y se acepta el cambio como algo inevitable y cuestiones como el plazo y el coste se pueden establecer antes del alcance, pues se persigue la puesta a disposición del cliente cada cierto tiempo y dentro de un coste de un

determinado conjunto de requisitos (los de mayor prioridad). De esta forma, el cliente puede saber cuándo obtendrá un conjunto de requisitos y el proveedor los recursos de los que dispone para producirlos. Las decisiones se toman en torno a la prioridad de los requisitos y el rol del proveedor será asegurar el flujo constante de producto según esta prioridad. Además, parte del conjunto de requisitos no abordados, por tener menor prioridad y no estar en el alcance de la iteración, a menudo acaban considerándose innecesarios. El producto obtenido con este enfoque es más acorde a la filosofía “Lean”. (Letelier Torres & Sánchez López, 2006, pág. 60).

LSD propone 7 principios del desarrollo enunciados por (Poppendieck & Poppendieck, 2003), revisados y redefinidos en 2006 por el mismo autor son los siguientes:

- 1) Eliminar el desperdicio: hacer desaparecer del proceso y el producto todo aquello que no aporta valor al cliente.
- 2) Calidad integrada: el desarrollo ha de realizarse desde el primer momento con calidad. Las acciones correctivas han de emprenderse lo más próximo a que se detecta su necesidad y lo que es más importante, debe existir un enfoque preventivo: se deben buscar las condiciones que eviten si quiera la posibilidad de que se den errores.
- 3) Crear conocimiento: el desarrollo de Software es un proceso de creación de conocimiento que va evolucionando a medida que se va produciendo y que, por tanto, se ha de evitar el derroche de tratar de capturarlo prematuramente. Se han de centrar esfuerzos en mejorar este conocimiento, en hacerlo más profundo y en dar respuesta al cambio.
- 4) Aplazar las decisiones (entregar tan rápido como sea posible): consecuencia de lo anterior, es necesario disponer de medios que permitan, una vez tomada una decisión, materializarla, sin sacrificar la calidad.
- 5) Delegar la responsabilidad al equipo: el empoderamiento, dotar a aquellos en los que reside el conocimiento y realizan el trabajo del liderazgo suficiente para tomar decisiones y realizarlo, evitando pasos adicionales de aceptación a otras instancias que obstaculizan el flujo normal de actividad.
- 6) Respetar a las personas (construir con integridad): integridad conceptual, dado que debe responder a una necesidad del cliente, éste debe percibir el producto como algo coherente, donde los requisitos a los que da solución se observan como un todo cohesionado armónicamente. Además, se necesita integridad técnica: una arquitectura coherente, usable, que responde a su cometido y a la que se puede dar mantenimiento, adaptar y ampliar.
- 7) Visión global: se debe evitar la tendencia a realizar mejoras locales a favor de un enfoque global.

Los principios mencionados anteriormente y especialmente aquellos relacionados con los valores de LD que proveen técnicas para la **identificación del valor del negocio, la activa participación del cliente y la satisfacción del cliente como máxima prioridad** fueron tenidos

en cuenta para la elaboración del procedimiento y se verán reflejados en la sección 7.3. Diseño del procedimiento para la Especificación y Validación de requisitos de software (PEVReS).

Quizá algunas de las técnicas más representativas dentro de las metodologías ágiles mencionada anteriormente son las *Historias de usuarios* y los *Prototipos*. Según (Letelier Torres & Sánchez López, 2006), las Historias de usuario son la técnica utilizada para especificar los requisitos del software. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento las historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas (Jeffries, Anderson, & Hendrickson, 2001).

La técnica de Prototipos, por su parte, permite que todo el sistema, o algunas de sus partes, se construyan rápidamente para comprender con facilidad y aclarar ciertos aspectos en los que se aseguren que el desarrollador, el usuario y el cliente estén de acuerdo en lo que se necesita, así como también la solución que se propone para dicha necesidad y de esta forma minimizar el riesgo y la incertidumbre en el desarrollo, esta técnica se encarga del desarrollo de diseños para que estos sean analizados y prescindir de ellos a medida que se adhieran nuevas especificaciones, es ideal para medir el alcance del producto, pero no se asegura su uso real. (Lawrence Pfleeger, 2002)

5.9. Las Pequeñas y Medianas Empresas del software

Las empresas de la industria del software tienen la posibilidad de adoptar enfoques, tanto para asegurar la calidad de sus procesos como la de sus productos. Sin embargo, no todas las empresas de la industria del software están en la capacidad financiera y operativa para implementar algunos de estos procesos y/o modelos, pues en su mayoría, según (SUPERSOCIEDADES, 2015, pág. 6), son pequeñas y medianas empresas (PYMES) que poseen ciertas características específicas.

Según (El Congreso de Colombia, 2000) en la Ley 590 del 2000 por la cual se dictan disposiciones para promover el desarrollo de las micro, pequeñas y medianas empresas (MiPymes) en Colombia, los parámetros para clasificarlas corresponden a los siguientes:

TABLA 7. CARACTERIZACIÓN DE PEQUEÑAS Y MEDIANAS EMPRESAS EN COLOMBIA

TIPO	PLANTA DE PERSONAL	ACTIVOS TOTALES
Microempresa	Menor o igual a 10	Inferior a 501 SMMLV
Pequeña Empresa	Entre 11 y 50	Entre 501 y menos de 5.001 SMMLV
Mediana Empresa	Entre 51 y 200	Entre 5.001 y 15.000 SMMLV

En la misma línea, (De la Cruz Londoño & Castro Guevara, 2014) establecen que algunas de las características principales de las pequeñas empresas son:

- El equipo de trabajo es poco calificado, suelen ser grupos menores a diez personas y sus proyectos no son de más de medio año.
- Cuentan con escasos recursos.

- El proceso de desarrollo y gestión del proyecto (planeación, organización, monitoreo) se basa en técnicas informales.
- No se encuentra una estructura organizacional clara y no se definen roles dentro de la organización.

5.10. Software Process Engineering Metamodel (SPEM)

SPEM (Software Process Engineering Metamodel) (OMG, 2002), es una especificación del OMG (Object Management Group) que está basado en MOF (MetaObject Facility) y es un metamodelo. SPEM permite representar una familia de procesos de desarrollo de software y sus componentes. Constituye un tipo de ontología de procesos de desarrollo de software (Wautelet, Kolp, & Achbany, 2006). Para ello provee un conjunto de elementos de modelado de procesos para describir cualquier proceso de desarrollo de software sin agregar modelos o restricciones de alguna área o disciplina específica, tal como gestión de proyectos o análisis (Menéndez Domínguez & Castellanos Bolaños, 2015).

SPEM proporciona una sintaxis y estructura para cada aspecto de los procesos desarrollo, incluyendo:

- Roles.
- Tareas.
- Artefactos.
- Lista de verificación
- Productos de trabajo.
- Técnicas y herramientas.
- Estructuras de trabajo.
- Capacidad de rastreo y refinamiento.
- Ayuda sensible al contexto, guía y lineamientos.
- Descripción textual de elementos.

Como se verá en el capítulo siguiente, SPEM fue utilizado para modelar el procedimiento para la especificación y validación de requisitos de software con la intención de facilitar su presentación y despliegue.

6. ESTRATEGIA METODOLÓGICA

El proyecto se realizó siguiendo las etapas propuestas por el Ciclo PHVA⁵, del cual puede verse un esquema general en la *Ilustración 14*. Esto se hace considerando que se desea elaborar un procedimiento como un elemento que se ajuste a las necesidades de las empresas y ellas puedan usarlo para mejorar uno de los procesos de desarrollo de software: el relacionado con los requisitos. Es de anotar, que definir un nuevo procedimiento para una empresa (que remplace lo que se hacía antes) es en sí una mejora de procesos, pero es importante aclarar que queda fuera del alcance de este proyecto medir la mejora lograda en las empresas a partir de la aplicación del procedimiento, pues este trabajo se trata sobre la identificación de técnicas y prácticas concretas para la especificación y validación de requisitos de software y su aplicación mediante dos pilotos en MiPymes desarrolladoras de software de la ciudad de Pereira. Las actividades que se llevaron a cabo en cada una de las etapas del ciclo PHVA se explican a continuación:

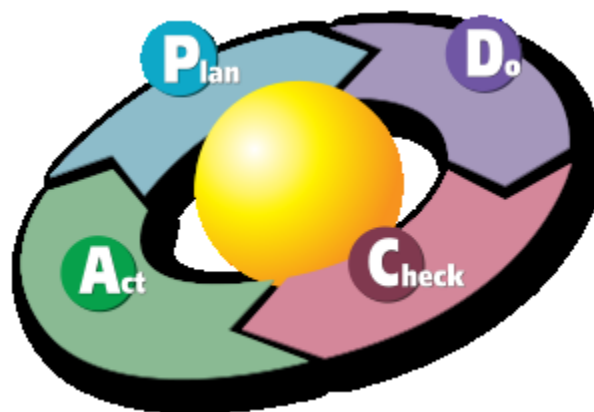


ILUSTRACIÓN 14. ESTRUCTURA DEL CICLO PHVA (FUENTE: [HTTP://WWW.PDCAHOME.COM/5202/CICLO-PDCA/](http://www.pdcahome.com/5202/ciclo-pdca/))

ETAPA 1 – PLANEAR

Esta etapa a su vez se dividió en dos grandes actividades:

La primera actividad fue una exploración que se realizó con respecto a la especificación y validación de requisitos, teniendo en cuenta tanto el estado del arte como también una descripción y una caracterización de una muestra por conveniencia representativa de una población (empresas desarrolladoras de software de la ciudad de Pereira).

La segunda actividad consistió en la generación del procedimiento para la especificación y validación de requisitos de software.

Se hace fundamental explicar en detalle cada de una de las actividades, donde se provee la información necesaria para determinar cómo se consiguió la finalidad de cada una.

⁵ También conocido como Círculo de Deming, ciclo PDCA (Plan-Do-Check-Act) por sus siglas en inglés o Espiral de mejora continua, propuesto por Edwards Deming.

ACTIVIDAD 1: Exploración

Durante la investigación se realizó un levantamiento de información en diferentes fuentes de información como lo son repositorios institucionales, bases de datos especializadas y bibliografía física de las diferentes bibliotecas de la región, con el propósito de consolidar un referente contextual que comprendiera una descripción del área problemática y una revisión de antecedentes conocido también como el estado del arte de la investigación, esta última estuvo desagregada desde el ámbito regional, nacional e internacional, además se consolidó otro capítulo denominado marco teórico, donde se detallaron las diferentes teorías y conceptos relacionados con el proceso de desarrollo de los requisitos de software y en especial a la especificación y validación de los mismos.

El estado del arte fue realizado por los estudiantes de la siguiente manera: Juan Guillermo Gálvez Botero estuvo encargado de los antecedentes del orden regional, nacional e internacional; Alonso Toro Lazo se encargó del marco teórico. Cada capítulo fue revisado por el otro estudiante, para establecer aquellos elementos pertinentes para el desarrollo del proyecto.

Instrumentos utilizados: para la recolección e identificación de los antecedentes tanto del orden regional, como nacional e internacional se usaron tablas como las siguientes, que describen de manera concreta la información relevante de los mismos:

TABLA 8. ANTECEDENTES RELEVANTES

AUTOR	TÍTULO	UNIVERSIDAD	TÉCNICAS ESPECIFICACIÓN	TÉCNICAS VALIDACIÓN	FALENCIAS DEL PROYECTO	APORTE ESPECÍFICO

Posteriormente, se realizó un encuentro con los empresarios del software de la ciudad de Pereira teniendo en cuenta los criterios determinantes para su clasificación los cuales se relacionan en su apartado correspondiente (Población y Muestra), además se contó con el acompañamiento de Parquesoft Pereira que incluyó a las empresas que hacen parte de este modelo de emprendimiento, una vez realizada dicha clasificación se realizó un muestreo por conveniencia representativo de la población y fueron estas las empresas invitadas al encuentro que tuvo las siguientes finalidades:

- Presentar el proyecto con el ánimo de poner en contexto al sector productivo sobre el trabajo de investigación que se estaba llevando a cabo.
- Aplicar una encuesta como instrumento de levantamiento de información que tenía como objetivo diagnosticar, proponer y describir mejoras en pro de tener un proceso de desarrollo de requisitos de calidad haciendo énfasis, en este caso, en la especificación y validación de requisitos como objeto de estudio de la investigación.

Población y muestra

Muestra por conveniencia de las empresas desarrolladoras de software en la ciudad de Pereira.

Criterios de selección de la muestra:

- Empresas cuya actividad económica este catalogada en la División 62 **DESARROLLO DE SISTEMAS INFORMÁTICOS (PLANIFICACIÓN, ANÁLISIS, DISEÑO, PROGRAMACIÓN, PRUEBAS), CONSULTORÍA INFORMÁTICA Y ACTIVIDADES RELACIONADAS** y específicamente en el desarrollo de sistemas informáticos (planificación, análisis, diseño, programación, pruebas) con código CIU 620 6201 según la clasificación Industrial Internacional Uniforme impartida por el Departamento Administrativo Nacional de Estadística (DANE).
- Que se encuentren ubicadas en la ciudad de Pereira.
- Prestas a suministrar la información requerida relacionada con el proyecto.

TABLA 9. INFORMACIÓN DE LA MUESTRA

Diseño muestra	Muestra por conveniencia
Población objetivo	MiPymes desarrolladoras de software de la ciudad de Pereira
Tamaño de la muestra	17 empresas de la ciudad de Pereira (De 30 empresas convocadas).
Momento estadístico	Octubre 2015 – enero 2016.
Financiación	Recursos propios.

Instrumentos utilizados: se hizo uso de una encuesta (**ANEXO A – Encuesta a empresas**), la cual está compuesta por siete conjuntos de preguntas:

- Formación profesional del personal encargado del desarrollo de los requisitos.
- Planificación, participación, capacidad y experiencia de los *stakeholders* en la captura de requisitos.
- Documentación de los procesos de negocio de las organizaciones llevada a cabo por el personal encargado del desarrollo de los requisitos.
- Uso de metodologías para el desarrollo de requisitos.
- Especificación de Requisitos de software.
- Validación de requisitos.
- Gestión de requisitos de software.

Con los resultados obtenidos de la encuesta se realizó un análisis para determinar la forma como las MiPymes desarrolladoras de software de la ciudad de Pereira llevan a cabo la especificación y validación de requisitos de software.

Esta etapa fue realizada por ambos estudiantes: Alonso Toro Lazo estuvo encargado del diseño y elaboración de la encuesta, Juan Guillermo Gálvez Botero fue el encargado de aplicar la encuesta a los empresarios. La consolidación y análisis de los datos se llevó a cabo por ambos estudiantes.

ACTIVIDAD 2: Descriptiva

Se recibió como entrada el marco teórico, los antecedentes y la descripción y caracterización que detalla cómo las empresas desarrolladoras de software llevan a cabo la especificación y validación de requisitos (Actividad 1). Esta información fue analizada a través de una comparación con los antecedentes y el marco teórico, y se determinaron las entradas, las precondiciones, las actividades y tareas del procedimiento, las salidas y el objetivo.

La comparación se realizó usando tablas, como las presentadas a continuación, donde se identificaron los elementos que contemplan los diferentes autores del marco teórico que fueron referentes del proyecto para la especificación y validación de requisitos, así como las técnicas que proponen los autores estudiados en los antecedentes y las prácticas usadas en las empresas, junto con el cubrimiento que logra cada práctica del elemento. Por ejemplo:

TABLA 10. TÉCNICAS EXTRAÍDAS DEL MARCO TEÓRICO

Técnica	Autor 1	Autor 2	Autor 3	Puntaje
Técnica 1				
Técnica 2				
...				

TABLA 11. TÉCNICAS EXTRAÍDAS DE LOS ANTECEDENTES

Técnica	Antecedente 1	Antecedente 2	Antecedente 3	Puntaje
Técnica 1				
Técnica 2				
...				

Una vez determinadas las técnicas mencionadas o recomendadas desde el marco teórico y los antecedentes, se realizó un análisis de la información a través de tablas de cruce como las presentadas a continuación, que permitieron encontrar las mejores prácticas, que luego se organizaron y complementaron para que incluyeran todos los elementos requeridos en un procedimiento.

TABLA 12. TABLA DE CRUCE DE INFORMACIÓN PARA ANÁLISIS

Técnica	MARCO TEÓRICO	ANTECEDENTES	Total	Cubrimiento*
Técnica 1				
Técnica 2				
...				

*Requerido, Recomendado, Opcional

Finalmente, se realizó un análisis de las técnicas requeridas y recomendadas usadas por las empresas, teniendo en cuenta los resultados de la encuesta planteada en la Etapa 1.

TABLA 13. ANÁLISIS DE TÉCNICAS USADAS POR LAS EMPRESAS

TÉCNICA	Empresa 1	Empresa 2	Empresa 3	% de uso
Técnica 1				
Técnica 2				

...				
-----	--	--	--	--

Esta comparación de técnicas y prácticas se vio enriquecida por el aporte de los dos estudiantes de maestría autores de este proyecto, pues cada uno combina su experiencia: uno en la industria de software y otro en la academia.

El diseño del procedimiento para la especificación y validación de requisitos de software se llevó a cabo bajo el estándar SPEM soportado con una descripción textual explícita, así como un conjunto de guías, formatos y recomendaciones. Lo anterior, con la intención de facilitar su presentación y despliegue.

Las tablas anteriormente descritas fueron el insumo principal para establecer el diseño del procedimiento para la especificación y validación de requisitos de software, inicialmente en Excel pues se consideró por conocimiento de los autores que es la herramienta que con mayor frecuencia usan las MiPymes para especificar sus procesos y procedimientos. A continuación, se presenta el formato que se propuso para el procedimiento:

TABLA 14. FORMATO PROPUESTO PARA EL PROCEDIMIENTO

Título:								
Descripción:								
Precondiciones:	No.	Precondición						
Entradas:	No.	Entrada						
Actividades-Tareas:	No.	Actividad	No.	Técnica	No.	Tareas (pasos)	Responsables	Guías y Herramientas
Salidas:	No.	Salida						

Instrumentos utilizados: diagramas de actividades, formatos, tablas y guías.

Esfuerzo invertido en esta etapa: 9 horas semanales por hombre (2 personas) durante 12 meses.

ETAPA 2 – HACER

Esta etapa comprendió el llevar a cabo dos pilotos de aplicación con dos unidades de análisis mediante los cuales se presentó la experiencia vivida en dos MiPymes desarrolladoras de software, donde cada estudiante se encargó de realizar una aplicación piloto del procedimiento y posterior a ello se consolidaron los resultados en un documento que describe cada una de las actividades propuestas para su aplicación.

Las actividades llevadas a cabo en esta etapa fueron:

- Selección del proyecto donde se aplicó el piloto en la empresa.
- Recopilación de datos de proyectos anteriores y determinación de aquellos que sirvieron para analizar los resultados del piloto. Por ejemplo: tiempo invertido en requisitos, errores, cantidad de modelos/documentos elaborados, revisiones realizadas, satisfacción de los desarrolladores mediante la aplicación de una encuesta de expectativas (**ANEXO F - Encuesta Expectativas**) y una encuesta de rendimiento (**ANEXO G – Encuesta Rendimiento**).
- Definición de la forma en la cual se le haría el seguimiento y se medirían los resultados (a partir de los datos identificados en el paso anterior).
- Identificación de los actores que participarían y el rol de cada uno.
- Comprometer a los participantes, explicándoles cómo se haría el seguimiento y la importancia de su participación crítica.
- Hacer la capacitación sobre el procedimiento.
- Ejecutar el proyecto con el nuevo procedimiento.
- Ir recopilando los datos necesarios (de acuerdo con lo definido en la planeación).

Instrumentos utilizados: se utilizó una encuesta de expectativas (**ANEXO F - Encuesta Expectativas**) y una encuesta de rendimiento (**ANEXO G - Encuesta Rendimiento**) con el ánimo de determinar el nivel de satisfacción, las cuales condujeron a aprender de la experiencia, aportando a los ajustes del procedimiento que se realizaron en la etapa del Actuar, además se utilizó un formato donde reposa la información obtenida con el ánimo analizar y reflexionar sobre la experiencia. De cada paso se tienen registros: documentos, actas de reuniones, etc.

Esfuerzo invertido en esta etapa: 9 horas semanales por hombre (2 personas) durante 6 meses.

ETAPA 3 – VERIFICAR

Esta etapa tuvo como propósito verificar que la aplicación del procedimiento se haya realizado de la forma indicada, con la intención de determinar los resultados obtenidos y las mejoras o ajustes a implementar, todo ello debidamente documentado.

Las actividades llevadas a cabo en esta etapa fueron:

- Análisis de los datos recolectados.
- Documentación de los resultados, destacando cuando se presentaron logros importantes o mejoras y cuando no. En los casos en los que no se lograron mejoras se identificaron y documentaron las causas.
- Seguimiento de la aplicación del procedimiento (realización de las actividades y aplicación de las técnicas propuestas) y registro de los casos en los que no se aplique el procedimiento de manera correcta.
- Documentación de lecciones aprendidas: lo bueno (que se debe continuar) y lo malo (que se debe corregir).

Esfuerzo invertido en esta etapa: 9 horas semanales por hombre (2 personas) durante 3 meses.

ETAPA 4 – ACTUAR

En esta etapa se realizaron los ajustes al procedimiento propuesto de acuerdo a las dificultades y mejoras obtenidas durante la etapa de verificación, con la intención de lograr un procedimiento más ajustado a las necesidades de las MiPymes. Para ello se llevaron a cabo las siguientes actividades:

- Realizada la aplicación y la validación del procedimiento propuesto para la especificación y validación de requisitos, se retomaron las consideraciones pertinentes y los resultados obtenidos con el ánimo de realizar los ajustes correspondientes (rediseño del procedimiento mediante modelado en SPEM para una mejor presentación y despliegue).
- Efectuar los ajustes necesarios en el procedimiento.
- Comunicar los resultados a las empresas para que puedan usar el nuevo procedimiento en otros proyectos.

Esfuerzo invertido en esta etapa: 9 horas semanales por hombre (2 personas) durante 3 meses.

7. RESULTADOS

7.1. Análisis del estado del arte de los requisitos

7.1.1. Análisis de Técnicas para Especificar y Validar Requisitos extraídas del Marco Teórico

Como parte de la *Etapa I-Exploratoria* de la metodología relacionada con el levantamiento del estado del arte de los requisitos de software y, con el fin de determinar las principales técnicas de especificación y validación de requisitos de software propuestas por los autores relacionados en el marco teórico, se asignaron pesos con valor de (1) a cada técnica identificada por cada autor, de tal manera que al hacer la sumatoria se pudieran determinar aquellas que lograron un mayor puntaje en un rango de valores de 1 a 4. Para efectos de selección, aquellas que obtengan puntajes superiores o iguales a dos (2) serán las analizadas con el propósito de ser incluidas en el procedimiento. La *Tabla 16* y la *Tabla 17* muestran el análisis realizado para las técnicas de especificación y las técnicas de validación (respectivamente), y clasifica de acuerdo a colores para mejor visibilidad, aquellas que obtuvieron mayores puntajes (rojo y verde).

Tanto la asignación de pesos como la selección de técnicas se hizo en consenso entre los autores del proyecto y el asesor del mismo (Juicio de Expertos), con la intención de dar relevancia a los antecedentes que son referentes principales para el proyecto. A su vez, se complementó identificando las ventajas y desventajas de las técnicas seleccionadas.

TABLA 15. CUBRIMIENTO DE ACUERDO A PESO

CUBRIMIENTO	PUNTAJE
Requerida	>3
Recomendada	2
Opcional	1

TABLA 16. TÉCNICAS PARA ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE EXTRAÍDAS DEL MARCO TEÓRICO

TÉCNICA	(IEEE, 2014)	(Wieggers & Beaty, 2013)	(Ambler, 2014)	(Jacobson, Spence, & Bittner, 2013)	PUNTAJE
Documento de la definición del sistema	X	X			2
Documento de Especificación de requisitos del sistema	X	X			2
Documento de Especificación de requisitos del software	X	X			2
Diagrama de flujo de datos (DFD)		X	X	X	3
Árbol de características		X			1
Diagramas de transición de estados (STD)		X			1
Diagramas entidad-relación (ERD)		X	X		2
Mapas de diálogo		X			1
Tablas de decisión		X			1
Árboles de decisión		X			1
Tablas de eventos de respuesta		X			1
Diagramas de casos de uso		X	X	X	3
Diagramas de actividad		X	X	X	3

Diccionario de datos		X			1
Prototipos		X	X		2
Historias de usuario		X		X	2
Diagrama de clases			X		1
Diagrama CRC (Class Responsibility Collaborator)			X		1
Diagramas de secuencia			X		1
Diagrama de estados			X		1
Escenarios de uso			X	X	2
Dividir los casos de uso				X	1
Notas adhesivas			X	X	2
Esquema de priorización MoSCoW				X	1
Diagrama de comunicación			X		1
Diagrama de componentes			X		1
Diagrama de estructura compuesto			X		1
Diagrama de implementación			X		1
Diagrama de interacción			X		1
Diagrama de objetos			X		1
Diagrama de paquetes			X		1
Diagrama de tiempo			X		1

TABLA 17. TÉCNICAS PARA VALIDACIÓN DE REQUISITOS DE SOFTWARE EXTRAÍDAS DEL MARCO TEÓRICO

TÉCNICA	(IEEE, 2014)	(Wiegers & Beaty, 2013)	(Ambler, 2014)	(Jacobson, Spence, & Bittner, 2013)	PUNTAJE
Checklist de defectos		X			1
Prototipos	X	X			2
Pruebas de aceptación	X	X			2
Walkthroughs o recorridos		X			1
Demos de iteración			X		1
Pruebas independientes paralelas			X		1
Casos de prueba (pruebas funcionales)			X	X	2
Validación de modelos	X	X			2

7.1.2. Análisis de Técnicas para Especificar y Validar Requisitos extraídas de los Antecedentes

Para establecer las principales técnicas de especificación y validación de requisitos de software identificadas desde los antecedentes, se asignaron pesos con valor de (3) a las técnicas propuestas por los autores que tendrán un aporte directo al procedimiento; (2) a los que tendrán un aporte indirecto y (1) a los otros antecedentes que eventualmente podrían aportar. En la *Tabla 18* se muestra la clasificación de antecedentes según su aporte al procedimiento y los puntajes que se asignaron a los mismos. La *Tabla 19* y la *20* muestran el análisis realizado para las técnicas de especificación y las técnicas de validación de requisitos (respectivamente):

TABLA 18. CLASIFICACIÓN DE ANTECEDENTES Y ASIGNACIÓN DE PESO

No.	ANTECEDENTE	INCIDENCIA SOBRE EL PROYECTO	PESO
1	(De la Cruz Londoño & Castro Guevara, 2014)	Directa	3
2	(Correa Botero & González Bedoya, 2009)	Indirecta	2
3	(Valencia, 1989)	Indirecta	2

4	(Camacho Zambrano, 2005)	Indirecta	2
5	(Merchan, Urrea, & Rebollar, 2008)	Indirecta	2
6	(Durán Toro & Bernárdez Jiménez, 2001)	Otros	1
7	(Báez & Barba Brunner, 2001)	Otros	1
8	(Sumano López, 2001)	Otros	1
9	(Davyt Dávila, 2001)	Otros	1
10	(Chikh & Alajmi, 2014)	Otros	1
11	(Dorigan & De Barros, 2014)	Otros	1

TABLA 19. TÉCNICAS DE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE EXTRAÍDAS DE LOS ANTECEDENTES

Técnica	1	2	3	4	5	6	7	8	9	10	11	Puntaje
Documento de la definición del sistema		X										2
Documento de Especificación de requisitos del sistema		X										2
Documento de Especificación de requisitos del software	X	X			X				X			8
Diagramas de casos de uso		X			X	X			X	X		7
Historias de usuario	X				X							5
Documento de priorización de requisitos					X							2
Especificación en lenguaje Z	X											3
Diagrama de flujo de datos	X		X									5
Modelo de objetos	X											3
Diagrama de secuencia	X											3
Modelo con redes Petri	X											3
Prototipos	X				X				X			6
Diagrama jerárquico o de bloques			X									2
Uso de Herramientas formales o semi-formales							X					1
Guiones y Diálogos para la elaboración de casos de uso								X				1
Diagrama Entidad-Relación										X		1
Modelo de clases conceptual									X			1
Diagramas de actividad									X			1
Representación XML del documento de SRS										X		1
DTD language (parte textual del modelado)										X		1
Lenguaje natural detallado											X	1
Procesamiento del lenguaje natural (PLN) a través de software											X	1

TABLA 20. TÉCNICAS DE VALIDACIÓN DE REQUISITOS DE SOFTWARE EXTRAÍDAS DE LOS ANTECEDENTES

Técnica	1	2	3	4	5	6	7	8	9	10	11	Puntaje
Prototipado	X	X		X	X			X	X			11
Modelo de validación		X										2
Walkthroughs o recorridos	X	X										6
Tablas para validación de requisitos y evaluación de su impacto		X										2
Técnica de Grupo Nominal (TGN)								X				1

Reunión de Reflexión y Diseño							X				1
Análisis de escritura del lenguaje natural									X		1

7.1.3. Cruce entre Técnicas del Marco Teórico y los Antecedentes

Una vez se consigue una clasificación de acuerdo a un peso asignado tanto a las técnicas de especificación y validación de requisitos desde los autores del marco teórico como también desde los antecedentes, se procede a realizar una tabla que consolida y cruza los pesos de cada una de las técnicas, con el objetivo de determinar las técnicas **Requeridas** que serán aquellas que tengan un peso que se ubique en el rango No.1; las técnicas **Recomendadas** serán aquellas que tengan un peso comprendido en el rango No.2 y las **Opcionales** serán las que se ubiquen en el último rango, La *Tabla No.22* muestra el análisis realizado y clasifica de acuerdo a colores para mejor visibilidad, aquellas que obtuvieron mayores puntajes (rojo y verde).

TABLA 21. DEFINICIÓN DE CUBRIMIENTO SEGÚN RANGO

No.	CUBRIMIENTO	RANGO
1	Requerida	9 <= Puntaje <= 13
2	Recomendada	5 <= Puntaje <= 8
3	Opcional	1 <= Puntaje <= 4

TABLA 22. TABLA DE CRUCE DE INFORMACIÓN PARA ANÁLISIS DE TÉCNICAS DE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE

TÉCNICA	MARCO TEÓRICO (Puntaje)	ANTECEDENTES (Puntaje)	Total	Cubrimiento
Documento de la definición del sistema	2	2	4	Opcional
Documento de Especificación de requisitos del sistema	2	2	4	Opcional
Documento de Especificación de requisitos del software	2	8	10	Requerida
Diagrama de flujo de datos (DFD)	3	5	8	Recomendada
Árbol de características	1		1	Opcional
Diagramas de transición de estados (STD)	1		1	Opcional
Diagramas entidad-relación (ERD)	2	1	3	Opcional
Mapas de diálogo	1		1	Opcional
Tablas de decisión	1		1	Opcional
Árboles de decisión	1		1	Opcional
Tablas de eventos de respuesta	1		1	Opcional
Diagramas de casos de uso	3	7	10	Requerida
Diagramas de actividad	3	1	4	Opcional
Diccionario de datos	1		1	Opcional
Prototipos	2	6	8	Recomendada
Historias de usuario	2	5	7	Recomendada
Diagrama de clases	1	1	2	Opcional
Diagrama CRC (Class Responsibility Collaborator)	1		1	Opcional
Diagramas de secuencia	1	3	4	Opcional
Diagrama de estados	1		1	Opcional
Escenarios de uso	2		2	Opcional
Dividir los casos de uso	1		1	Opcional
Notas adhesivas	2		2	Opcional
Esquema de priorización MoSCoW	1		1	Opcional

Diagrama de comunicación	1		1	Opcional
Diagrama de componentes	1		1	Opcional
Diagrama de estructura compuesto	1		1	Opcional
Diagrama de implementación	1		1	Opcional
Diagrama de interacción	1		1	Opcional
Diagrama de objetos	1	3	4	Opcional
Diagrama de paquetes	1		1	Opcional
Diagrama de tiempo	1		1	Opcional
Documento de priorización de requisitos		2	2	Opcional
Especificación en lenguaje Z		3	3	Opcional
Modelo con redes Petri		3	3	Opcional
Diagrama jerárquico o de bloques		2	2	Opcional
Uso de Herramientas formales o semi-formales		1	1	Opcional
Guiones y Diálogos para elaboración de casos de uso		1	1	Opcional
Representación XML del documento de SRS		1	1	Opcional
DTD language (parte textual del modelado)		1	1	Opcional
Lenguaje natural detallado		1	1	Opcional
Procesamiento del lenguaje natural (PLN) a través de software		1	1	Opcional

Una vez analizado el cubrimiento de las técnicas aportadas por los autores del marco teórico y las obtenidas de los antecedentes, se puede decir a manera de resumen, que las técnicas requeridas para la especificación de requisitos de software van a ser las siguientes:

- Documento de Especificación de requisitos del software (SRS)
- Diagramas de casos de uso

De igual forma, se determina que las técnicas recomendadas para la especificación de requisitos de software serán las presentadas a continuación:

- Diagrama de flujo de datos (DFD)
- Prototipos
- Historias de usuario

Las demás técnicas serán opcionales y eventualmente podrían aportar al desarrollo de alguna de las diferentes actividades propuestas en el procedimiento.

El aporte específico de los antecedentes a este análisis, principalmente del trabajo de (De la Cruz Londoño & Castro Guevara, 2014), tiene que ver con la manera en que las empresas de desarrollo de software de la región están llevando a cabo la adquisición y gestión de requisitos, indicando las principales técnicas y metodologías usadas para abordarlos, siendo apoyados por los autores del marco teórico. De igual forma, aportaron formatos, métodos y herramientas para llevar a cabo los procesos de especificación y validación de requisitos que podrán ser recomendadas por el procedimiento para facilitar su utilización. Las mismas fueron complementadas por los trabajos de (Valencia, 1989), (Merchan, Urrea, & Rebollar, 2008), (Davyt Dávila, 2001), (Correa Botero & González Bedoya, 2009), (Durán Toro & Bernárdez Jiménez, 2001) y (Sumano López, 2001).

De manera similar al análisis realizado para identificar las técnicas de especificación, se consolidó la información de las técnicas de validación identificadas en el marco teórico y en los antecedentes, como se observa en la siguiente tabla.

TABLA 23. TABLA DE CRUCE DE INFORMACIÓN PARA ANÁLISIS DE TÉCNICAS DE VALIDACIÓN DE REQUISITOS DE SOFTWARE

TÉCNICA	MARCO TEÓRICO (Puntaje)	ANTECEDENTES (Puntaje)	Total	Cubrimiento
Checklist de defectos	1		1	Opcional
Prototipos	2	11	13	Requerida
Pruebas de aceptación	2		2	Opcional
Demos de iteración	1		1	Opcional
Pruebas independientes paralelas	1		1	Opcional
Casos de prueba	2		2	Opcional
Validación de modelos	2		2	Opcional
Modelo de validación		2	2	Opcional
Tablas para validación de requisitos y evaluación de su impacto		2	2	Opcional
Técnica de Grupo Nominal (TGN)		1	1	Opcional
Reunión de Reflexión y Diseño		1	1	Opcional
Análisis de escritura del lenguaje natural		1	1	Opcional
<i>Walkthroughs</i> o recorridos guiados	1	6	7	Recomendada

Respecto a las técnicas para la validación de requisitos de software propuestas por los autores del marco teórico y las obtenidas de los antecedentes, se puede concluir que, las técnicas que serán requeridas por el procedimiento serán las siguientes:

- Prototipos

En la misma línea, se define que, de acuerdo al análisis realizado, aquellas técnicas que serán recomendadas para la validación de requisitos de software deberán ser las siguientes:

- *Walkthrough* o recorridos guiados

Con la intención de argumentar la pertinencia de las técnicas seleccionadas de especificación y validación de requisitos en el procedimiento, se identifican a continuación las principales ventajas y desventajas de las mismas:

TABLA 24. VENTAJAS Y DESVENTAJAS DE LAS TÉCNICAS SELECCIONADAS

TÉCNICA	VENTAJAS	DESVENTAJAS
Documento SRS	Permite lograr especificaciones para un producto del software en particular, un programa o juego de programas que realizan ciertas funciones en un ambiente específico.	Se presta para describir cualquier plan o detalles de aplicación, los cuales deben ser descritos en la fase de diseño.
	Puede escribirse por uno o más representantes del proveedor, uno o más representantes del cliente, o por ambos.	Al ser escrita en lenguaje natural, la SRS puede ser ambigua, pues el idioma natural es inherentemente ambiguo.

	Existen plantillas y herramientas que facilitan su elaboración.	Puede necesitar evolucionar así como el desarrollo de las actualizaciones del producto de software, puesto que resulta ser imposible de especificar cada detalle en el momento que el proyecto se inicia, razón por la cual se generan cambios adicionales, deficiencias, limitaciones e inexactitudes en el SRS.
	Ampliamente descrito por el estándar IEEE 830 reconocido mundialmente.	
Diagramas de casos de uso	Describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso en diferentes escenarios.	En sistemas grandes toman mucho tiempo para definir todos los casos de uso.
	Es una técnica diseñada para especificar el comportamiento de un sistema.	El análisis de calidad depende de cómo se haya realizado la descripción inicial del caso de uso.
	La mayoría de los requisitos funcionales, sino todos, se pueden expresar con casos de uso.	No permiten establecer los requisitos no funcionales.
	Pueden ser representados mediante notación UML, por lo que el lenguaje que utilizan es común y entendible para el usuario.	Los casos de uso deben complementarse con información adicional como: Reglas de negocio y Diccionario de datos que complementen los requisitos del sistema.
	Facilitan la comunicación con el cliente y hace que el analista se centre en las necesidades del usuario.	
	Existen plantillas y herramientas que facilitan su elaboración.	
Diagramas de flujo de datos (DFD)	Favorecen la comprensión del proceso al mostrarlo como un dibujo.	Los diagramas complejos y detallados suelen ser laboriosos en su planteamiento y diseño.
	Permiten identificar los problemas y las oportunidades de mejora del proceso.	No requiere de conocimiento técnico, pero su construcción depende de que el analista sea muy ordenado.
	Son una buena herramienta para capacitar a los nuevos empleados y también a los que desarrollan la tarea, cuando se realizan mejoras en el proceso.	
	Permite que cada actor del sistema se sitúe dentro del proceso, identificando sus roles y cadena de relaciones, por lo que se mejora considerablemente la comunicación entre las personas.	
Prototipos	Permiten el desarrollo de un sistema a partir de requisitos poco claros o cambiantes.	Si los requisitos no están demasiado claros los prototipos pueden conllevar a un desarrollo no eficaz del sistema final.

	Pueden ser usados tanto para identificar requisitos como para validarlos una vez especificados.	Si bien el prototipo permite capturar la idea, el usuario no percibe como será realmente el diálogo hombre-máquina, sobre todo si existe un requisito no funcional como el de usabilidad.
	Permiten conseguir una importante retroalimentación en cuanto a si el sistema diseñado con base en los requisitos identificados le permite al usuario realizar su trabajo de manera eficiente y efectiva.	Puede presentar problemas de incompletitud, lo que puede confundir a los usuarios haciéndolos pensar que el producto final quedará incompleto tal como el prototipo.
	El usuario participa más activamente en la construcción del producto de software (la solución), ya que lo puede “ver” y, dependiendo del tipo de prototipo, “utilizar” desde el primer momento.	Los prototipos pueden generar otro tipo de problemas si su presentación y discusión con los usuarios no es controlada: puesto que son modelos inconclusos, los usuarios suelen enfocarse en aspectos “superficiales” del prototipo que los pueden dejar inconformes luego de verlos por primera vez.
	Permiten amplia flexibilidad para el manejo de las interfaces de usuario.	Requiere participación activa del usuario, al menos, para evaluar el prototipo y mucho más involucramiento si se quiere que participe en su creación.
	Se requiere poco tiempo para generar un nuevo prototipo (maqueta).	
	Se requiere poco esfuerzo para aplicar los cambios.	
	Se reduce el riesgo o la incertidumbre sobre la implementación del software.	
	Proporciona al usuario un mayor conocimiento del sistema con una curva menor de aprendizaje.	
Historias de usuario	Permiten expresar los requisitos ya que expresan el problema que el sistema o producto software debe resolver.	En proyectos grandes y complejos resulta difícil saber por dónde comenzar a identificar las historias de usuario.
	Son un enfoque de requisitos ágil que se focaliza en establecer conversaciones acerca de las necesidades de los clientes.	Facilitan incluir características de la interfaz de usuario en la descripción de la historia de usuario, lo cual se debe evitar pues hace parte del diseño.
	Presentan descripciones cortas y simples de las funcionalidades del sistema, narradas desde la perspectiva de la persona que desea dicha funcionalidad, usualmente un usuario.	
	Pueden ser escritas en diferentes niveles de detalle, por lo que es posible escribir historias que cubren múltiples funcionalidades.	
	Permite la participación activa de los usuarios en las discusiones de tal manera que aporten a un mayor detalle sobre sus necesidades.	
Walkthrough o recorridos guiados	Promueven la creación de software fácil de leer y mantener.	Se puede caer en disputas basadas en estilos propios de los analistas o

programadores.	
Permiten la captura de errores en etapas tempranas.	Requiere que el autor sea quien modere las reuniones guiadas, por lo cual tiene gran influencia sobre los resultados.
Los <i>walkthrough</i> de estándares realizan una inspección exhaustiva de la interfaz para comprobar que cumple en todo momento y globalmente todos los puntos definidos en un estándar establecido.	Los puntos críticos no son abordados en profundidad.
Los <i>walkthrough</i> de características analizan únicamente un conjunto de características determinadas del producto, proporcionándole al usuario escenarios para el resultado final a obtener del uso del producto.	

A manera de conclusión, se puede expresar que el procedimiento propuesto hará uso de aquellas técnicas de especificación y validación de requisitos de software que fueron más representativas para los autores del marco teórico y los antecedentes, presentándolas con un carácter de exigencia alto (técnicas requeridas). De igual forma, se tendrá una colección de técnicas que serán sugeridas por el procedimiento pero que no necesariamente serán de obligatoria utilización (técnicas recomendadas), las cuales serán muy importantes en algunos casos, pero no indispensables. Finalmente, las otras técnicas de carácter opcional deberán estar disponibles para su uso cuando tengan algún aporte significativo al desarrollo de las actividades propuestas en el procedimiento, pero en todo caso su implementación estará bajo el criterio del personal encargado de las diferentes actividades.

7.2. Diagnóstico de las Empresas

Para caracterizar la manera como las MiPymes desarrolladoras de software de la ciudad de Pereira están llevando a cabo la especificación y validación de requisitos (***Etapas I-Exploratoria***), se diseñó un instrumento (**Ver ANEXO A - Encuesta a empresas**) que se aplicó a la muestra por conveniencia de las empresas desarrolladoras de software de la ciudad de Pereira, el mismo fue estructurado bajo los siguientes temas de interés:

TABLA 25. CLASIFICACIÓN DE PREGUNTAS POR TEMAS

No.	TEMA	PREGUNTA
1	Formación profesional del personal encargado del desarrollo de los requisitos.	1, 2
2	Planificación, participación, capacidad y experiencia de los <i>stakeholders</i> en la captura de requisitos.	4, 5, 6, 7, 8, 9, 11, 12, 13, 14
3	Documentación de los procesos de negocio de las organizaciones llevada a cabo por el personal encargado del desarrollo de los requisitos.	3, 10
4	Uso de metodologías para el desarrollo de requisitos.	16
5	Especificación de Requisitos de software.	15, 17, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31
6	Validación de requisitos.	32, 33, 34, 35

7	Gestión de requisitos de software.	18
---	------------------------------------	----

Es importante resaltar que no todos los temas citados anteriormente tienen el mismo grado de interés en la investigación que se adelanta, pero con el ánimo de hacer una caracterización completa, se formularon ciertas preguntas para cada tema específico. Así mismo, se debe mencionar que la encuesta realizada a las empresas fue a nivel exploratorio, para la cual no se hizo un análisis de correlación sino solo un análisis de variables individuales. A continuación, se detallan la ficha técnica de la encuesta utilizada para la recolección de información:

TABLA 26. FICHA TÉCNICA DE LA ENCUESTA

Diseño muestra	Muestra por convención.
Población objetivo	47 Micro, pequeñas y medianas empresas de la ciudad de Pereira dedicadas al desarrollo de software.
Técnica	Encuesta.
Tamaño de la muestra	17 empresas de la ciudad de Pereira
Momento estadístico	Noviembre 2015 – Diciembre 2015
Financiación	Recursos propios

7.2.1. Análisis de resultados de la encuesta aplicada a las empresas

El análisis presentado a continuación está constituido por las respuestas aportadas por las empresas a aquellas preguntas cerradas que permitiera realizar una tabulación consistente, haciendo claridad de que las preguntas abiertas, más reducidas en número que las cerradas, no se presentan a continuación pero sí fueron tenidas en cuenta para analizar el proceso llevado a cabo por cada una de las organizaciones encuestadas para especificar y validar requisitos, así como para el diseño del procedimiento. Para conocer de manera consolidada los resultados de la encuesta realizada a cada una de las empresas, ver **ANEXO B – Resultados encuesta**.

A continuación, se presenta un breve análisis de los temas en que se agruparon las preguntas de acuerdo a los resultados contenidos en el mencionado anexo.

Concluyendo sobre **la formación profesional del personal encargado del desarrollo de los requisitos**, se puede evidenciar que este proceso es generalmente llevado a cabo por un ingeniero de sistemas que no está específicamente formado en el área de requisitos de software.

Respecto a la **planificación, participación, capacidad y experiencia de los stakeholders en la captura de requisitos**, el estudio llevado a cabo permite concluir que a pesar de que las empresas realizan un diagnóstico para determinar quiénes serán las personas involucradas en el proyecto y socializan con ellas el proceso de desarrollo de los requisitos que se va a llevar a cabo, no se evalúa el grado de participación de los *stakeholders* en el proyecto ni el grado de dominio que poseen sobre el problema y sobre la solución deseada. Adicionalmente, no se determina si el cliente o usuario final ha participado de algún proceso relacionado con requisitos, tiene la suficiente capacidad para comunicar las necesidades que quiere satisfacer o la experiencia que posee respecto al proceso llevado a cabo en la empresa.

Con relación a la **documentación de los procesos de negocio de las organizaciones llevada a cabo por el personal encargado del desarrollo de los requisitos**, se concluye que antes de

llevar a cabo la captura de requisitos, la mayoría de los profesionales se documentan sobre los procesos de negocio de la empresa donde se va a implementar la solución, pero no establecen una relación entre los requisitos del negocio y los requisitos del usuario.

En lo referido al **uso de metodologías para el desarrollo de requisitos** tales como modelos, metodologías, normas o procesos para llevar a cabo el desarrollo de los requisitos, el 59% de ellas manifiesta que no lo hace, el 35% indica que sí y el 6% restante no se pronuncia al respecto.

Se concluye también respecto a la **especificación de requisitos de software** que las empresas utilizan formatos o plantillas diligenciados en lenguaje natural, en el que se identifican generalmente los requisitos de usuario y del sistema, las restricciones, suposiciones y dependencias. Sin embargo, en la mayoría de las empresas el documento SRS resultante no cumple con todas las características mencionadas por (Wiegers & Beaty, 2013) para una buena especificación de requisitos.

De la **validación de requisitos**, se pudo evidenciar que la mayoría de las empresas llevan a cabo actividades tendientes a la validación del documento SRS elaborado haciendo uso principalmente de formatos y herramientas. Sin embargo, resulta preocupante conocer que solo un 59% de las mismas involucran al usuario en este proceso.

Con relación al proceso de **gestión de requisitos** se puede indicar que “Asignar categorías a los requisitos en función de su criticidad, importancia, complejidad y riesgo”; así como “Asignar un responsable al desarrollo de cada requisito” son las actividades más realizadas, seguidas por “Almacenar los requisitos en un repositorio central” y “Realizar chequeos a las actividades designadas a los integrantes del grupo de desarrollo”.

7.2.2. Uso de técnicas por parte de las empresas

Una vez analizada la información de las encuestas aplicadas a las MiPymes desarrolladoras de software de la ciudad de Pereira que conforman la muestra y, teniendo identificadas las técnicas de especificación y validación de requisitos de software desde el marco teórico y los antecedentes, se relaciona en las siguientes tablas (*Tabla 27* y *Tabla 28*) el uso de dichas técnicas por parte de las empresas con la intención de determinar el porcentaje de utilización de las mismas, de tal manera que a partir de la información obtenida se puedan analizar las debilidades o falencias que tiene la industria del software local en este respecto y proponer desde el procedimiento las que serán *requeridas* (color rojo) y *recomendadas* (color verde). De igual forma, este análisis permitirá determinar el grado de conocimiento y experiencia que tienen las empresas sobre el uso de dichas técnicas y su importancia en los procesos de especificación y validación de requisitos de software.

Tabla 27. TÉCNICAS DE ESPECIFICACIÓN DE REQUISITOS DE SOFTWARE SEGÚN INFORMACIÓN DE LAS EMPRESAS

TÉCNICA	EMPRESAS																	% uso
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Documento de Especificación de requisitos del software	X		X	X	X				X	X		X	X		X	X	X	65%

Diagramas de casos de uso			X			X		X											24%
Diagrama de flujo de datos (DFD)																			0%
Prototipos	X	X	X	X	X	X	X									X			47%
Historias de usuario							X				X		X						18%

TABLA 28. TÉCNICAS DE VALIDACIÓN DE REQUISITOS DE SOFTWARE SEGÚN INFORMACIÓN OBTENIDA DE LAS EMPRESAS

TÉCNICA	EMPRESAS																	% uso
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
Prototipos	X		X	X	X	X						X						35%
Walkthroughs o recorridos		X			X			X		X					X		X	35%

Una vez realizado el análisis de las técnicas que, según el resultado de las encuestas usan las MiPymes desarrolladoras de software de la ciudad de Pereira según la muestra, se encuentra que la mayoría de las técnicas son conocidas y en varios casos utilizadas para llevar a cabo los procesos de especificación y validación de requisitos de software. Sin embargo, se nota claramente que solo 2 de las 17 empresas lo que equivale a un 12% aproximadamente hacen uso de las técnicas de especificación *requeridas* y el resto hace uso solo de alguna de ellas. Por el contrario, para las técnicas *recomendadas* se encuentra que solo 1 de las 17 empresas emplean al menos 2 de las 3 técnicas mencionadas, mientras que las demás solo emplean 1 técnica o no usan ninguna.

Con respecto a las técnicas para la validación de requisitos con carácter de *requeridas*, se encuentra que el 35% de las empresas las usa, mientras que las técnicas *recomendadas* son usadas por el 35% de las mismas, lo que equivale a 6 de las 17 empresas.

Lo anterior permite inferir que es necesario motivar la implementación de estas técnicas para facilitar el desarrollo de los requisitos y contribuir a la calidad de los mismos. Sin embargo, las técnicas empleadas por el procedimiento no necesariamente deberán tener el mismo criterio de cubrimiento obtenido en el análisis anterior (requerida, recomendada, opcional), pues el mismo procedimiento podrá proponer un diseño con criterios propios sin dejar de lado estos resultados.

7.3. Diseño del procedimiento para la Especificación y Validación de requisitos de software

El diseño del procedimiento para la especificación y validación de requisitos de software que hace parte de la (*Etapas 1 – Descriptiva*) de la metodología, se llevó a cabo con diagramas de actividades soportados con una descripción textual explícita, contando además con un conjunto de guías y recomendaciones. Así mismo, se modeló bajo SPEM para una mejor presentación y despliegue. A este procedimiento se le dio el nombre de “PEVReS”, (Procedimiento para la Especificación y Validación de Requisitos de Software).

Es importante mencionar que el mismo fue diseñado teniendo en cuenta técnicas y buenas prácticas adquiridas en el Curso de *Ingeniería de Requisitos* tomado por los autores en el marco del convenio No. 223 de 2015, celebrado entre en SENA y la Federación Colombiana de la Industria de Software y Tecnologías Informáticas Relacionadas (FEDESOFTE) y orientado por

instructores certificados por el IREB⁶ (*International Requirements Engineering Board*), consejo internacional de expertos en Ingeniería de Requisitos para la certificación individual de los profesionales en dicha disciplina. Algunas de estas técnicas adoptadas por PEVReS y analizadas también en el Capítulo 7. Resultados fueron las siguientes:

- Especificación de requisitos en lenguaje natural
- Esquema de clasificación de requisitos
- Perspectivas sobre requisitos
- Diagramas de Casos de uso
- Diagrama Entidad-Relación
- Diagrama de Clases
- Diagrama de Actividades
- Diagramas de Flujo de Datos
- Diagrama de Estados
- Prototipos
- Revisiones guiadas (Walkthroughs)
- Inspecciones
- Listas de comprobación (Checklist)

Con relación al principal referente del proyecto, la metodología propuesta por (De la Cruz Londoño & Castro Guevara, 2014), es importante indicar que el procedimiento PEVReS elaborado impacta directamente una de las áreas propuestas por dicha metodología, como se muestra en la *Ilustración 15*, resaltando el mismo especifica en detalle las actividades y pasos requeridos para llevar a cabo la especificación y validación de requisitos.

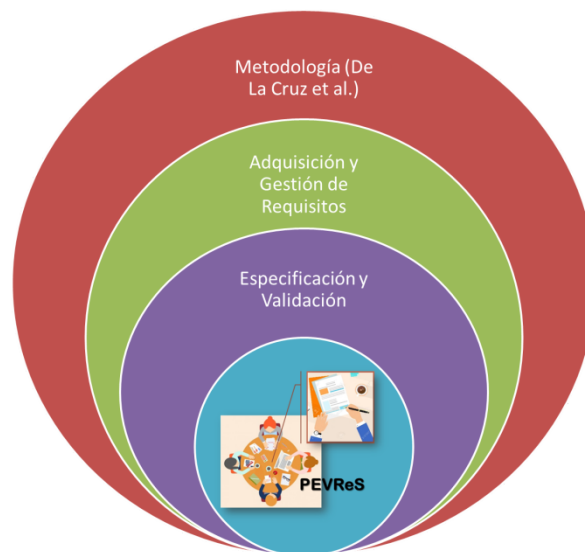


ILUSTRACIÓN 15. PEVReS FRENTE A METODOLOGÍA PROPUESTA POR (DE LA CRUZ LONDOÑO & CASTRO GUEVARA, 2014)

⁶ <https://www.ireb.org/en>

A continuación, se relaciona el procedimiento elaborado, sobre el cual hizo la respectiva validación a través de dos pilotos de aplicación. Es importante aclarar que para el procedimiento, los requisitos de software (funcionales y no funcionales) se derivan de los requisitos del sistema, entendiendo que el sistema tiene componentes de software, tal como se manifiesta en (IEEE, 2014, pág. 34). Para consultar en detalle el procedimiento elaborado una vez realizados los ajustes y mejoras resultantes de dicha validación, ver **ANEXO C – Procedimiento para especificar requisitos de software**.

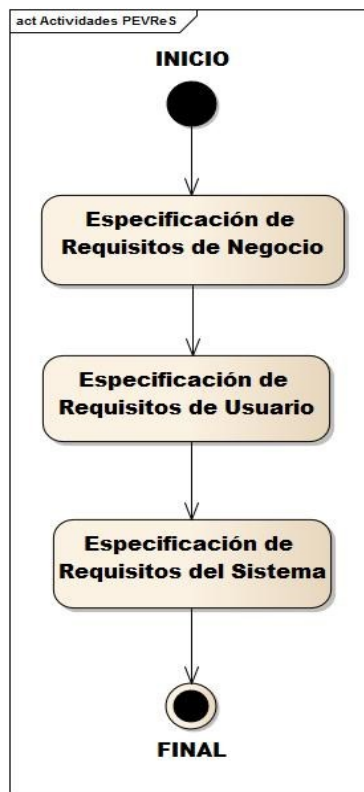


ILUSTRACIÓN 16. ACTIVIDADES QUE COMPONEN EL PROCEDIMIENTO DE ESPECIFICACIÓN DE REQUISITOS

Para la especificación del procedimiento se usó el formato presentado en la estrategia metodológica (capítulo 6 de este documento). En la primera parte del procedimiento se encuentran el título, la descripción, el responsable, las precondiciones y las entradas, como se ilustra a continuación.



Título:	PROCEDIMIENTO PARA ESPECIFICAR REQUISITOS DE SOFTWARE EN MIPYMES DESARROLLADORAS DE SOFTWARE DE LA CIUDAD DE PEREIRA	
Descripción:	<i>Su objetivo principal es lograr la definición del sistema software a desarrollar, tomando como punto de partida el contexto general del sistema (visión) y las necesidades de los involucrados en el proyecto</i>	
Responsable:	Líder del proceso de requisitos.	
Precondiciones:	No.	
	1	Haber realizado la planificación de la fase de especificación de requisitos de software.
	2	Haber realizado actividades de elicitación de requisitos.
	3	Haber realizado actividades de análisis (clasificación y priorización de requisitos).
	4	Haber adquirido un dominio total o parcial del problema y de la solución a desarrollar.
Entradas:	No.	
	1	Los requerimientos obtenidos del proceso de elicitación, clasificados y priorizados en el proceso de análisis.
	2	Registro de los interesados y su relación con los requerimientos.
	3	El acta de constitución del proyecto o <i>Project Charter (opcional)</i> .
	4	Procesos y procedimientos relacionados con el caso de negocio para el cual se va a implementar el sistema.
	5	Objetivos de negocio, Reglas de negocio y Restricciones identificadas.

Posteriormente, el procedimiento presenta tres actividades y para cada una de ellas las técnicas que permiten llevar a cabo la actividad, así como el cubrimiento de la técnica (requerida, recomendada, opcional), su responsable, los pasos para ejecutar la técnica y finalmente, los formatos, guías y herramientas que le sirven de apoyo. La primera actividad, relacionada con el entendimiento del negocio, se muestra a continuación.

No.	Actividad	No.	Técnica		Responsables	No.	Tareas (pasos)	Formatos, Guías y Herramientas
			Perspectiva funcional	Cubrimiento de la técnica				
1	Entendimiento del Negocio <i>Para realizar esta actividad tenga en cuenta las entradas especificadas para este procedimiento</i>	1.1	Diagrama de flujo de datos - DFD	Recomendada	Analista/Proveedor de información	1	Definir la notación de los símbolos descriptivos que serán utilizados en el Diagrama de Flujo de Datos.	Guía para la elaboración de DFD
						2	Realizar el diagrama de contexto del sistema a desarrollar, teniendo en cuenta las entradas de este procedimiento.	
						3	Identificar los procesos que conforman el sistema a desarrollar, teniendo en cuenta el diagrama realizado en el paso 2.	
						4	Identificar los flujos de datos, almacenes de datos, entradas y salidas que eslabonan los procesos identificados en el paso No.3.	
						5	Realizar el diagrama de flujo de datos (de Nivel 1) donde se relaciona lo que se identificó en los pasos No.3 y No 4.	
						6	Realizar un diagrama de flujo de datos (de Nivel 2) para expandir los procesos relacionados en el diagrama de nivel 1, dando mayor detalle (Opcional) .	
						7	Detallar por medio de un diagrama de flujo las restricciones y condiciones que existen al interior de los procesos críticos identificados en los pasos No.5 y 6 (Opcional) .	
						8	Realizar el Diccionario de Datos detallado de los diagramas realizados en los Pasos 5 y 6 .	
						9	Evaluar los Diagramas de Flujos de Datos y Diagrama de Flujo elaborados.	
		1.2	Diagrama de actividades	Opcional	Analista	A criterio del analista	Ejemplo Diagrama de Actividades	
		Perspectiva comportamiento		Cubrimiento de la técnica				
		1.3	Diagrama de estados	Opcional	Analista		Ejemplo Diagrama de Estados	
		1.4	Diagrama de secuencia	Opcional	Analista		Ejemplo Diagrama de Secuencia	
		Perspectiva datos		Cubrimiento de la técnica				
		1.5	Modelo Conceptual	Requerida	Analista	1	Definir y delimitar el alcance del dominio.	Ejemplo Modelo Conceptual
						2	Elaborar un glosario.	
						3	Identificar conceptos o entidades dentro del glosario.	
4	Definir relaciones entre conceptos.							
5	Elaborar el modelo conceptual.							
1.6	Diagrama entidad-relación (ERD)	Opcional	Analista	A criterio del analista	Ejemplo Diagrama Entidad-Relación			
1.7	Diagrama de clases	Opcional	Analista		Ejemplo Diagrama de Clases			

La segunda actividad relacionada con la Especificación de requisitos de usuario se presenta a continuación.

Actividades-Técnicas-Tareas	2	Especificación de Requisitos de Usuario <i>Para realizar la especificación de requisitos de usuario tenga en cuenta la especificación de requisitos de negocio realizada</i>	2.1	Diagramas de casos de uso	Recomendada (Aplicar esta técnica cuando se requiere mayor nivel de detalle y control sobre los requisitos)	Analista	1	Elaborar una breve descripción en la que se indique la funcionalidad del sistema, de acuerdo a su interacción con el exterior.	
							2	Identificar los actores del sistema (aquellos que interactúan con el mismo)	
							3	Identificar los casos de uso (función completa o secuencia de acciones donde el actor y el sistema interactúan)	
							4	Definir los casos de uso primarios	
							5	Establecer las relaciones (interacción) entre actores del paso 2 y los casos de uso identificados en el paso 3	
							6	Realizar el diagrama de casos de uso.	
							7	Clasificar los casos de uso (Opcional). Se propone usar las siguientes clasificaciones: - Núcleo del negocio (Funciones principales que debe ofrecer el sistema). - CRUD (Funciones de creación, consulta, actualización y eliminación de información). - Reportes (Presentación de información consolidada). - Soporte (Funciones adicionales para el sistema).	
							8	Determinar las precondiciones (Condiciones que deben cumplirse antes de ejecutar el caso de uso)	
							9	Establecer el flujo básico o normal de eventos	
							10	Establecer los flujos alternos de eventos, de acuerdo a los escenarios claves (una situación posible de interacción con el sistema)	
							11	Determinar las excepciones de los flujos establecidos en los pasos 9 y 10	Formato de Especificación de Casos de Uso
							12	Determinar las post-condiciones (Condiciones que deben cumplirse después de ejecutar el caso de uso)	
							13	Liste los requisitos No funcionales relacionados con el caso de uso	
							14	Establecer los objetivos relacionados con el caso de uso (Objetivos y/o características del negocio con los cuales se relaciona este caso de uso)	
							15	Evaluar la Especificación del Caso de Uso realizada	Checklist para verificar la Especificación de Casos de Uso
2.2	Historias de usuario	Recomendada (Aplicar esta técnica cuando se requiere mayor agilidad en el desarrollo del proyecto)	Analista/Proveedor de Información	1	Definir el formato que se utilizará para establecer la historia de usuario.	Formato de Historia de Usuario			
				2	Definir el Como en la Historia de Usuario (rol del usuario que hará uso de la funcionalidad).				
				3	Definir el Que en la Historia de Usuario (qué hace el usuario en la historia).				
				4	Definir el Para en la Historia de Usuario (el propósito de la historia, la meta que quiere alcanzar el usuario al ejecutar la historia).				
				5	Confirmar la Historia de Usuario mediante Prueba de Aceptación por parte del usuario.		Checklist para verificación de Historia de Usuario INVEST		
2.3	Prototipos	Requerida	Analista/Proveedor de información	1	Identificar una idea general de la solución.	Ejemplo de Prototipo			
				2	Identificar los requisitos conocidos profundizando en los aspectos donde es necesario más definición.				
				3	Realizar el diseño básico del prototipo inicial con base en los Puntos 1 y 2.				
				4	Diseñar y construir el prototipo detallado con énfasis en la interface de usuario y la máxima funcionalidad (Diseño técnico). Tomar como base el Punto 3				
				5	Verificar el prototipo con el cliente e identificar otros requisitos (Evaluación).		Checklist para verificación de Prototipos		
				6	Realizar modificaciones y ajustes al prototipo de acuerdo al Paso 5.				
				7	Realizar la documentación necesaria para su implementación.				

La tercera actividad hace referencia a Especificación de requisitos del sistema y al final se muestran las salidas del procedimiento en las que se destaca el Documento de especificación de requisitos (SRS), ver ANEXO D – Formato documento SRS.

	3	Especificación de Requisitos del Sistema <i>Para realizar la especificación de requisitos del sistema tenga en cuenta las técnicas aplicadas en la especificación de requisitos de usuario.</i>	3.1	Especificación en Lenguaje Natural de requisitos No Funcionales	Requerida	Analista	<ol style="list-style-type: none"> Determinar el grado de obligación del requisito. Definir el núcleo o verbo de proceso principal que especifica el requisito. Caracterizar la actividad del sistema con relación al verbo de proceso principal identificado en el Paso 7: <ul style="list-style-type: none"> - Actividad independiente del sistema (sin interacción del usuario). - Interacción del usuario (provee una funcionalidad). - Requisito de interface (dependencia de otros sistemas). Definir los objetos necesarios para complementar el verbo de proceso principal. Determinar las condiciones del sistema bajo las se ejecuta la funcionalidad. Incluir detalles adicionales del objeto. Adicionar un criterio medible para determinar el cumplimiento (numérico y finito). 	Formato para documentar requisitos en lenguaje natural
			3.2	Especificación en Lenguaje Natural de requisitos Funcionales	Recomendada	Analista	<p>Para realizar los siguientes pasos tenga en cuenta los Casos de uso, las Historias de usuario o los Prototipos realizados:</p> <ol style="list-style-type: none"> Determinar el grado de obligación del requisito. Definir el núcleo o verbo de proceso principal que especifica el requisito. Caracterizar la actividad del sistema con relación al verbo de proceso principal identificado en el Paso 7: <ul style="list-style-type: none"> - Actividad independiente del sistema (sin interacción del usuario). - Interacción del usuario (provee una funcionalidad). - Requisito de interface (dependencia de otros sistemas). Definir los objetos necesarios para complementar el verbo de proceso principal. Determinar las condiciones del sistema bajo las se ejecuta la funcionalidad. Incluir detalles adicionales del objeto. Adicionar criterios de medición. 	
Salidas:	No.		Salida					
	1	Documento de especificación de requisitos del software (SRS)						Checklist para verificar la Especificación de requisitos de software
	2	Oportunidades de mejora para la operación de los procesos de negocio involucrados en el desarrollo del proyecto.						
	3	Consideraciones especiales respecto al desarrollo del sistema.						

Para la validación de los requisitos de software y el documento SRS se elaboró el procedimiento que se presenta a continuación, el cual fue objeto de la respectiva validación a través de los pilotos de aplicación. Para consultar en detalle el procedimiento elaborado una vez realizados los ajustes y mejoras resultantes de dicha validación, ver **ANEXO E – Procedimiento para validar requisitos de software**.

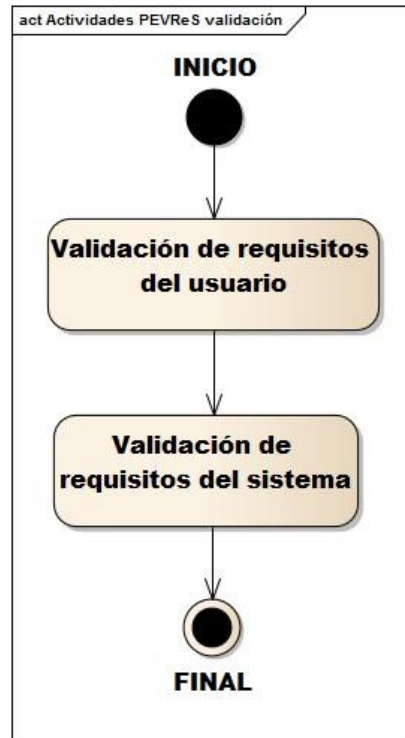


ILUSTRACIÓN 17. ACTIVIDADES QUE COMPONEN EL PROCEDIMIENTO DE VALIDACIÓN DE REQUISITOS

El objetivo principal del procedimiento es validar la especificación de los requisitos de software realizada mediante la aplicación del procedimiento PEVReS, donde los analistas se reúnen con el cliente y/o usuario del sistema para validar los requisitos que se identificaron y se especificaron, es decir, que estas especificaciones reflejan realmente lo que los usuarios necesitan.

Inicialmente, el procedimiento presenta el título, la descripción, el responsable del procedimiento, las precondiciones y las entradas necesarias para llevar a cabo las actividades relacionadas con la validación, tanto de los requisitos especificados como del documento SRS elaborado.



Título:	PROCEDIMIENTO PARA VALIDAR REQUISITOS DE SOFTWARE EN MIPYMES DESARROLLADORAS DE SOFTWARE DE LA CIUDAD DE PEREIRA	
Descripción:	<i>Su objetivo principal es validar la especificación de los requisitos de software realizada mediante la aplicación del Procedimiento PEVReS, donde los Analistas se reúnen con el cliente y/o usuario del sistema para validar los requisitos que se relevaron y se especificaron, es decir, que estas especificaciones reflejan realmente lo que los usuarios necesitan</i>	
Responsable:	Líder del proceso de requisitos.	
Precondiciones:	No.	
	1	Haber realizado la especificación de requisitos de software de acuerdo al Procedimiento PEVReS.
	2	Contar con la presencia de las partes que tienen la autoridad de aprobar la especificación de requisitos de software.
Entradas:	No.	
	1	Documento de especificación de requisitos del software (SRS).
	2	Prototipos elaborados.

Luego se presentan las actividades, técnicas (con su respectivo cubrimiento), responsables y pasos para llevar a cabo cada técnica, así como los formatos, guías y herramientas que apoyan su ejecución. Al final se presentan como salidas el documento SRS validado y el acta de reunión.

	No.	Actividad	No.	Técnica	Responsables	No.	Tareas (pasos)	Formatos, Guías y Herramientas	
				Cubrimiento de la técnica					
Actividades-Técnicas-Tareas	1	Validación de requisitos del usuario	1.1	Prototipos	Requerida	Analista/Proveedor de información	1	Exponga ante el usuario/cliente el prototipo realizado.	
							2	Identifique la tarea que se pretende realizar en el nuevo sistema.	
							3	Simule en el prototipo los pasos requeridos para llevar a cabo la tarea identificada en el Paso No.2 utilizando herramientas de prototipificación o la técnica Mag de oz.	
							4	Identifique el nivel de dificultad para ejecutar la tarea.	
							5	Identifique si existen errores en el prototipo.	
							6	Diligencie el formato indicado para la validación de prototipos.	Formato para la Validación de Prototipos
							7	Establezca los ajustes pertinente para el prototipo (Opcional).	
		1.2	Casos de prueba	Opcional	Analista/Proveedor de información	A criterio del analista	Formato para especificar casos de prueba		
	2	Validación del documento de especificación de requisitos de software (SRS)	2.1	Walkthroughs o recorridos	Requerida	Analista/Proveedor de información	1	Establecer los stakeholders que deben asistir a la sesión de validación.	
							2	Realice un recorrido detallado por la sección a revisar del documento SRS.	
							3	Determine si comprende el significado de cada requisito revisado.	
							4	Identifique los posibles errores o conflictos encontrados en el documento SRS.	
							5	Sugiera las acciones a realizar para solucionar los errores o conflictos identificados en el Paso No.4	
							6	Diligencia el formato indicado para la validación del documento SRS (Opcional).	Formato para la validación de recorrido
7							Manifieste su acuerdo o desacuerdo con el documento SRS en un acta de reunión.		
	2.2	Elaborar el manual de Usuario apartir de los requisitos	Opcional	Analista/Proveedor de información	A criterio del analista				
Salidas:	No.	Salida							
	1	Documento de especificación de requisitos del software (SRS) validado.							
	2	Acta de reunión de validación de Requisitos.							

7.4. Aplicación del procedimiento propuesto

La validación del procedimiento elaborado hace parte de la ***Etapa 2*** de la metodología del proyecto y se llevó a cabo a través de dos pilotos de aplicación con dos unidades de análisis: dos MiPymes desarrolladoras de software ubicadas en la ciudad de Pereira. A continuación, como parte de la planeación general para ambos pilotos de aplicación se muestran los criterios de selección que se tuvieron en cuenta para elegir los proyectos a analizar en ambas empresas, como también la información que se recolectó durante la ejecución de los pilotos. No se consideraron como criterios otros aspectos como lenguaje, plataforma, líneas de código, uso de *frameworks*, entre otros, que podrían representar variaciones en cada proyecto pues no se consideran relevantes para la etapa de requisitos, dando así mayor flexibilidad a la aplicación del procedimiento, pero por supuesto, dejando abierta la posibilidad a trabajos futuros. También se indica cómo se llevó a cabo la presentación y capacitación sobre el procedimiento al equipo de desarrollo.

TABLA 29. PLANEACIÓN GENERAL DEL PILOTO DE APLICACIÓN

CRITERIOS DE SELECCIÓN DEL PROYECTO	
Empresas a visitar	MiPymes desarrolladoras de software de la ciudad de Pereira.
No. de responsables	Máximo 2 personas encargadas del desarrollo de los requisitos.
Tipo de proyecto a intervenir	Marco tradicional de los datos (transaccionales).
Duración del proyecto (tiempo)	Entre 4 y 6 meses incluyendo desarrollo, pruebas y entrega.
Dependencias a visitar	Área de desarrollo de software.
Rol de las personas involucradas	Analistas, Desarrolladores, Líderes de procesos, Directores de proyectos, Representantes legales de las empresas.
Estado del proyecto nuevo	No haber llegado a la etapa de especificación de requisitos.
INFORMACIÓN A RECOLECTAR	
No. de revisiones de los requisitos	Cantidad de veces que fue verificada la especificación de requisitos.
Tiempo en requisitos	Tiempo total invertido en el desarrollo de los requisitos.
Tiempo en especificación	Tiempo invertido en la especificación y verificación de requisitos.
Tiempo en validación	Tiempo invertido en la validación de requisitos.
Técnica para especificar	Técnica que se usó para especificar requisitos.
Técnica para validar	Técnica que se usó para validar requisitos.
No. de defectos en la verificación de requisitos	No. Defectos encontrados en la especificación de requisitos durante la verificación.
No. de defectos en la validación de requisitos	No. Defectos encontrados en la especificación de requisitos durante la validación.
No. de defectos en producción	No. de defectos atribuibles a los requisitos encontrados en la etapa de producción mediante el uso de técnicas no suministradas por PEVReS y reportados por las pymes.
Nivel de satisfacción del equipo	Nivel de satisfacción del equipo desarrollador respecto al trabajo realizado con los requisitos.
No. de modelos elaborados	Cantidad de diagramas/modelos realizados durante la especificación de los requisitos.
No. de documentos elaborados	Cantidad de documentos resultante de la especificación de requisitos.

De igual forma, es necesario realizar un seguimiento y acompañamiento a la aplicación del procedimiento en cada piloto de aplicación, con el ánimo de recolectar las dificultades o mejoras que contribuyan al refinamiento del mismo para aplicaciones futuras, así como el seguimiento a la aplicación del procedimiento para determinar si se realizaron las actividades propuestas y se

aplicaron las técnicas establecidas, llevando además el registro de los casos en los que no se aplicó el procedimiento de manera correcta.

Así mismo, antes de usar el procedimiento elaborado se aplicará una encuesta donde se determinen las expectativas que tienen las personas responsables de su aplicación en ambas empresas, como también una encuesta después de su aplicación con el objetivo de determinar el rendimiento percibido, es decir, el "resultado" que los equipos percibieron una vez aplicado el procedimiento, con el ánimo de conseguir el nivel de satisfacción que equivaldrá a la sustracción entre el rendimiento percibido y las expectativas.

A continuación, se muestra el desarrollo del piloto de aplicación en la Empresa No.1:

7.4.1. Piloto de aplicación Empresa No. 1

Problema identificado: De acuerdo a los testimonios de algunos ingenieros partícipes del proceso de desarrollo de software en la empresa No. 1, el principal problema radica en que durante el desarrollo de requisitos no se tiene como resultado una especificación de requisitos verificada por el equipo de trabajo y validada formalmente por el usuario, que no tenga ambigüedades y que sea explícita, por el contrario existe un grado de ambigüedad que no ayuda a determinar con claridad lo que realmente se requiere desarrollar. Además, no existe un procedimiento lo suficientemente claro para validar los requisitos del sistema.

Por otro lado, no se identifican los *stakeholders* claves durante el desarrollo de los requisitos y falta contextualización respecto al dominio del problema, lo cual no permite entender claramente la integración con otros sistemas y el orden lógico en que se deben implementar. Al no existir una especificación lo suficientemente explícita, la estimación en cuanto a tiempo y recursos presenta un desfase relevante en el desarrollo del proyecto.

Finalmente, es importante destacar que no existe una cultura organizacional para llevar a cabo un proceso de desarrollo de requisitos serio y bien estructurado (detallado).

FASE 1: Diseño del piloto de aplicación

Durante esta fase se estableció con el líder del proceso relacionado con la Gestión de Tecnologías Informáticas y Sistema de Información de la Empresa No.1, el día, hora y lugar en el cual se iba a realizar la socialización del proyecto y del procedimiento para especificar y validar requisitos de software. Una vez realizada dicha socialización y previa autorización del líder, se procedió a programar las reuniones de capacitación a través al equipo responsable del desarrollo de los requisitos sobre el nuevo procedimiento para especificar y validar requisitos. Tanto la presentación del procedimiento como la capacitación para el uso del mismo se realizaron mediante exposiciones y demostraciones a través de ejemplos.

Para la selección de los proyectos, tanto el de referencia como el nuevo, se programó una reunión con el director de proyectos y los ingenieros desarrolladores.

En la misma línea, durante la planeación se estableció cual iba a ser la fuente de información primaria donde reposaba la información del proyecto de referencia, con el ánimo de recolectar la información requerida. Además, se estableció que el seguimiento a la aplicación del

procedimiento iba a ser mediante el registro en un archivo compartido de las dificultades y mejoras encontradas en el momento en el que se estaba llevando a cabo dicha actividad y el acompañamiento sería presencial en algunas ocasiones y a través de chat, correo y celular en otras.

Finalmente, se pactaron las fechas para la aplicación de la encuesta de expectativas, aclarando que la misma debía realizarse antes de iniciar la aplicación del procedimiento. Así mismo, se indicó sobre el diligenciamiento de la encuesta de rendimiento que debía ser diligencia posterior a la aplicación del mismo, con el fin de determinar el nivel de satisfacción del equipo.

A continuación, se relacionan las tareas planeadas que fueron la carta de navegación durante la ejecución del piloto de aplicación No.1:

TABLA 30. DISEÑO DEL PILOTO DE APLICACIÓN No.1

DISEÑO PILOTO DE APLICACIÓN No.1	
Reunión de socialización	3 de agosto de 2016
Reuniones de Capacitación	3 reuniones para el 5, 8 y 9 de agosto de 2016
Identificación de los proyectos	10 de agosto de 2016
Identificación de la fuente de información primaria del proyecto de referencia	Carpeta física principal del proyecto y repositorio de información de los proyectos de la empresa.
Aplicación de la encuesta de expectativas	Antes de iniciar la aplicación del procedimiento (10 de agosto de 2016).
Acompañamiento y seguimiento a la aplicación del procedimiento	Durante la especificación y validación de requisitos.
Aplicación de la encuesta de rendimiento	Después de haber aplicado el procedimiento
Roles partícipes del piloto de aplicación	Líder de proceso, Director de proyectos, Ingeniero desarrollador

FASE 2: Ejecución del piloto de aplicación

En este apartado se refleja el desarrollo de las actividades planeadas en la etapa anterior, iniciando con la reunión de socialización con el líder del proceso y el equipo de desarrollo de los requisitos, en la cual se comentó sobre el proyecto de investigación y sobre el procedimiento para especificar y validar requisitos, de la cual se elaboró un acta de reunión firmada por las partes y se realizó una presentación (**Ver ANEXO H - Presentación Procedimiento PEVReS**).

En los días siguientes se llevaron a cabo dos de las tres reuniones planeadas para realizar la respectiva capacitación, pues se consideró que con ellas era suficiente para dar inicio a la aplicación del procedimiento.

Teniendo en cuenta que el proceso de capacitación finalizó antes de lo planeado, el día 9 de agosto de 2016 se procedió a realizar la reunión con el director de proyectos y parte de los ingenieros desarrolladores, con la intención de identificar los proyectos a utilizar durante el piloto de aplicación (uno nuevo en el que se aplicará el procedimiento y otro ya realizado sin aplicarlo), además se identifica la fuente de información primaria del proyecto de referencia (proyecto ya realizado), sirviendo como soporte una base de datos donde reposaba la información relacionada

con todos los proyectos de la Empresa No.1. A continuación se relaciona la información que identifica cada proyecto:

TABLA 31. IDENTIFICACIÓN DEL PROYECTO DE REFERENCIA

IDENTIFICACIÓN DEL PROYECTO DE REFERENCIA	
PROYECTO	Sistema financiero
MÓDULO A INTERVENIR	Módulo de Facturación
DESCRIPCIÓN DEL PROYECTO	Este módulo se encarga de gestionar las facturas que se generan por medio de la venta de servicios prestados por la Empresa No.1.
No. DE RESPONSABLES	2 personas
TIPO DE PROYECTO	Sistema web transaccional.
DURACIÓN DEL PROYECTO (tiempo):	5 meses
DEPENDENCIAS VISITADAS	División de sistemas, Área de Desarrollo.
ROLES CAPACITADOS	Ingeniero desarrollador de software, con formación en ingeniería de sistemas especialista en desarrollo de software con 3 años de experiencia con conocimientos en PL/SQL y Java. Director de proyecto, con formación en ingeniería de sistemas con 5 años de experiencia en gestión de proyectos con conocimientos en PMBOK y administración en bases de datos Oracle.
LENGUAJE	Java
PLATAFORMA	SOA
No. LINEAS DE CODIGO	7.900 Aprox.

TABLA 32. IDENTIFICACIÓN DEL PROYECTO A INTERVENIR

IDENTIFICACIÓN DEL PROYECTO A INTERVENIR	
PROYECTO	Sistema financiero
MÓDULO A INTERVENIR	Módulo de Formulación de Proyectos de Operación Comercial (Hoja contratación de personal).
DESCRIPCIÓN DEL PROYECTO	Este módulo se encarga de gestionar toda la contratación de los diferentes integrantes que harán parte de la ejecución de un proyecto determinado en la empresa.
No. DE RESPONSABLES	2 personas
TIPO DE PROYECTO	Sistema web transaccional.
DURACIÓN DEL PROYECTO (tiempo):	4 meses
DEPENDENCIAS VISITADAS	División de sistemas, Área de Desarrollo.
ROLES CAPACITADOS	Ingeniero de desarrollo, con formación en ingeniería de sistemas con 5 años de experiencia con conocimientos en PL/SQL, Java y JavaScript. Director de proyecto, con formación en ingeniería de sistemas con 5 años de experiencia en gestión de proyectos con conocimientos en PMBOK y administración en bases de datos Oracle.
LENGUAJE	Java
PLATAFORMA	SOA
No. LINEAS DE CODIGO	8.100 Aprox.

Una vez identificados ambos proyectos, se procede al día siguiente a aplicar la encuesta de expectativas al equipo. Tal como estaba planeado, se compartió físicamente la encuesta de expectativas (**Ver ANEXO F - Encuesta Expectativas**) siendo diligenciada en su totalidad por los integrantes del equipo.

Considerando que se había logrado todo lo necesario para dar inicio a la recolección de la información, se inicia en paralelo tanto la aplicación del procedimiento en el nuevo proyecto como la revisión detallada de la carpeta física del proyecto de referencia, con el objetivo de recopilar la información que se planeó, además se recibió información al respecto por parte de los responsables del proyecto de referencia en su momento, a su vez se explora el repositorio de información de la Empresa No.1 donde se pudo interpretar el cronograma trazado para determinar el tiempo en requisitos, el tiempo en especificación y validación que requirió el proyecto de referencia.

Por consiguiente, se revisaron detalladamente las actas de las reuniones realizadas donde se pudo hallar el número de revisiones realizadas a los requisitos, el número de defectos durante la verificación, el número de defectos durante la validación y el número de defectos una vez desplegado en producción.

Así mismo, se tuvo acceso a la herramienta de modelado y diseño utilizada por la Empresa No.1, donde además de comprobar gran parte de la información anterior, se pudieron conocer las técnicas utilizadas para especificar y validar los requisitos utilizadas en el mismo proyecto.

Respecto al proyecto nuevo, se pudo conseguir la información planeada por medio del acompañamiento y seguimiento durante la aplicación del procedimiento, pues se logró obtener un registro juicioso por parte del equipo y más aún por los autores del procedimiento.

A continuación, se relaciona la información recolectada de ambos proyectos de forma paralela para tener una mejor visualización de la misma:

TABLA 33. DATOS RECOLECTADOS DE LOS PROYECTOS DEL PILOTO DE APLICACIÓN No.1

	EMPRESA No. 1	
	PROYECTO DE REFERENCIA	PROYECTO NUEVO
No. de revisiones de los requisitos	1	2
Tiempo en requisitos	20 días	45 días
Tiempo en especificación	8 días	15 días
Tiempo en validación	1 día	3 días
Técnica para especificar	Prototipos	<ul style="list-style-type: none"> • Especificación de requisitos de negocio en lenguaje natural • Prototipos • Especificación en Lenguaje Natural de requisitos No Funcionales y Restricciones • Especificación en Lenguaje Natural de requisitos Funcionales
Técnica para validar	Prototipos	Prototipos, <i>Walkthroughs</i>
No. de defectos en la verificación	5	10
No. de defectos en la validación	6	4
No. de defectos en producción	4	1
Nivel de satisfacción del equipo	NA	Satisfecho
No. de modelos elaborados	1	1
No. de documentos elaborados	0	1

No siendo ajeno al acompañamiento y seguimiento a la aplicación del procedimiento por parte de los autores del mismo y con la intención de recopilar las dificultades y mejoras resultantes que pudieran ser tenidas en cuenta para la realización de ajustes al procedimiento, la *Tabla No.34* presentada a continuación consigna las dificultades y mejoras resultantes de la aplicación del procedimiento por parte del equipo.

TABLA 34. SITUACIONES PRESENTADAS DURANTE LA APLICACIÓN DEL PILOTO DE APLICACIÓN NO.1

TIPO*	DESCRIPCIÓN
M	Se sugiere poner espacio para firmas tanto del usuario como del ingeniero encargado, en los formatos del procedimiento para la validación.
D	No se comprende claramente la descripción de la fuente de los datos de entrada de los requisitos funcionales presentada en el Documento SRS.
M	Se recomienda que no existan especificaciones redactadas en lenguaje natural de requisitos de usuario sino más bien que dichas especificaciones sean los prototipos, casos de uso o historias de usuario.
D	Existe ambigüedad en el nombre de la columna “Descripción de la tarea” en el formato para la validación de prototipos.
D	No se encuentra conexión entre las dos actividades presentes en el procedimiento para la validación de requisitos.
M	En las entradas del procedimiento para validar requisitos no sería necesario incluir los prototipos elaborados, ya que los mismos están incluidos en el Documento SRS realizado.
M	En los responsables de ejecutar la técnica “Casos de prueba” del procedimiento para validar requisitos, debería incluirse una persona del equipo de pruebas.

* Donde (D: Dificultad, M: Mejora)

Finalmente, una vez realizada la aplicación del procedimiento se procedió a aplicar la encuesta de rendimiento (**Ver ANEXO G - Encuesta Rendimiento**) siendo diligenciada en su totalidad por los integrantes del equipo, consiguiendo así los insumos suficientes para determinar el Nivel de Satisfacción del equipo.

FASE 3: Verificación del piloto de aplicación

Como parte de la *Etapa 3* de la metodología del proyecto y con respecto a los datos obtenidos en la fase inmediatamente anterior, es importante indicar que una vez analizados los datos obtenidos en cada uno de los proyectos del piloto de aplicación No.1, se puede observar según la *Ilustración No.18*, que al aplicar el procedimiento para especificar y validar requisitos se presenta un aumento notable en el tiempo empleado tanto para especificarlos como para validarlos, afectando a su vez el tiempo total para el desarrollo de los mismos. Lo anterior, se atribuye, entre otras razones, a que existe una curva de aprendizaje de la cual las MiPymes deben ser conscientes, pues el procedimiento establece la ejecución de actividades bien definidas documentadas paso a paso que requiere tanto la capacitación del personal destinado para su aplicación, como la disciplina que ello requiere. A continuación, se muestran gráficamente los tiempos invertidos para ambos proyectos:

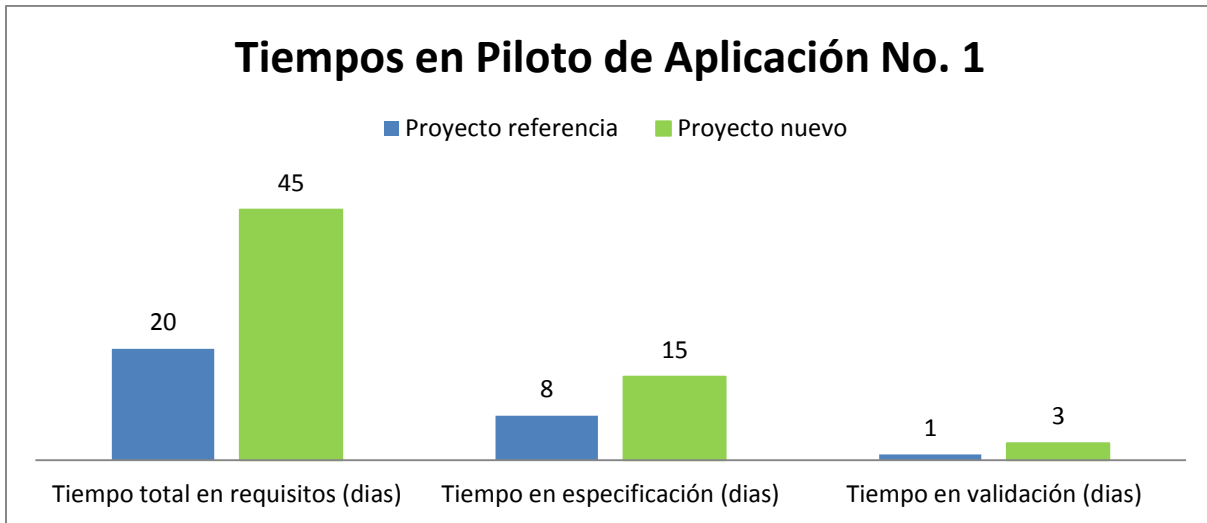


ILUSTRACIÓN 18. COMPARACIÓN DE TIEMPOS DEL PILOTO DE APLICACIÓN NO.1

Respecto al número de revisiones realizadas a la especificación de requisitos, se puede observar en la *Ilustración No.19*, que en el proyecto nuevo las mismas duplicaron la cantidad realizada en el proyecto de referencia, debido a que en este último solo se hacía una revisión entre el equipo desarrollador y el líder del proyecto pues era la metodología establecida por la empresa, mientras que para el proyecto nuevo el procedimiento propone realizar una revisión individual a la especificación realizada además de la revisión ya establecida por la empresa.

En la misma línea, es importante indicar que mediante la aplicación del procedimiento el número de modelos elaborados para especificar requisitos se mantuvo, logrando la elaboración de un documento SRS (de acuerdo al formato propuesto) validado por el usuario, el cual no se contemplaba en el proyecto de referencia tal como se refleja en la siguiente ilustración.

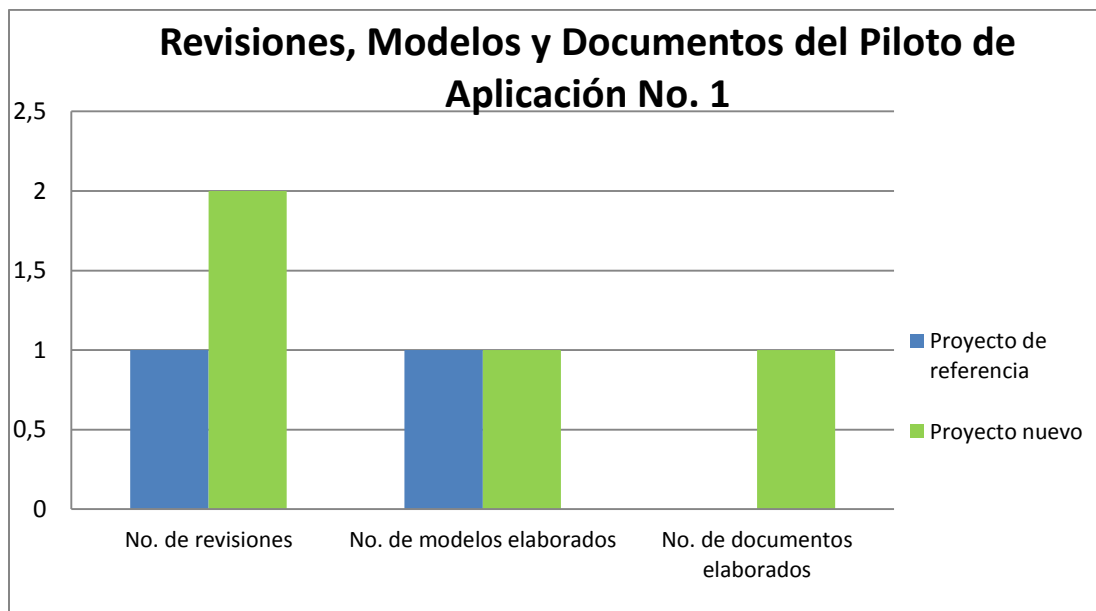


ILUSTRACIÓN 19. COMPARACIÓN DE REVISIONES, MODELOS Y DOCUMENTOS DEL PILOTO DE APLICACIÓN NO.1

Lo anterior tiene incidencia directa sobre el número de defectos encontrados en la verificación de la especificación, pues al ser revisada por el responsable de los requisitos inmediatamente después de haber sido elaborada y mediante la aplicación del formato sugerido por el procedimiento, se logró detectar mayor número de defectos, para este caso en un 60% más.

Así mismo, al haber encontrado mayor número de defectos en la verificación, se redujeron en un 33% los defectos encontrados en la validación realizada con el cliente, lo que a su vez generó una disminución en los errores encontrados en producción atribuibles a los requisitos en un 75%. La siguiente ilustración refleja el impacto favorable para el proyecto por la detección significativa de errores en etapas tempranas.

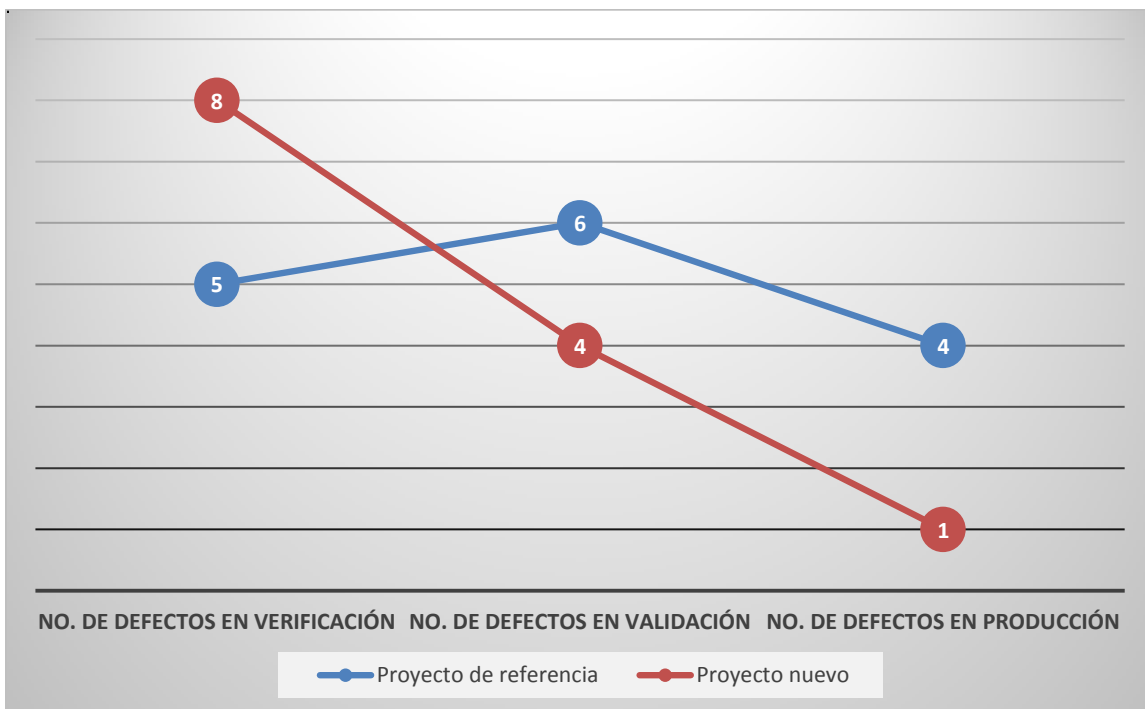


ILUSTRACIÓN 20. DEFECTOS ENCONTRADOS EN AMBOS PROYECTOS

Si bien la técnica empleada para especificar los requisitos fue prototipos y el número de modelos elaborados son iguales en ambos proyectos como se mencionó anteriormente, se atribuye la situación anterior a la aplicación de las técnicas de especificación en lenguaje natural empleadas para especificar los requisitos y al uso de las técnicas de validación propuestas por el procedimiento.

Finalmente, se concluye que, si bien para la aplicación del nuevo procedimiento fue necesario emplear más tiempo, se considera que el esfuerzo fue recompensado por el hallazgo de mayor número de defectos en etapas tempranas, lo que finalmente constituye una ganancia respecto al costo de corregir dichos errores en etapas posteriores.

Nivel de satisfacción de la aplicación del procedimiento

Para determinar el nivel de satisfacción del equipo encargado del desarrollo de los requisitos en la Empresa No. 1, se analizaron los resultados obtenidos una vez fue aplicada la encuesta de expectativas las cuales se tipificaron como Expectativas *Elevadas*, *Moderadas* o *Bajas* mediante la asignación de los valores 3, 2 o 1 respectivamente. El nivel de expectativa obtenido luego de haber promediado los resultados obtenidos de la aplicación de la encuesta a las personas encargadas fue *Moderadas (2)*, la siguiente Ilustración refleja los resultados de cada una de las siguientes preguntas formuladas:

1. ¿Espera que la aplicación del procedimiento satisfaga los problemas de especificación y validación de requisitos con que cuenta el proceso actual de su empresa?
2. ¿Considera que con la aplicación del procedimiento, su proceso actual para el desarrollo de requisitos será más efectivo?
3. ¿Aspira que los recursos necesarios para la aplicación del procedimiento para especificar y validar requisitos estén dentro de lo que la empresa pueda proporcionar?
4. ¿Espera que el procedimiento propuesto sea explícito respecto a los pasos necesarios para su aplicación?
5. ¿Espera que el procedimiento le entregue los elementos necesarios para realizar una completa especificación y validación de requisitos?

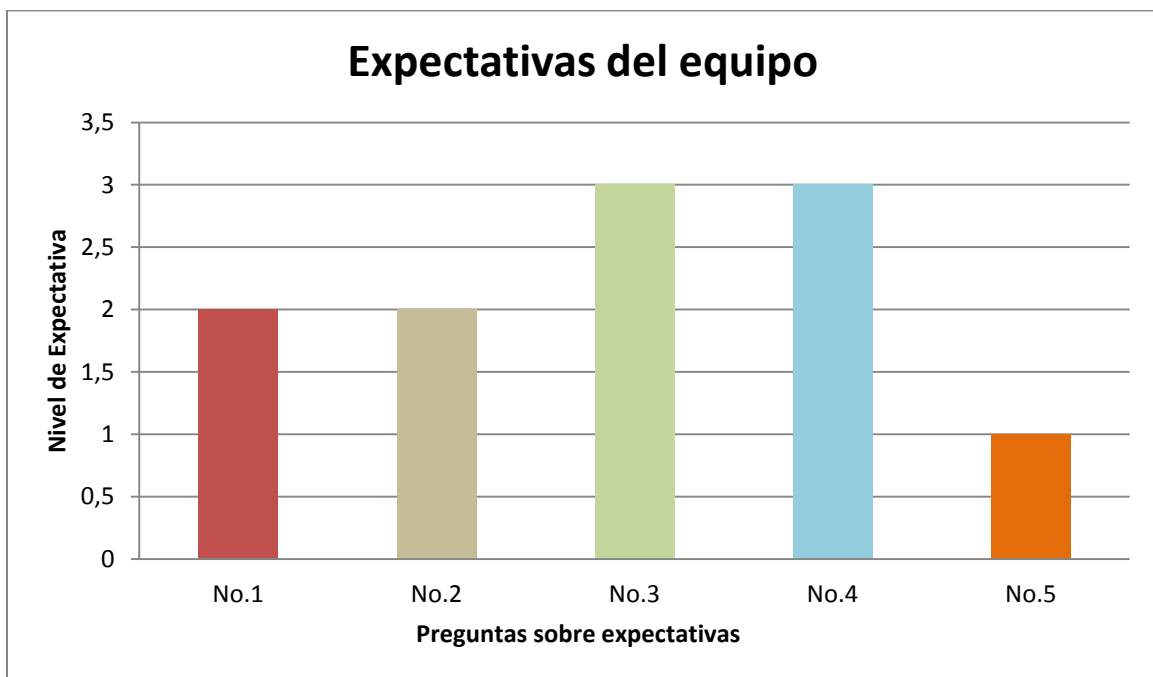


ILUSTRACIÓN 21. EXPECTATIVAS DEL EQUIPO EN EL PILOTO DE APLICACIÓN NO.1

Una vez obtenido el nivel de expectativa y habiendo realizado la aplicación del procedimiento, se procedió a aplicar una segunda encuesta (**Ver ANEXO G - Encuesta Rendimiento**) con la intención de determinar el rendimiento que percibieron las personas encargadas sobre la aplicación del procedimiento propuesto. Para ello, se usaron las siguientes opciones de respuesta a las preguntas formuladas: **Excelente=10**, **Bueno=7**, **Regular=5**, **Malo=3**. De esta manera, se logró establecer que el rendimiento percibido en promedio fue (**8**). A continuación, se refleja gráficamente los resultados a las siguientes preguntas formuladas:

1. ¿La secuencia y estructuración de las actividades definidas en el procedimiento para especificar y validar requisitos están acordes con el objetivo que se quiere alcanzar?
2. ¿El procedimiento para especificar y validar requisitos es explícito respecto a los pasos necesarios para la aplicación de cada una de las técnicas propuestas?
3. ¿Los formatos y guías propuestos en el procedimiento para especificar y validar requisitos fueron el soporte suficiente para la ejecución de las actividades?
4. ¿Considera que los resultados obtenidos después de haber aplicado el procedimiento para especificar y validar los requisitos cumplieron con las expectativas trazadas?
5. ¿Los recursos necesarios en cuanto a tiempo y personas para la aplicación del procedimiento fueron suficientes según lo programado?

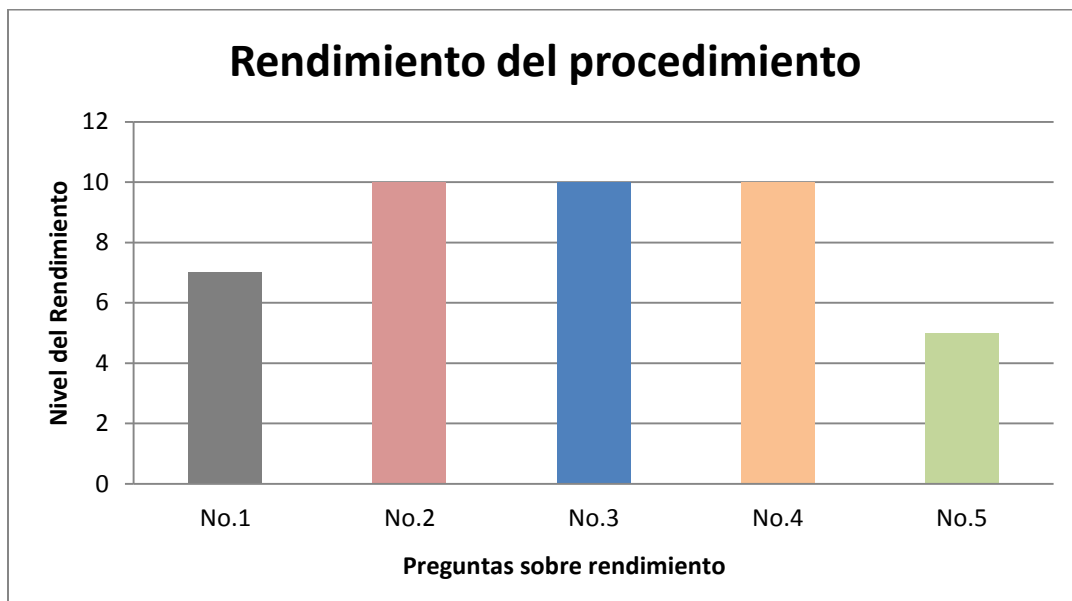


ILUSTRACIÓN 22. RENDIMIENTO DEL PROCEDIMIENTO EN EL PILOTO DE APLICACIÓN NO.1

Se observa en la Ilustración anterior un valor bajo en la respuesta a la pregunta No. 5, atribuible a la curva de aprendizaje requerida para adquirir dominio sobre el uso del procedimiento por parte de los responsables de los requisitos en la empresa. Esto puede ser motivo de una posible recomendación a las Mipymes a la hora de aplicar el procedimiento presentado.

Posteriormente se aplica la siguiente fórmula que según (Armstrong & Kotler, 2013, pág. 14) permite medir el nivel de satisfacción, utilizando los valores promedios obtenidos anteriormente tanto de las expectativas como del rendimiento percibido.

$$\text{Nivel de Satisfacción} = \text{Rendimiento Percibido} - \text{Expectativas}$$

$$\text{Nivel de satisfacción} = 8 - 2 = 6$$

Finalmente, se estableció la siguiente escala para determinar el nivel de satisfacción:

TABLA 35. NIVEL DE SATISFACCIÓN SEGÚN RANGO

NIVEL DE SATISFACCIÓN	RANGO
Complacido	8 <= Puntaje <= 10
Satisfecho	5 <= Puntaje <= 7
Insatisfecho	1 <= Puntaje <= 4

Donde, una vez ubicado el resultado final correspondiente a **6**, se obtuvo como resultado que el equipo desarrollador quedo **SATISFECHO** respecto al trabajo realizado con los requisitos, pues las expectativas que se tenían antes de aplicar el procedimiento fueron en su mayoría cumplidas con los resultados obtenidos una vez aplicado el mismo.

7.4.2. Piloto de Aplicación Empresa No. 2

Problema Empresa No. 2: La principal dificultad expresada por los representantes de la empresa es que generalmente existen problemas de comunicación atribuidas al uso del lenguaje entre el usuario y el ingeniero, pues el primero manifiesta una necesidad y el segundo la entiende e interpreta de otra manera, es decir, existe ambigüedad en los requisitos. Así mismo, se presenta incompletitud en la especificación de los requisitos, ya que el usuario no expresa todos sus requisitos en las etapas de *licitación* o análisis, lo que se ve reflejado en un producto que no cumple todas las expectativas del usuario.

Por otro lado, no se hace una adecuada identificación de restricciones, reglas de negocio, suposiciones y dependencias, las cuales afectan en gran medida el desarrollo del proyecto. Tampoco se controla cuando existen requisitos externos generados por el entorno o contexto del proyecto, que no fueron contemplados durante el desarrollo de los requisitos.

FASE 1: Diseño del piloto de aplicación

Como parte de planeación de este piloto de aplicación se pactó inicialmente con el Representante legal de la Empresa No.2, la agenda para realizar la socialización del proyecto y del procedimiento para especificar y validar requisitos de software. Luego, se programaron las reuniones de capacitación a los ingenieros responsables del desarrollo de los requisitos sobre el nuevo procedimiento, como también la reunión para seleccionar los proyectos, tanto el de referencia como el nuevo.

Otra tarea planeada que se llevó a cabo fue el establecimiento de la fuente de información de la cual se pudiera extraer la información del proyecto de referencia para ser analizada

posteriormente frente a la información obtenida del proyecto nuevo. Además, se estableció que el seguimiento a la aplicación del procedimiento iba a ser mediante el registro en un archivo compartido, de las dificultades presentadas y posibles mejoras que se pudieran presentar. Por exigencia de la empresa, el acompañamiento sería de forma presencial debido a que por ser una práctica nueva para la empresa se consideró importante que el contacto con los autores del procedimiento fuera de esta manera.

Finalmente, se programó el día para para la aplicación de la encuesta de expectativas, aclarando que la misma debía realizarse antes de iniciar la aplicación del procedimiento. Así mismo, se indicó sobre el diligenciamiento de la encuesta de rendimiento que debía ser diligencia posterior a la aplicación del mismo, con el fin de determinar el nivel de satisfacción del equipo.

A continuación, se relacionan las tareas planeadas para la ejecución del piloto de aplicación No.2:

TABLA 36. DISEÑO DEL PILOTO DE APLICACIÓN No.2

DISEÑO PILOTO DE APLICACIÓN No.2	
Reunión de socialización	11 de agosto de 2016
Reuniones de Capacitación	2 reuniones para el 12 y 13 de agosto de 2016
Identificación de los proyectos	15 de agosto de 2016
Identificación de la fuente de información primaria del proyecto de referencia	Carpeta física principal del proyecto
Aplicación de la encuesta de expectativas	Antes de iniciar la aplicación del procedimiento (15 de agosto de 2016).
Acompañamiento y seguimiento a la aplicación del procedimiento	Durante la especificación y validación de requisitos.
Aplicación de la encuesta de rendimiento	Después de haber aplicado el procedimiento
Roles partícipes del piloto de aplicación	Representante legal, Ingeniero desarrollador

FASE 2: Ejecución del piloto de aplicación

Esta etapa presenta el desarrollo de las actividades planeadas en la Fase No.1, las cuales tienen que ver con la recolección de la información de los proyectos a utilizar durante el piloto de aplicación (uno nuevo en el que se aplicará el procedimiento y otro ya realizado sin aplicarlo), así como el seguimiento y el acompañamiento realizado durante la aplicación del procedimiento.

Como primera medida y tal como se había planeado se llevó a cabo la reunión de socialización con el representante legal de la empresa No.2, en la cual se expuso el proyecto de investigación y el procedimiento. Las apreciaciones de la reunión fueron consignadas en un acta firmada por las partes y se realizó una presentación (**Ver ANEXO H - Presentación Procedimiento PEVReS**).

Posteriormente se realizaron las dos reuniones planeadas para realizar la capacitación a las personas encargadas por la empresa para hacer la aplicación del procedimiento, debido a las limitaciones de tiempo fue necesario programar una sesión más para finalizar el proceso de capacitación.

Una vez terminada la capacitación el día 15 de agosto de 2016, se realizó la selección de los proyectos a utilizar durante el piloto de aplicación y se identifica la fuente de información primaria del proyecto de referencia (proyecto ya realizado). A continuación, se relaciona la información que identifica cada proyecto:

TABLA 37. IDENTIFICACIÓN DEL PROYECTO DE REFERENCIA

IDENTIFICACIÓN DEL PROYECTO DE REFERENCIA	
PROYECTO	Sistema para la gestión de asistencias y registro de actividades
MÓDULO A INTERVENIR	Módulo de registro de asistencia y actividades del gimnasio
DESCRIPCIÓN DEL PROYECTO	Este módulo permite el registro de las diferentes asistencias de los usuarios y las actividades que ellos realizan durante su permanencia en el gimnasio.
No. DE RESPONSABLES	2 personas
TIPO DE PROYECTO	Sistema web transaccional
DURACIÓN DEL PROYECTO (tiempo):	5 meses
DEPENDENCIAS VISITADAS	Departamento de informática.
ROLES CAPACITADOS	Ingeniero de desarrollo, con formación en ingeniería de sistemas y MBA con 8 años de experiencia con conocimientos en Java, PL/SQL, JavaScript, GWT y Swing. Director de proyecto, con formación en ingeniería de sistemas con 2 años de experiencia en gestión de proyectos con conocimientos en PMBOK.
LENGUAJE	Java
PLATAFORMA	SOA
No. LINEAS DE CODIGO	8.700 Aprox.

TABLA 38. IDENTIFICACIÓN DEL PROYECTO A INTERVENIR

IDENTIFICACIÓN DEL PROYECTO A INTERVENIR	
PROYECTO	Sistema de caracterización y ponderación socioeconómica.
MÓDULO A INTERVENIR	Módulo de solicitudes socioeconómicas.
DESCRIPCIÓN DEL PROYECTO	Este módulo permite el diligenciamiento de solicitudes socioeconómicas con el objetivo de otorgar apoyos socioeconómicos a las personas que lo requieren.
No. DE RESPONSABLES	2 personas
TIPO DE PROYECTO	Sistema web transaccional
DURACIÓN DEL PROYECTO (tiempo):	6 meses
DEPENDENCIAS VISITADAS	Departamento de informática.
ROLES CAPACITADOS	Ingeniero de desarrollo, con formación en ingeniería de sistemas y MBA con 8 años de experiencia con conocimientos en Java, PL/SQL, JavaScript, GWT y Swing. Director de proyecto, con formación en ingeniería de sistemas con 2 años de experiencia en gestión de proyectos con conocimientos en PMBOK.
LENGUAJE	Java
PLATAFORMA	SOA
No. LINEAS DE CODIGO	9.000 Aprox.

Una vez identificados ambos proyectos, se procedió el mismo día a realizar la aplicación de la encuesta de expectativas al equipo. Tal como estaba planeado, se compartió físicamente la encuesta de expectativas (**Ver ANEXO F - Encuesta Expectativas**) siendo diligenciada por todos los integrantes del equipo.

Considerando que ya se tenía lo necesario para iniciar la recolección de la información planeada, se inicia en paralelo tanto la aplicación del procedimiento en el nuevo proyecto como la revisión detallada de la carpeta física del proyecto de referencia en la que se pudo analizar el cronograma existente para determinar el tiempo en requisitos, el tiempo en especificación y validación que requirió el proyecto de referencia.

Acto seguido, se dio la revisión de las actas de las reuniones sostenidas con los usuarios y entre el mismo equipo de desarrollo, donde se pudo conseguir el número de revisiones realizadas a los requisitos, el número de defectos durante la verificación, el número de defectos durante la validación, el número de defectos una vez desplegado en producción y las técnicas utilizadas para especificar y validar los requisitos.

Respecto al proyecto nuevo, se pudo conseguir la información que se esperaba mediante el acompañamiento y seguimiento a la aplicación del procedimiento. Esto permitió además consolidar en un archivo digital todo lo sucedido en la aplicación de tal manera los autores del procedimiento pudieran, con el apoyo del equipo desarrollador, determinar las dificultades, mejoras y lecciones aprendidas que permitieran refinar posteriormente el procedimiento.

A continuación, se presenta de forma paralela la información recolectada de ambos proyectos para un mejor análisis:

TABLA 39. DATOS RECOLECTADOS DE LOS PROYECTOS DEL PILOTO DE APLICACIÓN No.2

	EMPRESA No. 2	
	PROYECTO ANTERIOR	PROYECTO NUEVO
No. de revisiones de los requisitos	0	1
Tiempo en requisitos	25 días	36 días
Tiempo en especificación	5 días	20 días
Tiempo en validación	1 días	2 días
Técnica para especificar	Casos de uso	<ul style="list-style-type: none"> • DFD • Especificación de requisitos de negocio en lenguaje natural • Prototipos • Especificación en Lenguaje Natural de requisitos No Funcionales y Restricciones • Especificación en Lenguaje Natural de requisitos Funcionales
Técnica para validar	Revisiones	Prototipos, <i>Walkthroughs</i>
No. de defectos en la verificación	3	18
No. de defectos en la validación	8	5
No. de defectos en producción	10	2
Nivel de satisfacción del equipo	NA	Satisfecho
No. de modelos elaborados	1	2
No. de documentos elaborados	0	1

De acuerdo al registro digital de información mencionado en el párrafo anterior, se presentan a continuación en la *Tabla No.40* las dificultades y mejoras resultantes de la aplicación del procedimiento:

TABLA 40. SITUACIONES PRESENTADAS DURANTE LA APLICACIÓN PILOTO DE APLICACIÓN No.2

TIPO*	DESCRIPCIÓN
M	Se sugiere que la técnica para especificar requisitos de funcionales en lenguaje natural sea requerida para garantizar que los mismos sean incluidos en el Documento SRS.
M	Se recomienda que antes de ejecutar el procedimiento para la validación de requisitos se tenga como <u>precondición el haber realizado una planificación de la misma.</u>
M	El procedimiento propone la verificación de la especificación realizada por parte de quien la desarrollo, sería interesante que además dicha verificación fuera realizada por otra persona (Inspección).
D	El paso 5 de la técnica prototipos “Verificar el prototipo con el cliente e identificar otros requisitos (Evaluación)” presenta un error debido a que la verificación no debería llevarse a cabo con el cliente, sino por parte del mismo responsable de la técnica.
D	El paso 7 de la técnica prototipos “Realizar la documentación necesaria para su implementación”, es ambigua, pues no especifica a que documentación hace referencia.
D	El procedimiento de validación de requisitos no muestra una secuencia lógica que permita relacionar las actividades propuestas, pues la Actividad No. 2 puede contener a la Actividad No. 1.

* Donde (D: Dificultad, M: Mejora)

FASE 3: Verificación del piloto de aplicación

Con respecto a los datos obtenidos en la fase anterior y una vez analizada la información de cada uno de los proyectos del piloto de aplicación No.2, se puede observar según la *Ilustración No.23*, que al aplicar el procedimiento para especificar y validar requisitos se presentó un aumento significativo en el tiempo empleado para especificar y validar los requisitos, afectando a su vez el tiempo total para el desarrollo de los mismos tal como se sucedió en el piloto de aplicación No.1. A continuación se muestran gráficamente los tiempos invertidos para ambos proyectos:

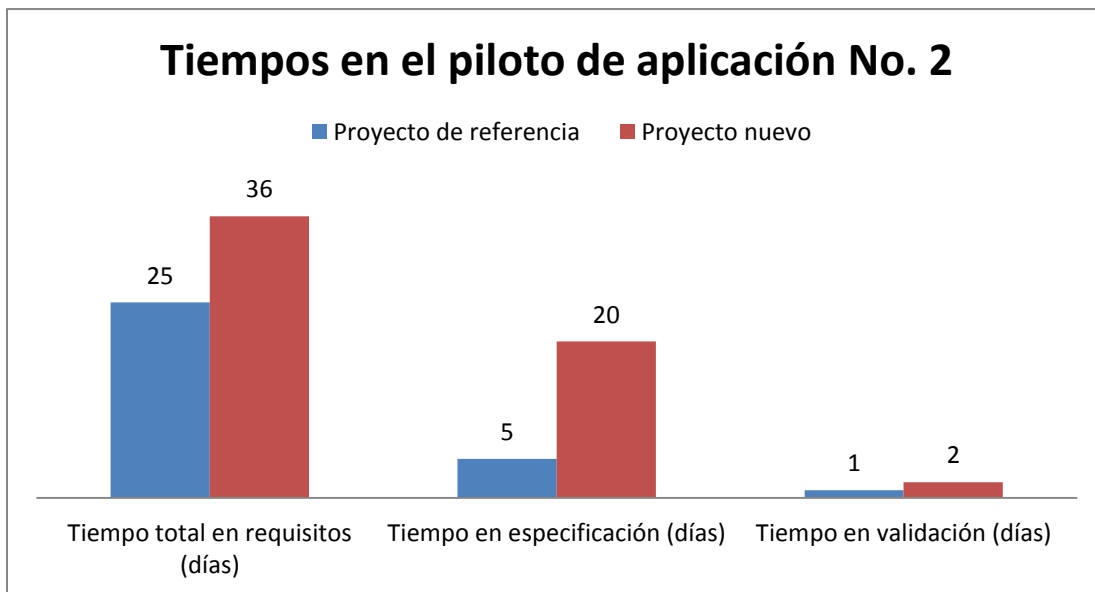


ILUSTRACIÓN 23. COMPARACIÓN DE TIEMPOS DEL PILOTO DE APLICACIÓN No.2

Respecto al número de revisiones realizadas a la especificación de requisitos, se puede observar en la *Ilustración No.24*, que en el proyecto de referencia no se llevaban a cabo revisiones a la especificación de requisitos lo que, si se logró en el proyecto nuevo, pues es una de las propuestas del procedimiento.

De la aplicación del procedimiento también se puede observar que el número de modelos elaborados para especificar requisitos se duplicó, pues aparte de realizar los diagramas requeridos por el procedimiento se realizó un Diagrama de Flujo de Datos para lograr un mejor entendimiento de los requisitos del negocio. Los resultados conllevaron a conseguir un documento SRS (de acuerdo al formato propuesto) validado por el usuario, el cual no se hacía en el proyecto de referencia tal como se refleja en la siguiente ilustración.

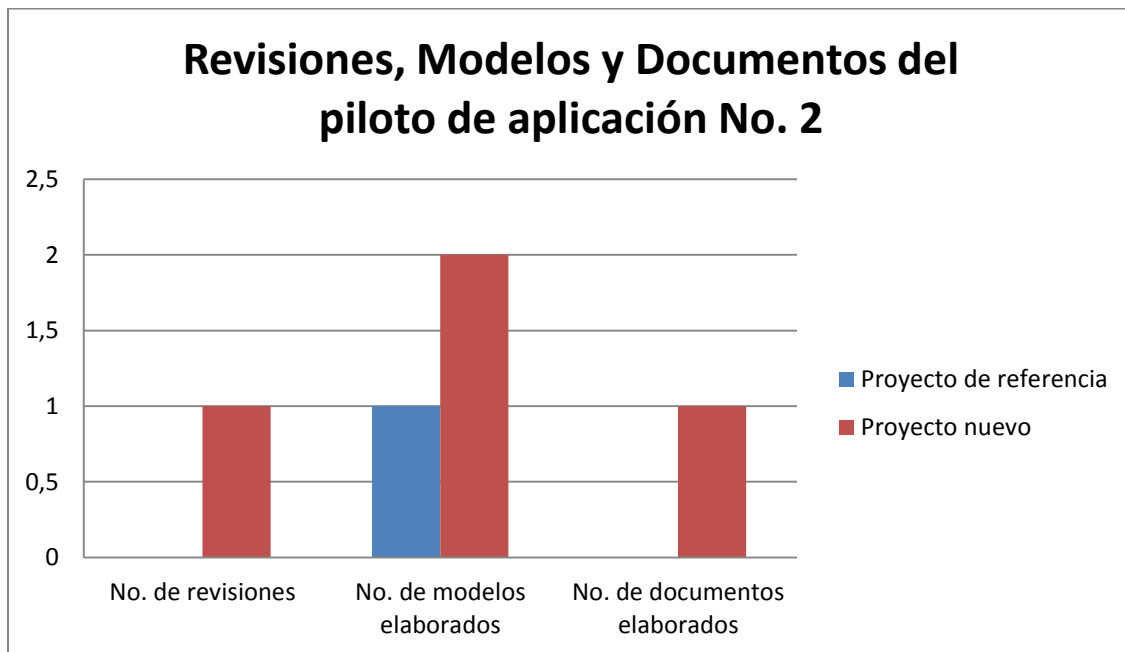


ILUSTRACIÓN 24. COMPARACIÓN DE REVISIONES, MODELOS Y DOCUMENTOS DEL PILOTO DE APLICACIÓN NO.2

De acuerdo a los resultados de la aplicación en el proyecto nuevo se logró evidenciar que el procedimiento influyó significativamente sobre el número de defectos encontrados durante la verificación de la especificación, pues se logró detectar un 83% más de defectos. Así mismo, al haber encontrado mayor número de defectos en la verificación, se presentó una disminución el 38% los defectos encontrados en la validación realizada con el cliente, lo que a su vez se vio reflejado en el porcentaje de errores encontrados en producción que corresponde a un 80% menos.

Luego de analizar esta situación, se pudo inferir que los resultados anteriormente descritos son producto de la aplicación de las técnicas requeridas por el procedimiento para especificar y validar requisitos, así como de algunas técnicas con carácter de recomendadas tales como el Diagrama de Flujo de Datos utilizada en este piloto de aplicación.

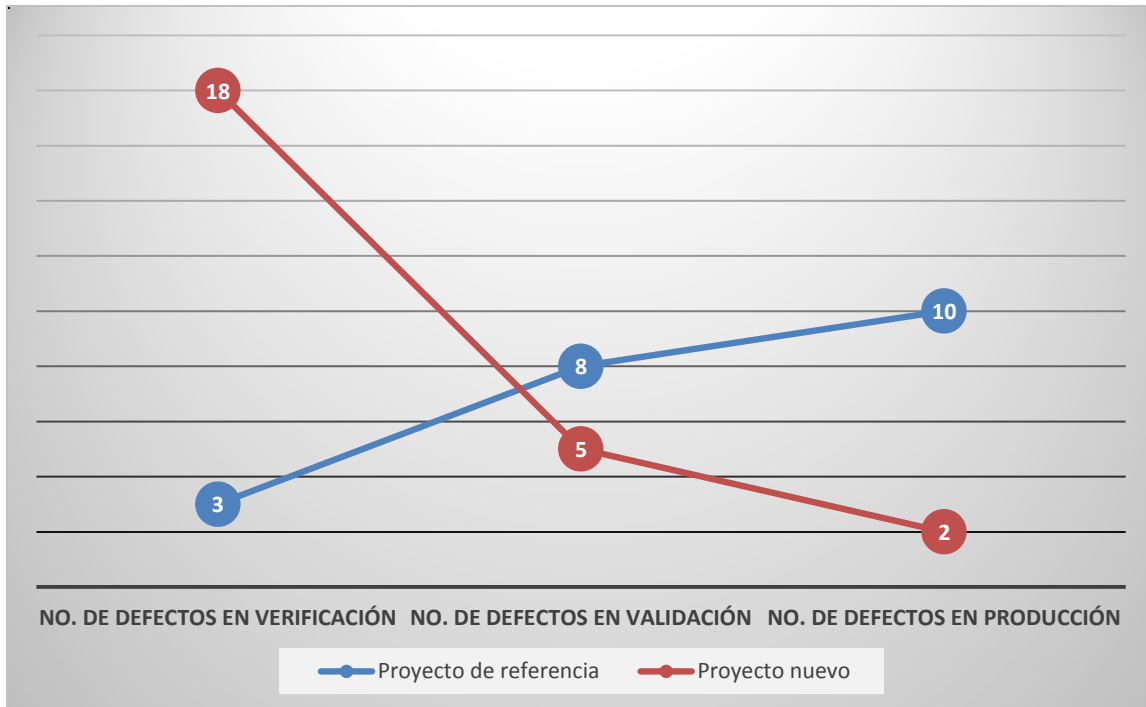


ILUSTRACIÓN 25. COMPARACIÓN DE DATOS OBTENIDOS PILOTO DE APLICACIÓN NO.2

Como conclusión de este piloto de aplicación se puede decir que el efecto de invertir más tiempo y recursos en la aplicación correcta del procedimiento conllevó al descubrimiento temprano de un mayor número de defectos atribuibles a los requisitos de software, lo que significa una disminución importante en el costo de la calidad del proyecto, pues previno la necesidad de corregir dichos errores en etapas posteriores lo que se hubiera traducido en un aumento del costo del proyecto.

Nivel de satisfacción de la aplicación del procedimiento

Para determinar el nivel de satisfacción del equipo encargado del desarrollo de los requisitos en la Empresa No. 2, se aplicó inicialmente la misma encuesta (**Ver ANEXO F - Encuesta Expectativas**) realizada a la Empresa No. 1. El nivel de expectativa obtenido luego de haber promediado los resultados obtenidos de la aplicación de la encuesta a las dos personas encargadas fue *Elevadas* (3). La siguiente Ilustración refleja los resultados de cada una de las preguntas formuladas:

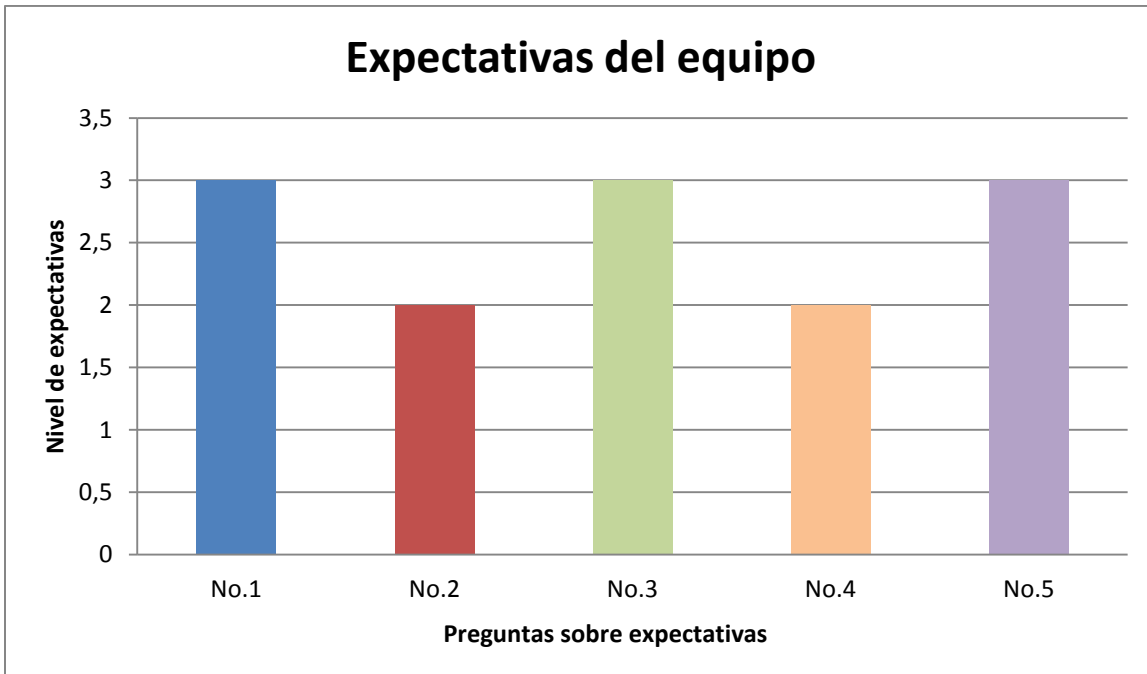


ILUSTRACIÓN 26. EXPECTATIVAS DEL EQUIPO EN EL PILOTO DE APLICACIÓN No.2

Una vez obtenido el nivel de expectativa y habiendo realizado la aplicación del procedimiento, se procedió a aplicar una segunda encuesta (**Ver ANEXO G - Encuesta Rendimiento**) con la intención de determinar el rendimiento que percibieron las personas encargadas sobre la aplicación del procedimiento propuesto. Para ello, se usaron las siguientes opciones de respuesta a las preguntas formuladas: *Excelente*=10, *Bueno*=7, *Regular*=5, *Malo*=3. De esta manera, se logró establecer que el rendimiento percibido en promedio fue (8). A continuación, se reflejan gráficamente los resultados obtenidos de las preguntas formuladas:

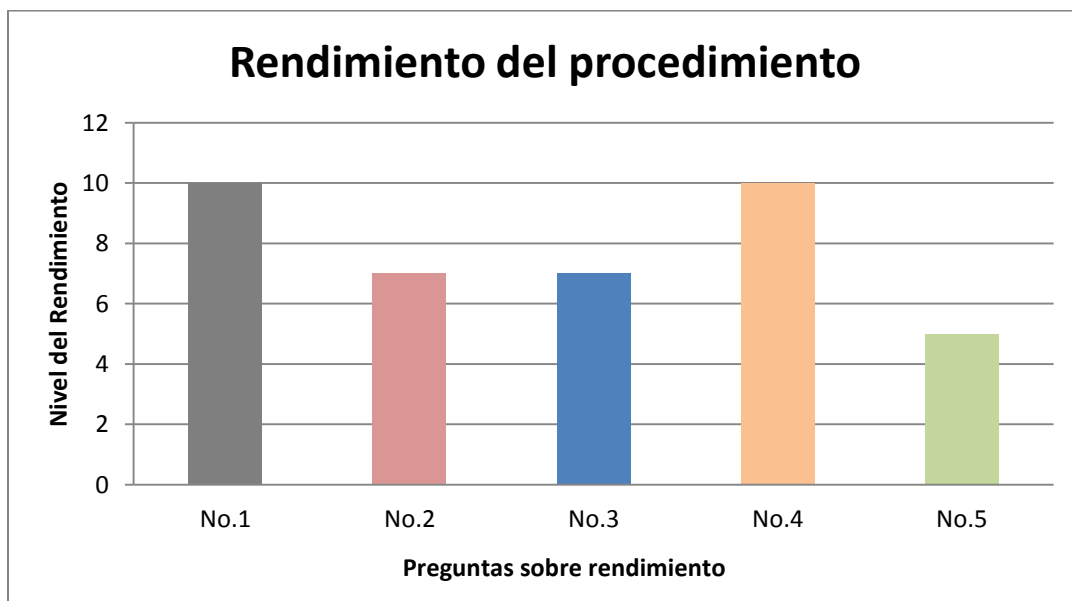


ILUSTRACIÓN 27. RENDIMIENTO DEL PROCEDIMIENTO PILOTO DE APLICACIÓN No.2

Al igual que en la *Ilustración No. 22*, se observa que los encuestados tuvieron una percepción desfavorable respecto al tiempo y personas requeridas para la aplicación del procedimiento durante el desarrollo de los requisitos, lo cual, como se expresó anteriormente se atribuye a la necesidad de invertir tiempo adicional en el aprendizaje del uso del nuevo procedimiento, como también de recurso humano para su aplicación mientras se adquiere dominio sobre el mismo.

Posteriormente se aplica la siguiente fórmula que según (Armstrong & Kotler, 2013, pág. 14) permite medir el Nivel de satisfacción, utilizando los valores promedios obtenidos anteriormente tanto de las expectativas como del rendimiento percibido.

$$\text{Nivel de Satisfacción} = \text{Rendimiento Percibido} - \text{Expectativas}$$

$$\text{Nivel de satisfacción} = 8 - 3 = 5$$

Donde, se estableció la siguiente escala para el nivel de satisfacción:

TABLA 41. NIVEL DE SATISFACCIÓN SEGÚN RANGO

NIVEL DE SATISFACCIÓN	RANGO
Complacido	8 <= Puntaje <= 10
Satisfecho	5 <= Puntaje <= 7
Insatisfecho	1 <= Puntaje <= 4

El resultado final corresponde a 5, lo que significa que el equipo responsable del desarrollo de los requisitos en la Empresa No. 2 quedó **SATISFECHO** con el procedimiento.

Análisis de consistencia de las encuestas de Expectativas y Rendimiento

Para estimar la fiabilidad de las encuestas de Expectativas y de Rendimiento se procedió a utilizar un procedimiento que permite calcular la confiabilidad y validez de los instrumentos denominado Alfa de Cronbach (BIOESTADISTICO, 2014), el cual indica que la validez se refiere al grado en que el instrumento mide lo que se pretende calcular y la confiabilidad hace referencia a la confianza que se concede a los datos. Como se indica en la *Ilustración No.28* un alfa de cronbach superior al 0.8 indica que existe fiabilidad en el instrumento que se aplicó, mientras que si el alfa no supera el 0.6 se habla de un instrumento inconsistente y posiblemente inestable.



ILUSTRACIÓN 28. NIVEL DE CONSISTENCIA ALFA DE CRONBACH (BIOESTADISTICO, 2014)

Para determinar el *Alfa de Cronbach* de ambas encuestas se aplicó el método **Varianza de los Ítems** mediante la siguiente formula:

$$\alpha = \frac{K}{K-1} \left[1 - \frac{\sum V_i}{V_t} \right]$$

Dónde:

α = Alfa de Cronbach

K = Número de ítems (preguntas).

V_i = Varianza individual de cada ítem.

V_t = Varianza total.

TABLA 42. DATOS ENCUESTA DE EXPECTATIVAS DE LOS DOS PILOTOS DE APLICACIÓN

ENCUESTAS	PREGUNTAS					TOTAL
	1	2	3	4	5	
1	3	2	3	3	1	12
2	2	1	2	2	1	8
3	2	2	3	3	2	12
4	2	1	3	2	1	9
5	2	1	2	2	2	9
6	3	3	3	3	3	15
7	3	2	3	2	3	13
8	3	3	3	2	3	14
VARIANZAS	0,29	0,70	0,21	0,27	0,86	6,57

Una vez aplicada la formula se obtuvieron los siguientes datos:

TABLA 43. RESULTADO DEL ALFA DE CRONBACH

K	5
$\sum V_i$	2,32
Vt	6,57
SECCIÓN 1	1,25
SECCIÓN 2	0,65
ABSOLUTO S2	0,65
α	0,81

Dado que el resultado del *Alfa de Cronbach* obtenido fue del 0.81, se considera que existe evidencia de que los elementos miden el mismo constructo, es decir, que las preguntas formuladas apuntan a determinar las expectativas de los encuestados de ambas empresas y las respuestas permiten medir de forma adecuada lo que esperan del procedimiento PEVReS.

De igual forma se realizó el análisis de consistencia para la encuesta de rendimiento con los siguientes datos:

TABLA 44. DATOS ENCUESTA DE RENDIMIENTO DE LOS DOS PILOTOS DE APLICACIÓN

ENCUESTAS	PREGUNTAS					TOTAL
	1	2	3	4	5	
1	7	10	10	10	5	42
2	10	10	10	10	5	45
3	5	10	10	10	5	40
4	7	10	10	10	5	42
5	10	7	7	10	5	39
6	10	10	7	10	5	42
7	10	5	7	10	5	37
8	10	7	7	10	5	39
VARIANZAS	3,98	3,98	2,57	0,00	0,00	6,21

Los datos obtenidos una vez aplicada la fórmula, se presentan a continuación:

TABLA 45. RESULTADO DEL ALFA DE CRONBACH

K	5
$\sum V_i$	10,54
Vt	6,21
SECCIÓN 1	1,25
SECCIÓN 2	-0,70
ABSOLUTO S2	0,70
α	0,87

Puesto que el *Alfa de Cronbach* obtenido fue del 0.87, la encuesta permitió evidenciar que el rendimiento percibido por los encuestados de ambas empresas respecto a la aplicación del procedimiento PEVReS fue adecuado.

7.5. Realización de ajustes al procedimiento propuesto

De acuerdo al seguimiento realizado a la aplicación del procedimiento en los pilotos de aplicación, se tienen en cuenta las dificultades presentadas y las mejoras sugeridas en las *Tablas No. 34 y 40* para la realización de ajustes al procedimiento de especificación y validación de requisitos y para dar cumplimiento a la *Etapa 4* de la metodología del proyecto relacionada con el actuar respecto al ciclo PHVA, se realizaron los ajustes al procedimiento propuesto de acuerdo a las dificultades y mejoras obtenidas durante la verificación de los pilotos de aplicación en ambas empresas (*Etapa 3* de la metodología). Los ajustes realizados fueron los siguientes:

TABLA 46. AJUSTES REALIZADOS AL PROCEDIMIENTO DE ESPECIFICACIÓN Y VALIDACIÓN DE REQUISITOS

TIPO*	DESCRIPCIÓN	ACCIÓN
M	Se sugiere poner espacio para firmas tanto del usuario como del ingeniero encargado, en los formatos del procedimiento para la validación.	Implementada
D	No se comprende claramente la descripción de la fuente de los datos de entrada de los requisitos funcionales presentada en el formato del Documento SRS.	Se cambia la redacción
M	Se recomienda que no existan especificaciones en lenguaje natural de requisitos de usuario sino más bien que dichas especificaciones sean los prototipos, casos de uso o historias de usuario.	Implementada
D	Existe ambigüedad en el nombre de la columna “Descripción de la tarea” en el formato para la validación de prototipos.	Se cambia la redacción
D	No se encuentra conexión entre las dos actividades presentes en el procedimiento para la validación de requisitos.	Se cambia el nombre de la actividad No.2 del procedimiento a “Validación de Requisitos del Sistema”
M	En las entradas del procedimiento para validar requisitos no sería necesario incluir los prototipos elaborados, ya que los mismos están incluidos en el Documento SRS realizado.	Implementada
M	En los responsables de ejecutar la técnica “Casos de prueba” del procedimiento para validar requisitos, debería incluirse una persona del equipo de pruebas.	Implementada
M	Se sugiere que la técnica para especificar requisitos de funcionales en lenguaje natural sea requerida para garantizar que los mismos sean incluidos en el Documento SRS.	Implementada
M	Se recomienda que antes de ejecutar el procedimiento para la validación de requisitos se tenga como precondition el haber realizado una planificación de la misma.	Implementada
M	El procedimiento propone la verificación de la especificación realizada por parte de quien la desarrollo, sería interesante que además dicha verificación fuera realizada por otra persona (Inspección).	Implementada
D	El paso 5 de la técnica prototipos “Verificar el prototipo con el cliente e identificar otros requisitos (Evaluación)” presenta un error debido a que la verificación no debería llevarse a cabo con el cliente, sino por parte del mismo responsable de la técnica.	Se corrige la inconsistencia
D	El paso 7 de la técnica prototipos “Realizar la documentación necesaria para su implementación”, es ambigua, pues no especifica a que documentación hace referencia.	Se elimina dicho paso, pues se considera innecesario
D	El procedimiento de validación de requisitos no muestra una secuencia lógica que permita relacionar las actividades propuestas, pues la Actividad No. 2 puede contener a la Actividad No. 1.	Se corrige la presentación de las actividades para que haya una secuencia entre las mismas.

Para una mejor presentación y despliegue del procedimiento inicialmente diseñado en Excel, se modeló bajo el estándar SPEM utilizando el *Eclipse Process Framework Composer* como se puede evidenciar en las siguientes ilustraciones y en el **ANEXO J - Procedimiento PEVReS modelado bajo SPEM**:



ILUSTRACIÓN 29. INFORMACIÓN GENERAL SOBRE PEVRES

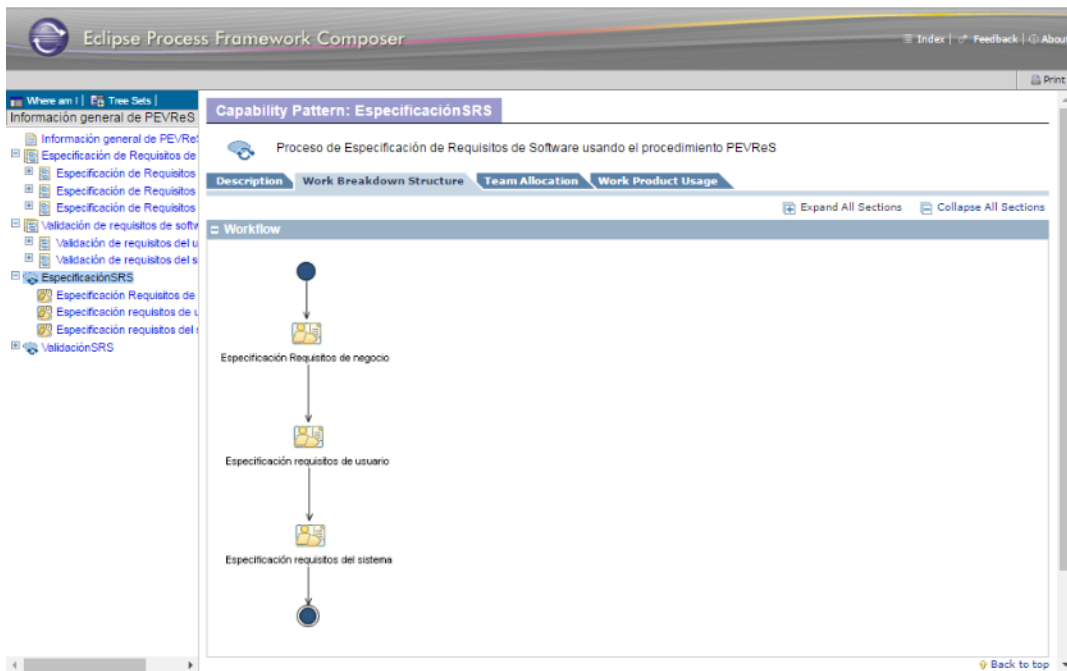


ILUSTRACIÓN 30. ACTIVIDADES DE ESPECIFICACIÓN DEL PROCEDIMIENTO

Eclipse Process Framework Composer

Where am I | Tree Sets | Index | Feedback

Task: Historias de usuario

Técnica que permite realizar las historias de usuario para realizar la especificación de requisitos de usuario.
Disciplines: Especificación de Requisitos de Usuario

Expand All Sections | Collapse All Sections

Purpose

Realizar la especificación de requisitos de usuario.

Back to top

Relationships

Roles	Primary Performer:	Additional Performers:
	<ul style="list-style-type: none"> Analista 	<ul style="list-style-type: none"> Proveedor de información
Inputs	Mandatory: <ul style="list-style-type: none"> Haber adquirido dominio del problema (precondición) Haber realizado análisis (precondición) Haber realizado elicitación (precondición) Haber realizado la planificación(precondición) 	Optional: <ul style="list-style-type: none"> Objetivos de negocio, Reglas de negocio y Restricciones (entrada) Procesos y procedimientos (entrada) Project Charter (entrada) Registro de intereses (entrada) Requerimientos iniciales (entrada)
Outputs	<ul style="list-style-type: none"> Consideraciones especiales (salida) Documento de Especificación de Requisitos de Software (SRS) Oportunidades de mejora (salida) 	

Back to top

Main Description

Para realizar la especificación de requisitos de usuario tenga en cuenta la especificación de requisitos de negocio realizada.

Back to top

Steps

- 1. Definir el formato**
Definir el formato que se utilizará para establecer la historia de usuario.
- 2. Definir el como**
Definir el Como en la Historia de Usuario (rol del usuario que hará uso de la funcionalidad).
- 3. Definir el que**
Definir el Que en la Historia de Usuario (qué hace el usuario en la historia).
- 4. Definir el para**
Definir el Para en la Historia de Usuario (el propósito de la historia, la meta que quiere alcanzar el usuario al ejecutar la historia).
- 5. Evaluar la historia de usuario**
Evaluar la Historia de Usuario realizada (Verificación por parte del autor e inspección por un par).

Back to top

Illustrations

Examples

- Plantilla de Historia de Usuario

Back to top

Key Considerations

Técnica Recomendada
(Aplicar esta técnica cuando se requiere mayor nivel de detalle y control sobre los requisitos)

Back to top

Alternatives

Diagrama de casos de uso
Prototipos

Back to top

More Information

Checklists

- Checklist de Verificación de historias de usuario

Back to top

ILUSTRACIÓN 31. DESPLIEGUE DE TAREAS EN PEVRES

8. DISCUSIÓN DE RESULTADOS

Los resultados obtenidos en este proyecto permiten describir los logros alcanzados en cada una de las etapas de la metodología propuesta, la primera de ellas tuvo como propósito realizar un levantamiento de estado de arte de los requisitos, así como una descripción y caracterización de la forma en que las MiPymes desarrolladoras de software de la ciudad de Pereira llevan a cabo la especificación y validación de requisitos, también se diseñó el procedimiento para la especificación y validación de requisitos de software. En la segunda etapa se realizó la validación mediante dos pilotos de aplicación en dos unidades de análisis (empresas desarrolladoras de software de la ciudad de Pereira), la tercera etapa consistió en la verificación de los datos recolectados en la etapa inmediatamente anterior y finalmente en la cuarta etapa se realizaron los respectivos ajustes al procedimiento.

Al llevar a cabo una investigación desde los ámbitos regional, nacional e internacional respecto a los estudios adelantados sobre requisitos de software y, puntualmente en la especificación y validación de requisitos, la **Tabla No. 3** revela que desde el contexto regional las investigaciones y trabajos realizados al respecto son relativamente pocos y sus resultados no han sido lo suficientemente aplicados como para impactar de manera considerable las necesidades de la industria local del software.

Al analizar e interpretar la información recolectada de acuerdo a los temas establecidos en la **Tabla No.25** para la encuesta que se aplicó a la muestra por conveniencia de las MiPymes desarrolladoras de software de la Ciudad de Pereira, se descubren ciertos aspectos que es de gran importancia mencionar en esta discusión de resultados, por ejemplo para el **Tema No.1** relacionado con la formación profesional del personal encargado del desarrollo de los requisitos, se conoció que pocas de las empresas cuentan con un proceso bien definido para el desarrollo de los requisitos, además el desarrollo de los mismos es generalmente llevado a cabo por un ingeniero de sistemas que no está específicamente formado en el área de requisitos de software.

Con respecto al **Tema No.2** relacionado con la *elicitación* de requisitos resulta preocupante el hecho de que gran parte de la MiPymes no evalúan el grado de participación de los *stakeholders* mientras se desarrolla el proyecto, por lo que su intervención en el mismo es muy poca, además no se identifica el grado de dominio que poseen los involucrados sobre el problema que se desea solucionar, generando inconvenientes en el planteamiento de la solución deseada debido a la deficiente capacidad de comunicar explícitamente la necesidad que se quiere satisfacer.

En la misma línea, al ahondar en el **Tema No.5** sobre especificación de requisitos se evidenció que el uso del lenguaje natural para lograr la definición de los requisitos es familiar y de fácil uso para todos los involucrados en el proyecto y predomina sobre el uso de modelos formales. Sin embargo, el desconocimiento de una estructura clara que suministre los elementos necesarios para minimizar la presencia de ambigüedades e inconsistencias en los requisitos especificados en lenguaje natural, hace que por lo general el documento de especificación de requisitos de software SRS resulte incompleto y con gran número de defectos, lo que conllevaba finalmente a obtener un producto software que no cumple con las funcionalidades y criterios de calidad requeridos por el usuario para satisfacer sus expectativas.

También se puede observar que la especificación obtenida durante el desarrollo de los requisitos no es aprovechada por la gran mayoría de las empresas como base para el establecimiento de un contrato, lo que podría contribuir a una estimación de tiempo, costo y alcance que no corresponda con la realidad del proyecto una vez se ejecute.

Con base en el **Tema No.6** en el que se preguntó por la validación de requisitos, se permitió conocer que un porcentaje importante de las mismas no involucran al cliente en el proceso de validación de los requisitos especificados, lo cual se podría considerar como una de las principales causas de fracaso de los proyectos atribuibles a los requisitos.

Lo anterior, que está relacionado con el primer objetivo del proyecto, es evidencia de que se logró caracterizar la manera en que las MiPymes desarrolladoras de software de la ciudad de Pereira están llevando a cabo la especificación y validación de requisitos, consiguiendo además establecer un estado del arte de los antecedentes desde el ámbito regional, nacional e internacional. Como producto de ello, se obtuvo un artículo tipo B publicado en revista indexada de reconocimiento nacional, presentado en el **ANEXO I - Artículo publicado Entre Ciencia e Ingeniería No. 19**.

La selección de las técnicas del marco teórico, los antecedentes y las usadas por las MiPymes desarrolladoras de software de la ciudad de Pereira que fueron analizadas como insumo principal para el diseño del procedimiento PEVReS, se hizo en consenso entre los autores del proyecto y el asesor del mismo a través de juicio de expertos, identificando además sus principales ventajas y desventajas con la intención de argumentar su pertinencia en el procedimiento. Mediante este análisis se logró identificar los elementos de proceso relacionados con las actividades, técnicas, roles, guías, entre otros, requeridos para el procedimiento de especificación y validación de requisitos de software, dando cumplimiento al segundo objetivo del proyecto.

Por otro lado, la etapa de la metodología donde se diseñó el procedimiento tuvo como resultado un procedimiento estructurado a través de una secuencia de actividades puntuales, que establecen el uso de técnicas soportadas desde el referente teórico, el referente contextual y el diagnóstico de las empresas, el cual proporciona además un conjunto de guías, formatos y herramientas que lo diferencian de otras propuestas y que permiten a la persona encargada de especificar los requisitos, conseguir una definición correcta de los mismos en un documento SRS alineado con el procedimiento, basado en estándares mundialmente reconocidos y validado por el usuario. Por consiguiente, se da por cumplido el tercer objetivo del proyecto cuyo principal resultado fue el diseño del procedimiento incluyendo una especificación en SPEM.

Respecto a la segunda etapa relacionada con la aplicación del procedimiento, las **Tablas 33 y 39** correspondientes a los pilotos de aplicación 1 y 2 respectivamente reflejan que en ambos casos se presentó un aumento considerable en el tiempo empleado para el desarrollo de los requisitos, tal como se puede apreciar en la etapa de verificación de cada uno de los pilotos de aplicación. Este aumento de tiempo se puede atribuir tanto a la curva de aprendizaje del nuevo procedimiento, como a las nuevas técnicas que se aplican, que no eran usadas antes por las empresas.

Con relación a la tercera etapa en la que se realizó la verificación de los datos obtenidos una vez aplicado el procedimiento, se observa un aumento en la identificación de defectos en etapas

tempranas, y una reducción de defectos en etapas posteriores, lo que no solo representa una mayor calidad en el producto final, sino una disminución de costos y de tiempo en correcciones y reprocesos; compensando así el tiempo de más invertido en la especificación y validación de requisitos.

Así mismo, la ejecución de esta etapa tuvo como resultado una serie de mejoras y recomendaciones al procedimiento para aplicaciones futuras, producto de las dificultades encontradas durante su aplicación, como también niveles de satisfacción de 6 y 5 (satisfecho) respectivamente en cada una de las empresas, resaltando que con la aplicación del procedimiento no solo se lograron mejores resultados en el proceso de desarrollo de los requisitos, sino también satisfacción en el uso del procedimiento por parte de los desarrolladores. Tanto la encuesta de expectativas como la de rendimiento que ayudaron a determinar los niveles de satisfacción respecto al uso del procedimiento fueron analizadas mediante el *Alfa de Cronbach* para estimar la confiabilidad y validez de los instrumentos.

De la misma manera, en la cuarta etapa se logró obtener un procedimiento ajustado a las empresas en las cuales se llevaron a cabo los pilotos de aplicación, resaltando que se presentaron pocos errores durante la aplicación del procedimiento considerando todo lo nuevo propuesto por el mismo, y que las recomendaciones muestran apropiación del proceso por parte de las empresas. Sin embargo, se considera que al ser aplicado en otras organizaciones pueden surgir ajustes adicionales. Como mejora adicional se modeló el procedimiento a través de SPEM para una mejor presentación y despliegue del mismo, aunque su aplicación en las empresas hará parte del trabajo futuro.

Los resultados obtenidos en la segunda, tercera y cuarta etapa de la metodología del proyecto, dan cuenta del cumplimiento exitoso del cuarto objetivo del proyecto, pues logran evidenciar la validación del procedimiento en dos pilotos de aplicación representados en dos MiPymes desarrolladoras de software de la ciudad de Pereira.

Respecto a los resultados principales del proyecto relacionados con el diseño del procedimiento y su validación frente a los antecedentes y sus contenidos, se pudo observar mediante los pilotos de aplicación que personas con poca experiencia en las técnicas propuestas por el procedimiento lograron especificar y validar requisitos de software sin requerir capacitación adicional. Sin embargo, como el tiempo del proceso de desarrollo de los requisitos en las empresas se extendió debido a la curva de aprendizaje requerida para lograr un mejor dominio del procedimiento, sería necesario ahondar un poco más en la facilidad de uso del mismo y sus técnicas, con la intención de estudiar las dificultades que se presentaron y sobre las cuales se hicieron las recomendaciones y propuestas de mejora.

Con relación a la metodología propuesta por (De la Cruz Londoño & Castro Guevara, 2014) principal antecedente del proyecto, se puede indicar que a diferencia del procedimiento elaborado, ésta no especifica en detalle las actividades y pasos requeridos para conseguir un documento de especificación de requisitos estructurado y validado por el usuario ni suministra una relación precisa entre las entradas y salidas de cada una de las etapas propuestas para el desarrollo de los requisitos. Además, para la validación de la metodología se usó como instrumento el juicio de expertos, mientras que el procedimiento PEVReS, a diferencia de otras propuestas presentadas en los antecedentes, tuvo una aplicación real en dos MiPymes

desarrolladoras de software de la ciudad de Pereira, lo que permitió lograr un procedimiento validado por los usuarios, más ajustado a las necesidades de las mismas y sin requerir de personas expertas en el área de requisitos al momento de su aplicación, superando algunas de las falencias identificadas en el referente contextual.

9. CONCLUSIONES

El trabajo realizado para lograr el Procedimiento para especificar y validar requisitos de software en MiPymes desarrolladoras de software de la ciudad de Pereira, basado en estudios previos en la región, permitió establecer las siguientes conclusiones:

- La encuesta realizada a la muestra por conveniencia permitió conocer aspectos importantes sobre el proceso de requisitos en las empresas desarrolladoras de software de la ciudad de Pereira, lo cual junto con el análisis de las técnicas obtenidas del referente contextual, sirvió de base para el desarrollo del procedimiento pero también podría servir para otros estudios y trabajos futuros.
- Se logró validar el procedimiento mediante la aplicación de dos pilotos de aplicación en dos MiPymes desarrolladoras de software de la ciudad de Pereira, consiguiendo que ambas refinaran su proceso de definición y validación de requisitos.
- La aplicación del procedimiento en las empresas partícipes de la validación logró que los equipos responsables del desarrollo de los requisitos alcanzaran niveles de satisfacción que dan cuenta de los buenos resultados obtenidos en sus proyectos, quedando satisfechos por los resultados percibidos.
- Para los dos pilotos de aplicación llevados a cabo, la aplicación del procedimiento logró encontrar errores en etapas tempranas mediante el uso de las técnicas propuestas por el mismo, lo cual contribuyó a una mejor calidad del producto, siendo más cercano a los requisitos.
- La realización de las actividades propuestas para especificar y validar requisitos ayudó a la obtención de un documento de especificación de requisitos (SRS) en lenguaje natural validado por el usuario, a diferencia de la especificación que hacían antes las empresas que no quedaba en un documento ni era validada por el usuario.
- Uno de los factores de éxito en la aplicación del procedimiento fue el uso de las técnicas, guías y formatos que apoyaron la especificación y validación de los requisitos, de tal manera que el esfuerzo de los miembros del equipo solo se centró en la ejecución de los diferentes pasos que propone cada técnica definida en el procedimiento.
- Al encontrar la mayor cantidad de defectos en el desarrollo de los requisitos se reduce notablemente el número de defectos encontrados en la etapa de producción atribuibles a los requisitos, aunque se incrementó el tiempo empleado en el desarrollo de los mismos.
- La validación del procedimiento a través de los pilotos de aplicación contribuyó a realizar los ajustes pertinentes al procedimiento propuesto gracias a las lecciones aprendidas y mejoras recomendadas por las personas encargadas de su aplicación.

- Resulta importante refinar el procedimiento para la validación de requisitos, pues en los proyectos intervenidos mediante los pilotos de aplicación siguen existiendo defectos en la etapa de producción atribuibles a los requisitos.
- Finalmente, y de acuerdo a la discusión de resultados presentada, se considera que los objetivos propuestos por el proyecto fueron alcanzados satisfactoriamente y contribuyeron a la mejora de procesos en las empresas en las cuales se aplicó el procedimiento, gracias a su alineación con cada una de las etapas de la metodología.

10. RECOMENDACIONES

- Que para el proceso de desarrollo de requisitos se tenga total involucramiento y compromiso por parte del usuario.
- Se debe tener una planificación previa a la especificación y validación de requisitos.
- Mientras se adquiere la curva de aprendizaje por la aplicación del nuevo procedimiento, se recomienda estimar un poco más de tiempo para la fase de requisitos.
- Que se ejecuten a cabalidad las actividades propuestas por el procedimiento, para lograr un documento de especificación acorde al formato establecido por el procedimiento.
- El documento de especificación de requisitos debe ser siempre validado por el usuario de acuerdo a las técnicas establecidas en el procedimiento.
- Cuando se necesite un mayor nivel de detalle o entendimiento sobre los requisitos del negocio, del usuario o del sistema, se recomienda hacer uso de las técnicas recomendadas u opcionales que propone el procedimiento.
- Hacer uso de la gestión de requisitos como complemento a la aplicación del procedimiento elaborado.
- Como trabajo futuro, se plantea que puedan ser elaborados procedimientos que apoyen la *elicitación* de requisitos como etapa previa a la especificación.

11. REFERENCIAS BIBLIOGRÁFICAS

- Abarca Romero, M., Arisaca Mamani, R., & Dávila Ramón, A. (2015). Implementación del Perfil Básico de la ISO/IEC 29110 de una Pequeña Empresa Desarrolladora de Software: Lecciones Aprendidas. Arequipa, Perú.
- Ambler, S. W. (2014). *An Introduction to Agile Modeling*. Obtenido de Agile modeling: <http://www.agilemodeling.com/essays/introductionToAM.htm>
- Arboleda, H., & Casallas, R. (2004). QualDev Process: Procesos Adaptables de Desarrollo de Software para Proyectos Ágiles. *Journal INGENIERIA Y COMPETITIVIDAD - Facultad de Ingeniería, Universidad del Valle*, 6(2), 64-74. Recuperado el 09 de 10 de 2015, de <http://revistaingenieria.univalle.edu.co:8000/index.php/incompe/article/view/90/89>
- Arias Chaves, M. (7 de Julio de 2006). *La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software*. Recuperado el 06 de Junio de 2014, de Portal de revistas académicas de la Universidad de Costa Rica: <http://www.intersedes.ucr.ac.cr/ojs/index.php/intersedes/article/viewFile/119/118>
- Armstrong, G., & Kotler, P. (2013). *Fundamentos de marketing* (2013 ed.). (G. D. Chávez, Ed.) Naucalpan de Juárez, Mexico.
- Báez, M. G., & Barba Brunner, S. I. (2001). Metodología DoRCU para la Ingeniería de Requerimientos. *Workshop em Engenharia de Requisitos*, 210-222. Obtenido de Pontificia Universidade Católica do Rio de Janeiro: http://www.inf.puc-rio.br/wer/WERpapers/pdf_counter.lua?wer=WERO1&file_name=baez.pdf
- Basharat, I., Fatima, M., Nisa, R., Hashim, R., & Khanum, A. (2013). Requirements engineering practices in small and medium software companies: An empirical study. *2013 Science and Information Conference*, 218 - 222.
- BIOESTADISTICO. (05 de 12 de 2014). *Alfa de Cronbach, índice de consistencia interna*. Recuperado el 15 de 12 de 2016, de Bioestadístico: <http://bioestadistico.com/>
- BIOESTADISTICO. (05 de 12 de 2014). Alfa de Cronbach, índice de consistencia interna. Perú.
- Boehm, B. (1981). *Software Engineering Economics*. New Jersey: Prentice Hall.
- Burgos Pintos, A., & Garbarino de la Rosa, S. M. (2010). *Agile modeling*. Obtenido de Agile modeling: http://osl2.uca.es/wikiCE/index.php/Agile_modeling
- Burnay, C., Jureta, I. J., & Faulkner, S. (2014). What stakeholders will or will not say A theoretical and empirical study of topic importance in Requirements Engineering elicitation interviews. *Information Systems*, 46, 61-81.
- Camacho Zambrano, A. N. (2005). Herramienta para el análisis de requerimientos dentro de la pequeña empresa desarrolladora de software en Bogota. Bogota, Cundinamarca.
- Carrizo Moreno, D. (2012). Comparación de efectividad de las técnicas de educación de requisitos software vision novel y experta. *Ingeniare. Revista chilena de ingeniería*, 20(3), 386-397.
- Casallas, R. (2007). Algunos mitos y desafíos de la Ingeniería de Software. *Revista Sistemas ACIS*, 8-17. Recuperado el 09 de 10 de 2015, de http://correo.acis.org.co/fileadmin/Revista_102/columnista.pdf

- CEPAL. (2009). *Desafíos y oportunidades de la industria del software en América Latina*. Bogotá: Mayol Ediciones.
- Chikh, A., & Alajmi, H. (2014). Towards a dynamic software requirements specification. *Computer Applications and Information Systems (WCCAIS), 2014 World Congress on*, 1-7.
- Correa Botero, F., & González Bedoya, G. (2009). Guía para implementar buenas prácticas en las áreas de procesos de gestión de requerimientos y planeación del proyecto para las microempresas desarrolladoras de software, basada en CMMI. Pereira, Risaralda, Colombia.
- Cristiá, M. (2011). *Introducción a la Ingeniería de Requerimientos*. Recuperado el 20 de Junio de 2014, de Facultad de Ciencias Exactas, Ingeniería y Agrimensura | Universidad Nacional de Rosario:
<http://www.fceia.unr.edu.ar/~mcristia/publicaciones/ingreq-a.pdf>
- Dávila, N. D. (2001). Ingeniería de requerimientos: Una guía para extraer, analizar, especificar y validar los requerimientos de un proyecto. Montevideo, Uruguay. Obtenido de
<http://u.jimdo.com/www21/o/s3ca10756ba149b19/download/m895f5b4d9cd5f0b1/1251925731/ingreq.pdf>
- Davyt Dávila, N. (2001). *Ingeniería de Requerimientos: una guía para extraer, analizar, especificar y validar los requerimientos de un proyecto (tesis de pregrado)*. Obtenido de UNIVERSIDAD ORT URUGUAY:
<http://u.jimdo.com/www21/o/s3ca10756ba149b19/download/m895f5b4d9cd5f0b1/1251925731/ingreq.pdf>
- De la Cruz Londoño, C. A., & Castro Guevara, G. A. (2014). *Metodología para la adquisición y gestión de requerimientos en el desarrollo de software para pequeñas y medianas (tesis de posgrado)*. Pereira: Universidad Tecnológica de Pereira.
- Dorigan, J. A., & De Barros, R. M. (2014). A Process Model for Standardization and Increase in the Requirements Quality. *IEEE LATIN AMERICA TRANSACTIONS*, 12(8), 1502-1507.
- Durán Toro, A., & Bernárdez Jiménez, B. (Octubre de 2001). *Metodología para la elicitación de Requisitos de Sistemas Software*. (U. d. Sevilla, Ed.) Recuperado el 04 de 05 de 2016, de Universidad de Sevilla:
<https://www.lsi.us.es/docs/informes/lsi-2000-10.pdf>
- El Congreso de Colombia. (10 de 07 de 2000). *Ley 590 de 2000*. Recuperado el 14 de 09 de 2016, de Supersociedades:
<http://www.supersociedades.gov.co/superintendencia/normatividad/Documents/Ley%20590%20de%20000.pdf>
- FEDESFT. (2013). *Estudio para la caracterización del sector de productos y servicios de la industria de software y servicios asociados*. Bogotá: Fedesoft.
- García Ramírez, F., & Puello Marrugo, P. (2010). Gestión de Requisitos en la Ingeniería del Software. *INGENIATOR | REVISTA VIRTUAL DE LOS PROGRAMAS DE INGENIERÍA | UNIVERSIDAD DE SAN BUENAVENTURA*, 1(1), 57-65.
- IEEE. (1993). *IEEE Software Engineering Standard: Glossary of Software Engineering Terminology*. Recuperado el 01 de 07 de 2014, de IEEE Computer Society:
http://dis.unal.edu.co/~icasta/GGP/_Ver_2012_1/Documentos/Normas/610-12-1990.pdf
- IEEE. (2004). *SWEBOK*. Piscataway: IEEE.

- IEEE. (2014). *SWEBOK Guide V3.0*. Piscataway: IEEE.
- IEEE Computer Society. (1990). *Std 610.12-1990*. Recuperado el 01 de 07 de 2014, de IEEE Standard Glossary of Software Engineering Terminology:
http://dis.unal.edu.co/~icasta/GGP/_Ver_2012_1/Documentos/Normas/610-12-1990.pdf
- INTECO. (2008). *Guía práctica de gestión de requisitos*. Recuperado el 21 de Junio de 2014, de Instituto Nacional de Tecnologías de la Comunicación | Ministerio de industria, energía y comercio del Gobierno de España:
https://www.inteco.es/file/NRDmviQoTbI_jZcyjTYRlw
- IREB. (1 de marzo de 2015). *Programa de Estudios Profesional Certificado en Ingeniería de Requisitos de IREB*. Obtenido de https://www.ireb.org/content/downloads/2-syllabus-foundation-level/ireb_cppe_syllabus_fl_es_v22.pdf
- ISO. (2012). *ISO/IEC 29110:2012 Software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 1: Overview*. ISO.
- ISO. (2012). *ISO/IEC 29110:2012 Software engineering — Lifecycle profiles for Very Small Entities (VSEs) — Part 5-1-2: Management and engineering guide: Generic profile group: Basic profile*. ISO.
- Jacobson, I., Spence, I., & Bittner, K. (2013). *Casos de uso 2.0*. Suecia: 3 IVAR JACOBSON INTERNATIONAL SA. Obtenido de https://www.ivarjacobson.com/sites/default/files/field_iji_file/article/use_case_2.0_-_spanish_translation.pdf
- Jeffries, R., Anderson, A., & Hendrickson, C. (2001). *Extreme Programming Installed*. Boston: Addison-Wesley.
- Khankaew, S., & Riddle, S. (2014). A review of practice and problems in requirements engineering in small and medium software enterprises in Thailand. *IEEE 4th International Workshop on Empirical Requirements Engineering (EmpiRE)*, 1-8.
- Laporte, C. (2010). *Contributions to Software Engineering and to the Development and Deployment of International Software Engineering Standards for Very small Entities*. Brest, Francia: UNIVERSITÉ DE BRETAGNE OCCIDENTALE.
- Laporte, C. Y., April, A., & Renault, A. (2006). *Applying ISO/IEC Software Engineering Standards in Small Settings: Historical Perspectives and Initial Achievements*. Luxembourg: In Proceedings of SPICE 2006 Conference.
- Lawrence Pfleeger, S. (2002). *Ingeniería del software teoría y práctica*. Buenos Aires: Prentice Hall.
- Lema Becerra, D., & Rodas Valencia, L. F. (2012). Análisis y Diseño de una herramienta gráfica para los procesos de ingeniería de requisitos. Pereira, Risaralda.
- Letelier Torres, P., & Sánchez López, E. A. (2006). Metodologías Ágiles en el Desarrollo de Software. *Ciencia y Técnica Administrativa*, 5(26), 1-7.
- Lopera Carmona, D. C. (2012). *Análisis estratégico de la industria colombiana de software a partir de la simulación de escenarios de competencia utilizando Dinámica de Sistemas*. Recuperado el 23 de 05 de 2014, de Universidad Nacional de Colombia: <http://www.bdigital.unal.edu.co/8466/1/39175215.2012.pdf>

- Maita, Y. (2012). *Red colaborativa postgrado Universidad Central de Venezuela*. Obtenido de Los Requerimientos y su importancia en el desarrollo del Software:
http://kuainasi.ciens.ucv.ve/red_educativa/blogs/20?language_id=1
- Martínez Vaca, B., & Ospina Arredondo, C. A. (2011). Proceso para la formulación de iniciativas de requerimientos de software a partir de los objetivos estratégicos organizacionales (tesis de maestría). Santiago de Cali, Valle del Cauca, Colombia. Obtenido de
https://bibliotecadigital.icesi.edu.co/biblioteca_digital/bitstream/10906/68004/1/proceso_formulacion_software.pdf
- Menéndez Domínguez, V. H., & Castellanos Bolaños, M. E. (2015). SPEM: Software Process Engineering Metamodel. *Revista Latinoamericana de Ingeniería de Software*, 92-100.
- Merchan, L., Urrea, A., & Rebollar, R. (2008). Definición de una metodología ágil de ingeniería de requerimientos para empresas emergentes de desarrollo de software del sur-occidente colombiano. *Revista Científica Guillermo de Ockham, Universidad de San Buenaventura*, 6(1), 37-50.
- Oberg, R., Probasco, L., & Ericsson, M. (2003). *Applying requirements management with use cases*. Obtenido de Rational Software Corporation: <http://www.uml.org.cn/RequirementProject/pdf/apprmuc.pdf>
- OMG. (2002). *"Software Process Engineering Metamodel (SPEM) Specification" version 1.0*. Recuperado el 26 de 12 de 2016, de Object Management Group: <http://www.omg.org>
- Pagani, J. B. (2013). *¿Requisitos o Requerimientos?* Recuperado el 15 de 12 de 2015, de Investigación IT: <http://investigacionit.com.ar/requisitos-o-requerimientos/>
- Peláez Valencia, L. E. (2010). SWEBOK – IEEE | Guide to the Software Engineering Body of Knowledge; Un resumen ejecutivo. Pereira, Risaralda, Colombia.
- Peláez Valencia, L. E. (2011). *Propuesta metodológica para estandarizar el proceso de construcción y evaluación del producto software que permita a las PyMES medir la calidad del software*. Pereira.
- Pérez Huebe, M. (2005). *Ingeniería de Requerimientos*. Pachuca, Mexico.
- Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley Longman Publishing Co.
- Pressman, R. S. (2002). *Ingeniería del Software. Un enfoque práctico* (Quinta edición ed.). Madrid, España: McGraw-Hill.
- Pressman, R. S. (2006). *Ingeniería del software, un enfoque práctico* (Sexta edición ed.). Madrid, España: McGraw-Hill.
- Pressman, R. S. (2010). *Ingeniería del Software Un enfoque práctico* (Septima edición ed.). Mexico: McGraw-Hill.
- Quispe, A., Marquez, M., Silvestre, L., Ochoa, S. F., & Robbes, R. (2010). *Requirements Engineering Practices in Very Small Software Enterprises: A Diagnostic Study*. Recuperado el 06 de Mayo de 2016, de Departamento de Ciencias de la Computación Universidad de Chile: <http://users.dcc.uchile.cl/~rrobbes/p/SCCC2010-requirements.pdf>

- RAE. (2014). *Requerimiento*. Recuperado el 15 de 12 de 2015, de Diccionario de la Real Academia Española: <http://buscon.rae.es/drae/srv/search?val=requiremimiento>
- Rangel Grisales, M., & Peláez Betancur, V. (2013). Métodos de obtención de requerimientos y sus enfoques de selección (tesis de pregrado). Pereira, Risaralda, Colombia.
- Salazar Osorio, H. G., Rengifo Romero, H. J., Machuca Villegas, L. E., & Aranda Bueno, J. A. (2012). UNA GUÍA GENERAL PARA LA ESPECIFICACIÓN Y VERIFICACIÓN FORMAL DE REQUERIMIENTOS USANDO EVENT-B™ Y RODIN™. *Revista Educación en Ingeniería*, 7(14), 82-91.
- Sommerville, I. (2005). *Ingeniería del Software* (Séptima Edición ed.). Madrid, España: Pearson Educación S.A.
- Sommerville, I. (2011). *Ingeniería de Software*. Madrid: Pearson.
- Sumano López, M. D. (2001). *Áncora: Metodología para el Análisis de Requerimientos de Software conducente al Reuso*. Obtenido de Universidad Veracruzana: <http://www.uv.mx/personal/asumano/files/2010/07/Guia.pdf>
- SUPERSOCIEDADES. (2015). *Desempeño del sector software 2012-2014*. Bogotá: Superintendencia de sociedades. Obtenido de <http://www.supersociedades.gov.co/noticias/Documents/2015/EE1-%20Estudio%20Sectorial%20Software-%202015%20VII%2030.pdf>
- Thayer, R., & Dorfman, M. (2000). *Software Requirements Engineering* (Segunda edición ed.). Los Alamitos, California: IEEE Computer Science Press.
- Thayer, R., & Dorfman, M. (1997). *Software Requirements Engineering* (Segunda edición ed.).
- Torres de Paz, R. M. (2009). *Departamento de Lenguajes y Sistemas Informáticos Universidad de Sevilla*. Recuperado el 18 de 05 de 2014, de El proceso de Ingeniería de Requisitos en el ciclo global del software: <http://www.lsi.us.es/docs/doctorado/memorias/memo-inv-rosa-m-torres.pdf>
- UNCTAD. (2012). *Informe sobre la economía de la información 2012*. Suiza: Naciones Unidas.
- Valencia, M. E. (1989). Guía para la preparación de un documento de requerimientos. Santiago de Cali, Valle del Cauca, Colombia.
- Varela Galvis, A., & Arango Sterling, G. E. (2012). *Instrumento para la generación del proceso de desarrollo de requerimientos de software para micro y pequeñas empresas (tesis de maestría)*. Recuperado el 13 de Octubre de 2015, de Biblioteca Digital Universidad ICESI: http://bibliotecadigital.icesi.edu.co/biblioteca_digital/bitstream/10906/70626/1/instrumento_generacion_proceso.pdf
- Wautelet, Y., Kolp, M., & Achbany, Y. (2006). *S-Tropos: An iterative SPEM-Centric software project management process*. Recuperado el 26 de 12 de 2016, de Louvain School of Management Research Institute: http://www.isys.ucl.ac.be/staff/youssef/Articles/WP_SPEM.pdf
- Wieggers, K., & Beaty, J. (2013). *Software Requirements* (Third Edition ed.). Redmon, Washington: Microsoft Press.
- Yang, Y., He, M., Li, M., Wang, Q., & Barry, B. (2008). Phase distribution of software Development Effort. *ESEM '08 Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*, 61-69.

Zapata J, C. M., & Arango Isaza, J. F. (2004). Alineación entre metas organizacionales y elicitación de requisitos del software. *Revista Dyna*, 71(143), 101-110.

12. ANEXOS

ANEXO A – Encuesta empresas

ANEXO B - Resultados encuesta

ANEXO C – Procedimiento para especificar requisitos de software

ANEXO D – Formato documento SRS

ANEXO E – Procedimiento para validar requisitos de software

ANEXO F – Encuesta expectativas

ANEXO G – Encuesta rendimiento

ANEXO H - Presentación Procedimiento PEVReS

ANEXO I - Artículo publicado Entre Ciencia e Ingeniería No. 19, realizado a partir de los antecedentes y marco teórico como parte del proceso de investigación.

ANEXO J – Procedimiento PEVReS modelado bajo SPEM.