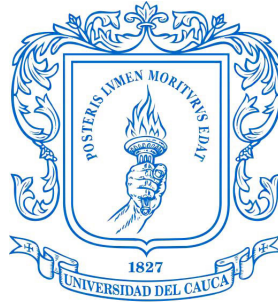


Gestión de proyectos de desarrollo de software a través de la norma ISO 9001 y SCRUM



Trabajo de grado

Miguel Andrés Morcillo Mosquera

Director: PhD. MSc. César Jesús Pardo Calvache¹
Codirector: PhD. MSc. Francisco José Pino Correa²

Universidad del Cauca

Facultad de Ingeniería Electrónica y Telecomunicaciones

Departamento de Sistemas

¹Grupo de I+D en Tecnologías de la información (GTI)

²Grupo de I+D en Ingeniería de Software (IDIS)

Popayán, enero de 2019

Índice

Capítulo I. Introducción	7
1.1. Problemática	7
1.2. Objetivos	10
1.2.1. Objetivo general	10
1.2.2. Objetivos específicos	10
1.3. Estrategia de la Investigación	11
1.4. Estructura del documento	12
Capítulo II. Marco teórico y estado del arte	14
2.1. Marco Teórico	14
2.1.1. Actualidad de la industria del desarrollo de software	14
2.1.2. Enfoques ágiles	16
2.1.3. SCRUM	17
2.1.3.1. ¿Por qué SCRUM y no XP?	18
2.1.4. ISO 9001	21
2.2. Estado del Arte	22
2.3. Revisión sistemática de la literatura	22
2.3.1. Foco de la pregunta	23
2.3.2. Pregunta de investigación	23
2.3.3. Planeación de la revisión sistemática de la literatura	23
2.3.4. Resultados de la revisión sistemática de la literatura	25
Capítulo III. Armonización	34
3.1. Homogeneización de Enfoques	34
3.1.1. Adquisición de conocimiento de los enfoques involucrados	35
3.1.2. Análisis de estructura y terminología	36
3.1.2.1. ISO 9001	36
3.1.2.2. SCRUM	38
3.1.3. Homogeneización de los enfoques usando una plantilla común de elementos de proceso	39
3.1.3.1. Homogeneización del estándar ISO 9001:2015	39
3.1.3.2. Homogeneización del enfoque ágil SCRUM	42
3.2. Comparación de enfoques	43
3.2.1. Diseño de la comparación	43
3.2.1.1. Establecer las entidades a ser comparadas en función de las necesidades de investigación	44
3.2.1.2. Definir el grado de relación	44
3.2.1.3. Fijar la direccionalidad de la comparación	46
3.2.1.4. Definir una plantilla de comparación	46

3.2.2.	Comparación entre ISO 9001:2015 y SCRUM	46
3.2.2.1.	Comparación de entidades	47
Capítulo IV. Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015		51
4.1.	Análisis de la conformidad entre SCRUM e ISO 9001:2015	51
4.1.1.	Capítulo 8. Operación	51
4.1.2.	Capítulo 9. Evaluación del desempeño	52
4.1.3.	Capítulo 10. Mejora	53
4.2.	Consideraciones y alternativas para lograr la conformidad en los requisitos no relacionados	53
4.3.	Proceso integrado para la gestión de proyectos ágiles con SCRUM e ISO 9001:2015	54
4.3.1.	Proceso de Diseño y Desarrollo	54
4.3.2.	Proceso de Planeación del Sprint	59
4.3.3.	Proceso de Ejecución del Sprint	64
4.3.4.	Proceso de Fabricación del Código	68
Capítulo V. Estudio de caso		72
5.1.	Diseño	72
5.2.	Sujetos y unidades de análisis	72
5.3.	Procedimiento de campo y recolección de datos	73
5.3.1.	Etapas de adaptación	74
5.3.1.1.	Mes 1	74
5.3.1.2.	Mes 2	75
5.3.1.3.	Mes 3	75
5.3.1.4.	Mes 4	75
5.3.2.	Etapas de aplicación	76
5.3.2.1.	Inicio del proyecto	77
5.3.2.2.	Ejecución del proyecto	77
5.3.2.3.	Puesta en marcha	79
5.4.	Análisis de resultados y lecciones aprendidas	79
5.4.1.	Esfuerzo	79
5.4.2.	Encuesta de conformidad con el proceso	80
5.4.3.	Factores de éxito y fracaso identificados	82
5.4.4.	Beneficios y lecciones aprendidas	84
5.5.	Plan de validación	85
5.6.	Limitaciones	85
Capítulo VI. Conclusiones y trabajos futuros		86
6.1.	Análisis de objetivos de investigación	86

6.2. Conclusiones	87
6.3. Trabajos futuros	88
Referencias	90
Anexos	95
Anexo A. Tablas de Homogeneización	95
Anexo B. Modelos de Procesos en BPMN	118
Anexo C. Relaciones identificadas en la comparación de enfoques	122
Anexo D. Revisión sistemática acerca de la integración del estándar ISO 9001 con enfoques ágiles de desarrollo de software	131

Índice de figuras

1.	Proceso SCRUM en BPMN. Fuente: propia.	20
2.	Publicaciones por año.	27
3.	Estudios por país.	28
4.	Grado de relación de la comparación. Fuente: propia.	45
5.	Conformidad de SCRUM con respecto al capítulo 8 de ISO 9001:2015 C_3C_8	48
6.	Conformidad de SCRUM con respecto al capítulo 9 de ISO 9001:2015 C_3CH_9	49
7.	Conformidad de SCRUM con respecto al capítulo 10 de ISO 9001:2015 C_3CH_{10}	50
8.	Captura de pantalla del reporte del sprint 4 en la herramienta de gestión. Fuente: propia.	78
9.	Esfuerzo de <i>La empresa</i> durante la etapa de adaptación. Fuente: propia.	80
10.	Resultados de la encuesta de evaluación del proceso. Fuente: propia.	82
11.	Modelo del proceso <i>Planeación del Sprint</i>	119
12.	Modelo del proceso <i>Ejecución del Sprint</i>	120
13.	Modelo del proceso <i>Fabricación del código</i>	121

Índice de tablas

1.	Palabras clave relacionadas con la pregunta de investigación.	24
2.	Cadenas de búsqueda.	25
3.	Estudios primarios seleccionados.	27
4.	Porcentaje de Modelos, enfoques, estándares y/o metodologías analizadas. . .	29
5.	Tipos de empresa en los estudios de caso.	30
6.	Técnicas de Armonización utilizadas.	30
7.	Factores de éxito y fracaso.	33
8.	Roles definidos por HoMethod [35].	35
9.	Comparación de atributos de ISO 9001:2015 y SCRUM.	35
10.	(WER) Sintaxis para identificar los requerimientos en ISO 9001:2015. Fuente: [35]	38
11.	Elementos comunes de estructuras de procesos. Fuente: [35]	39
12.	Extracto de Homogeneización del Capítulo 8. Operación, de ISO 9001:2015.	41
13.	Extracto de la homogeneización de los roles de SCRUM.	43
14.	Nomenclatura de comparación. Fuente: propia.	45
15.	Plantilla de Comparación. Fuente: propia.	46
16.	Comparación de actividades de los requisitos del capítulo 8 de ISO 9001:2015 con SCRUM.	48
17.	Comparación de actividades de los requisitos del capítulo 9 de ISO 9001:2015 con SCRUM.	49
18.	Comparación de actividades de los requisitos del capítulo 10 de ISO 9001:2015 con SCRUM.	50
19.	Proceso de Diseño y Desarrollo	58
20.	Proceso de Planeación del Sprint	63
21.	Proceso de Ejecución del Sprint	67
22.	Proceso de Fabricación del código	71
23.	Características de la organización objeto del estudio de caso. Fuente: [35] . .	73
24.	Equivalencia entre el proceso integrado propuesto en este proyecto y el proceso definido por <i>La empresa</i> estudio de caso. Fuente: propia.	76
25.	Equivalencia entre el proceso definido por <i>La empresa</i> y el proceso aplicado en el proyecto piloto. Fuente: propia.	77
26.	Escala de <i>Likert</i> para la encuesta de evaluación. Fuente: propia.	80
27.	Premisas de la encuesta de evaluación del proceso. Fuente: propia.	81
28.	Factores de éxito y fracaso en el estudio de caso durante las etapas de Adaptación y Aplicación. Fuente: propia.	84
29.	Homogeneización del Capítulo 8. Operación, de ISO 9001:2015.	102
30.	Homogeneización del Capítulo 9. Evaluación del desempeño, de ISO 9001:2015.	105

31.	Homogeneización del capítulo de <i>10. Mejora</i> , de ISO 9001 en los elementos de proceso de la estructura común.	106
32.	Homogeneización de los roles de SCRUM.	109
33.	Homogeneización de los Eventos de SCRUM.	115
34.	Homogeneización de los artefactos de SCRUM.	118

Capítulo I. Introducción

1.1. Problemática

Un sistema de gestión de la calidad en cualquier organización, es un apoyo al direccionamiento estratégico y funciona como herramienta de gestión para evaluar y dirigir el desempeño de la organización y los procesos de la misma en términos de satisfacción del cliente y partes interesadas, y en el marco de los objetivos estratégicos.

La definición, institucionalización, documentación y control de los procesos de una organización a partir de un estándar, modelo o metodología para el desarrollo de software busca conseguir productos medibles y por lo tanto mejorables [1]. Además, permite a las organizaciones satisfacer la demanda actual de calidad y mejora continua implementando las mejores prácticas organizacionales y de ingeniería [2]. De esta manera, la implementación de un sistema de gestión de la calidad brinda a las organizaciones la capacidad de proporcionar regularmente productos y servicios que satisfacen los requisitos del cliente, y de abordar riesgos y oportunidades asociadas al contexto de mercado y a los objetivos de la organización, consiguiendo así una ventaja competitiva de cara al cliente y accediendo a nuevos nichos de mercado [3]. El impacto que tiene la gestión de la calidad sobre el producto final también se ve reflejado en los costos del proyecto, incluyendo una reducción en los errores, re procesos, mantenimiento, entre otros [4].

Dada la necesidad de mejores prácticas para los procesos misionales y organizacionales en el desarrollo de software, ha surgido un amplio abanico de estándares y modelos que dan soporte a diferentes necesidades en función del tamaño de la organización, la meta de calidad y objetivos estratégicos; algunos ejemplos de los modelos y estándares más implementados para la gestión de los procesos en el desarrollo de software son ISO¹/IEC² 12207 [5], ISO/IEC 15504 [6], ISO 9001 [3], CMMI-DEV³ [7], entre otros (en adelante: enfoques tradicionales).

Como se evidencia en [8], las micro, pequeñas y medianas empresas (MiPyMEs) representan el mayor porcentaje de la industria del software a nivel mundial, sin embargo, los enfoques tradicionales mencionados no han sido diseñados para ser aplicados a este tipo de organizaciones dado que su implementación implica una extensa documentación y gran esfuerzo en las etapas de planeación y diseño [9], siendo una situación difícil para las MiPyMEs por sus particularidades como la inestabilidad de los requerimientos, procesos caóticos, poco personal y la necesidad de crear valor de negocio rápidamente, resultando estos modelos inflexibles, costosos y complejos, no solo por el precio asociado a sus

¹ Por sus siglas en inglés: International Organization for Standardization.

² Por sus siglas en inglés: Comisión Electrotécnica Internacional.

³ Integración de modelos de madurez de capacidades para el desarrollo de softwar, programa de mejora de procesos del CMMI Institute.

certificaciones sino también a la inversión de tiempo y esfuerzo en talento humano, consultorías, auditorías y capacitaciones de personal, entre otros factores [10] [11].

Como alternativa a los modelos de referencia mencionados, algunas organizaciones han decidido implementar enfoques ágiles, así como: SCRUM [12], Lean Software Development (LSD) [13], Agile Unified Process (AUP) [14], Kanban [15], Método de desarrollo de sistemas dinámicos (DSDM) [16], Cristal Clear [17], eXtreme Programming (XP) [18] y Desarrollo de Software Adaptativo (ASD) [19], entre otros. De estos enfoques se puede decir que son menos exhaustivos respecto a la documentación y burocracia, más efectivos y flexibles en los entornos característicos de las MiPyMEs y agregan valor integrando al cliente en el proceso de desarrollo del producto software [20]. Además, los enfoques ágiles se basan en un conjunto de principios definidos en el Manifiesto Ágil⁴, los cuales buscan conseguir principalmente: la satisfacción del cliente, disminución de errores, disminución del tiempo de desarrollo y de entrega, flexibilidad ante los requisitos cambiantes, entre otros [21]. La implementación de enfoques ágiles en las organizaciones permite que las prácticas de desarrollo sean más livianas y sobre todo, enfocadas a proveer un proceso más eficiente, con entregas tempranas de valor, mayor comunicación, colaboración, e iterativo e incremental que promueva el pensamiento evolutivo y los procesos adaptativos [22]. Sin embargo, como se menciona en [23] y [24], los enfoques ágiles también presentan algunas debilidades así como: (I) dificultad en el mantenimiento de productos software a largo plazo debido a la poca documentación, (II) incertidumbre y desconocimiento en prácticas claras y detalladas sobre cómo abordar la mejora del proceso y (III) la aparente incompatibilidad con enfoques tradicionales de calidad [2] [21] [10] [22] [23] [24] [25] [26] [27] [28] [29] [30].

Por otra parte, las organizaciones de desarrollo de software además de gestionar la calidad del proceso y producto software, también requieren gestionar la calidad de sus procesos misionales y operativos, para esto uno de los modelos de referencia más implementados es el estándar ISO 9001, con más de 1 millón de organizaciones certificadas alrededor del mundo y 12000 en Colombia [31], el estándar cuenta con 30 años de experiencia en sistemas de gestión de la calidad y ha sido mejorado y actualizado en 5 oportunidades, siendo la versión más reciente el estándar ISO 9001:2015.

Actualmente, y como resultado de la realización de una revisión sistemática de la literatura, fue posible observar que existen algunas iniciativas donde se lleva a cabo la integración o identificación de relaciones entre prácticas de modelos para la gestión de organizaciones como el estándar ISO 9001 y enfoques ágiles, por ejemplo: (I) integración de la norma ISO 9001:2000 y XP [32], (II) aplicación de conceptos de enfoques ágiles como XP y SCRUM en modelos como CMM-DEV nivel 2 e ISO 9001:2000 [30] [33], (III) la aplicación del método de armonización XP Maturity Model (XPMM) para la incorporación de XP a grandes organizaciones certificadas en ISO 9001:2000 utilizando el esquema CMM [23],

⁴ El Manifiesto para el desarrollo ágil de software, <http://agilemanifesto.org>

(IV) migración de procesos basados en enfoques tradicionales o *plan-driven* a enfoques ágiles como SCRUM y XP [4] y (v) implementación ISO 9001:2008, SCRUM y herramientas específicas de código abierto [22], entre otros. Sin embargo, ha sido posible notar que en la integración de las prácticas no se lleva a cabo un análisis claro y exhaustivo que permita conocer el nivel de compatibilidad entre los enfoques estudiados. Asimismo, se puede observar que la literatura se enfoca en mayor medida a grandes organizaciones que ya han tenido experiencia con enfoques tradicionales o *plan-driven* [2] [34], además con respecto a los enfoques ágiles predomina XP, que naturalmente carece de una actividad de mejora continua, a diferencia de SCRUM que cuenta con una reunión de “retrospectiva” e “identificación de impedimentos” [32]. Por otra parte, en algunas propuestas como la presentada en [22] la versión de ISO 9001 no es la más reciente y los resultados presentados se han obtenido utilizando herramientas y tecnologías específicas que reducen la genericidad al replicar la solución propuesta. De esta manera, se puede decir que las MiPyMEs de desarrollo de software carecen de herramientas que les permitan tomar el camino correcto en la implementación de un sistema de gestión de la calidad que además soporte la definición de procesos ágiles de desarrollo como una ventaja competitiva en el contexto de estas organizaciones.

Teniendo en cuenta lo anterior y partiendo de la premisa que las MiPyMEs de desarrollo de software del país necesitan implementar modelos de referencia y enfoques ágiles que permitan gestionar los procesos de software, organizacionales y administrativos, es pertinente preguntarse: ¿Es posible integrar las prácticas de gestión de proyectos de software con base en el enfoque ágil SCRUM y las prácticas de gestión de la calidad definidas en el estándar ISO 9001:2015? La respuesta a esta pregunta no es simple, y aunque en la literatura consultada se evidencian algunas iniciativas relacionadas, no se ha tenido en cuenta la definición de una guía integrada, clara y detallada que soporte la gestión de proyectos basada en SCRUM y el estándar ISO 9001:2015. Tal herramienta brindaría a las MiPyMEs de desarrollo de software la posibilidad de implementar un sistema de gestión de la calidad junto con un proceso de desarrollo ágil, permitiéndoles elevar la satisfacción de sus clientes, disminuir la entropía de sus procesos y sobre todo fortalecer la industria colombiana y latinoamericana del desarrollo de software.

Por lo tanto, en este proyecto se lleva a cabo una investigación que comienza por la obtención del estado del arte con respecto a la problemática mencionada, la cual permitió identificar los conceptos y factores para tener en cuenta a la hora de integrar un enfoque ágil con un estándar de calidad. Posteriormente se lleva a cabo una armonización entre los enfoques SCRUM e ISO 9001:2015 tomando como referencia las técnicas de armonización definidas en [35]; de esta manera fue posible homogeneizar los enfoques para posteriormente llevar a cabo una comparación de elementos de proceso e identificar las relaciones entre los mismos. La armonización permitió definir un conjunto de modelos de proceso basados en las actividades, roles y artefactos de SCRUM; ajustándolos de tal forma que den soporte a los requisitos de ISO 9001:2015. Los modelos de procesos y sus descripciones están disponibles en una

guía electrónica⁵. Finalmente, la solución propuesta es evaluada a través de un estudio de caso en una pequeña empresa de desarrollo de software de la región obteniendo resultados satisfactorios en términos de esfuerzo y conformidad de los participantes con la solución aplicada.

A cotninuación, y teniendo en cuenta lo anterior se describen los objetivos general y específicos que guiaron la realización de este proyecto.

1.2. Objetivos

1.2.1. Objetivo general

Definir un proceso basado en SCRUM que permita soportar el cumplimiento de las prácticas establecidas en el capítulo 8. *Operación* del estándar ISO 9001:2015 a través de una perspectiva ágil de gestión de proyectos de software.

1.2.2. Objetivos específicos

- OE1.** Realizar una revisión sistemática de la literatura acerca de la integración de prácticas ágiles y el estándar ISO 9001, lo cual permitirá identificar las propuestas, soluciones y herramientas definidas por otros autores.
- OE2.** Identificar y definir las relaciones entre SCRUM y el capítulo 8. *Operación* del estándar ISO 9001:2015 a través de la comparación a bajo nivel de abstracción utilizando como interfaz el estándar ISO 90003:2014 y las técnicas de armonización de modelos propuestas en [35].
- OE3.** Definir un proceso integrado para la gestión de proyectos ágiles que permita a las micro, pequeñas y medianas empresas desarrollar productos y servicios de software a través de un enfoque ágil y conforme a los requisitos de la norma ISO 9001:2015.
- OE4.** Generar una guía electrónica en BPMN utilizando la herramienta Bizagi Modeler que permita conocer los elementos del *Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015* y esté disponible vía web.
- OE5.** Evaluar la propuesta llevando a cabo un estudio de caso en una empresa de software de la región.

⁵<http://artemisa.unicauca.edu.co/~miguelmorcillo/public>

1.3. Estrategia de la Investigación

Para llevar a cabo la ejecución del este proyecto, se utilizaron los métodos Investigación-Acción con múltiples ciclos de manera secuencial [36] y Estudio de Caso [37]. Teniendo en cuenta las fases y actividades propuestas por esta metodología, se llevaron a cabo 4 ciclos de investigación, a continuación se describen los ciclos y las actividades que fueron realizadas de manera secuencial e incremental para el desarrollo del proyecto.

- **Ciclo 1. Análisis conceptual:** En este ciclo se llevó a cabo la investigación del estado del arte acerca de la integración entre enfoques ágiles y el estándar ISO 9001. Se identificaron técnicas y elementos de integración, factores y positivos y negativos a tener en cuenta, entre otros aspectos. Asimismo, la realización de este ciclo permitió identificar las propuestas y soluciones existentes y los elementos sensibles a tener en cuenta para la definición de la solución.
 - Actividad 1.1: *Analizar y entender la literatura:* Se compararon los diferentes estudios acerca de la integración entre enfoques ágiles y el estándar ISO 9001, identificando aquellos estudios que incluyen como objeto de investigación el estándar ISO 9001 en cualquier versión y uno o más enfoques ágiles.
 - Actividad 1.2: *Estudio de la literatura:* Se identificaron qué elementos se deben considerar al momento de gestionar proyectos de desarrollo de software desde un enfoque ágil, siendo conforme a los requisitos del estándar ISO 9001 en el contexto de una MiPyME de desarrollo de software.
 - Actividad 1.3: *Síntesis de la literatura:* Para realizar la síntesis de la literatura se tuvo en cuenta criterios de selección para identificar estudios que contribuyen directa o indirectamente a la investigación con respecto a la integración del estándar ISO 9001 y enfoques ágiles.
- **Ciclo 2. Definir y diseñar un proceso para la gestión de proyectos ágiles de desarrollo de software conforme a los requisitos de ISO 9001:2015:** En esta etapa se definió un proceso cuyo conjunto de subprocesos están basados en las etapas del ciclo de SCRUM, incluyendo actividades, roles y artefactos, ajustándolos de tal forma que den soporte a los requisitos del capítulo 8. *Operación* de ISO 9001:2015, y tomando como referencia las directrices de aplicación en el software en ISO 90003:2014 [38], conservando además los principios ágiles.
 - Actividad 2.1: Armonización del estándar ISO 9001:2015 y SCRUM.
 - Actividad 2.2: Diseño (modelado) de procesos utilizando las actividades, roles y artefactos de la Actividad 2.1, y los elementos necesarios para dar soporte a los requisitos del capítulo 8. *Operación* de ISO 9001:2015
- **Ciclo 3. Evaluación de la propuesta:** La evaluación de la propuesta se llevó a cabo

mediante el uso del método de investigación Estudio de Caso. Tal estudio se realizó en una organización de desarrollo de software.

- Actividad 3.1 *Planificación*: Se llevó a cabo la capacitación para el personal seleccionado por la respectiva organización, coordinación, organización y diseño del estudio de caso.
- Actividad 3.2 *Observación*: Se recolectaron los datos durante la ejecución del estudio de caso en la organización.
- Actividad 3.2 *Reflexión*: Se generó un reporte como resultado del análisis de los datos obtenidos durante la ejecución del estudio de caso. Asimismo, se lleva a cabo retroalimentación y evaluación del aprendizaje obtenido.
- **Ciclo 4. Documentación y socialización**: Este ciclo se llevó a cabo de manera transversal durante todo el proyecto. Se realizaron las actividades descritas a continuación.
 - Actividad 4.1: Elaboración de la monografía y los anexos que resulten durante la realización de este proyecto.
 - Actividad 4.2: Elaboración de un artículo de investigación que describa los resultados obtenidos durante la realización y aplicación de la propuesta.
 - Actividad 4.3: Generación de una guía electrónica en la cual estarán disponibles los procesos y subprocesos resultantes de la propuesta en la notación BPMN.
 - Actividad 4.4: Presentación y sustentación de los resultados obtenidos durante el desarrollo del proyecto.

1.4. Estructura del documento

Este documento está compuesto de 6 capítulos distribuidos de la siguiente manera: (I) **Capítulo I. Introducción**, presenta la problemática que motivó a la realización de este proyecto y los objetivos del mismo. (II) **Capítulo II. Marco teórico y estado del arte**, presenta los conceptos clave que son abordados en este proyecto, además del estado del arte obtenido a través de la realización y el análisis de una revisión sistemática de la literatura. (III) **Capítulo III. Armonización**, donde se lleva a cabo la homogeneización e integración de los enfoques abordados en este proyecto, SCRUM e ISO 9001:2015. (IV) **Capítulo IV. Proceso Integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015**, donde se presenta el conjunto de modelos de proceso y sus descripciones que conforman la solución propuesta. (V) **Capítulo V. Estudio de caso**, donde se lleva a cabo la evaluación de la propuesta en una PyME de la región. (VI) **Capítulo VI. Conclusiones y trabajos futuros**, se presentan las conclusiones obtenidas posterior a la realización de este

proyecto y las vías de trabajos futuros. Y (VII) **Bibliografía**, donde se muestra la lista de referencias citadas en este documento.

Capítulo II. Marco teórico y estado del arte

En este capítulo se introducen los conceptos relacionados al problema que se pretende resolver y su contexto, partiendo de un recuento de la realidad de las MiPyMes⁶ desarrolladoras de software en Colombia, a una descripción general de los enfoques estudiados en este proyecto: ISO:9001:2015 y SCRUM. Asimismo, se presenta una revisión sistemática de la literatura con la cual se buscó conocer las propuestas, iniciativas y trabajos relacionados respecto a la integración del estándar ISO 9001 y metodologías ágiles desde principios de siglo, no solo con el objetivo de conocer el estado de la literatura, sino también de identificar nuevas líneas de investigación que puedan ser abordadas como proyectos futuros.

2.1. Marco Teórico

2.1.1. Actualidad de la industria del desarrollo de software

La industria del desarrollo de software en Colombia ha sido una de las grandes apuestas de los últimos gobiernos, desde hace una década es posible observar que se han venido implementando planes y estrategias para fortalecer el sector de TI⁷ a lo largo de todo el territorio nacional, por nombrar algunos: (i) El Plan Vive Digital⁸ y (ii) Gobierno en Línea⁹, que han llevado a cabo una serie de convocatorias para fortalecer el capital humano en conocimiento y desarrollo de competencias blandas en el contexto de la industria de desarrollo de software y TI [39]; permitiendo a profesionales y estudiantes del sector TI acceder a cursos y certificaciones en Ingeniería del Software, Big Data, Pruebas de Software, Cloud Computing, entre otros; de los cursos y certificaciones en Ingeniería del software han estado: PMP¹⁰ SCRUM Product Owner, Certified SCRUM Master (CSM) y SCRUM Developer [39]. En este sentido, es posible observar que las iniciativas no solo han sido destinadas a la producción de bienes y servicios en tecnología si no también a la familiarización y capacitación respecto al correcto uso de la misma en todo el país [40]. El sector de teleinformática, software y TI por su capacidad de innovar, desarrollar, y transferir tecnología a todos los sectores económicos del país y por su potencial de crecimiento y generación de empleo es considerado como uno de los de mayor proyección en Colombia [40]. Algunos de los programas que han fortalecido la industria, infraestructura y cobertura de las tecnologías de la información son: Programa de Transformación Productiva (PTP¹¹) y

⁶ Micro, Pequeñas y Medianas empresas.

⁷ Tecnologías de la información, Information Technology.

⁸ El Plan Vive Digital, disponible en: <https://goo.gl/oVTFYZ>

⁹ Estrategia Gobierno en Línea, disponible en: <https://goo.gl/fVWsHk>

¹⁰ Project Management Professional (PMP).

¹¹ Programa de Transformación Productiva, disponible en: <http://goo.gl/ePJ3GQ>

el programa de Fortalecimiento de la industria TI (FITI¹²).

En 2014 el Ministerio de las tecnologías de la información y las telecomunicaciones (MinTIC) llevó a cabo el *Censo del Directorio de Empresas Activas de la Industria del Software y Servicios Asociados con TI* en el que fueron tenidas en cuenta 4016 empresas de las cuales se puede decir según el valor de sus ventas, que el 40% tenía ingresos menores a 294 millones de pesos, el 19% tenía entre 294 y 3000 millones de pesos, el 4% entre 3001 y 17000 millones de pesos y el 3% de las empresas tenía ventas mayores a 17000 millones de pesos, por otra parte un 24% de las empresas no respondió y el 10% restante corresponde a empresas que fueron creadas en 2014 y por lo tanto aún no tenían información importante de sus ventas. Teniendo en cuenta lo anterior, es posible concluir que con relación al valor de las ventas de las empresas encuestadas, el mayor porcentaje en la industria son pequeñas empresas y que el porcentaje de empresas con activos elevados o grandes empresas es apenas del 5% de la muestra [40]. La preponderancia del sector es de empresas enfocadas a prestar servicios o realizar productos individualizados, en contraposición a las grandes empresas, las cuales se caracterizan por la estandarización de sus productos y servicios [40].

Por otra parte, la industria del desarrollo de software se caracteriza por un constante dinamismo debido al ritmo acelerado de innovación y avance de la tecnología en el mundo, según estudios realizados por MINTIC desde el año 2010 el sector ha presentado un constante crecimiento que se ha visto reflejado en los indicadores económicos y PIB¹³ del país, de allí la necesidad de fortalecerlo y sobre todo, de brindarle las herramientas necesarias para que las diferentes industrias del sector puedan estar a la vanguardia en cuanto a servicios se refiere y con las capacidades necesarias para competir tanto en el mercado local como Internacional [40].

La gran mayoría de las empresas del sector en Colombia presentan estructuras de tipo vertical enfocada a áreas en las que se jerarquizan roles y responsabilidades, entre las áreas más comunes se destacan el área operativa, el área comercial y el área administrativa, aún cuando las empresas son de menor tamaño este tipo de estructura organizacional se mantiene, sin embargo tienden a asignar varios roles y responsabilidades a una sola persona lo que complica el proceso de toma de decisiones al entrelazar diferentes áreas, llevando a procesos caóticos y resultados inesperados [40].

Teniendo en cuenta el potencial de las TICs en Colombia, el tamaño de las empresas desarrolladoras de software y la feroz competencia y dinamismo del mercado como consecuencia del avance de la tecnología, es necesario proveer a las MiPyMEs desarrolladoras de software con estructuras organizacionales y prácticas de trabajo que sean más apropiadas respecto al tamaño de la organización, como lo son las prácticas ágiles que

¹² Programa de Fortalecimiento de la Industria TI, disponible en: <http://goo.gl/MkXf6H>

¹³ Producto interno bruto.

promueven mayor comunicación, colaboración e interacción entre individuos, permitiendo además llevar a cabo una definición e institucionalización de procesos que puedan ser controlados, y por lo tanto mejorados continuamente en el marco de un sistema de gestión de la calidad conforme a ISO 9001, esto con el objetivo de fortalecer la industria no sólo en Colombia si no en toda Latinoamérica, permitiendo a las MiPyMEs acceder a un mercado de alta competitividad, innovación y calidad.

De acuerdo con [41], el promedio de vida de las empresas PyMEs¹⁴ en el país para 2014 era de 12 años debido a su incapacidad de autosostenimiento, ya que no contaban con la totalidad de los requisitos que solicitan las entidades bancarias para adquirir solvencia y solidificar su funcionamiento; pero sobre todo por el déficit de recursos humanos altamente capacitados. Teniendo en cuenta el elevado número de empresas PyMEs que conforman el sector y el promedio de vida de las mismas, es necesario que se desarrollen diversas estrategias para aumentar su promedio de vida, puesto que, según [41] estas organizaciones aportan el 60 % del PIB y el 64 % del empleo en Colombia; por lo tanto para el fortalecimiento del sector de las TIC se debe contemplar como un elemento estratégico la necesidad de desarrollar mejores prácticas, capacitaciones, procesos de gestión de conocimiento, oportunidades y capacitación en talento humano para que dichas empresas logren adaptarse y responder a las necesidades del mercado [40].

Respecto a los modelos de calidad, el 94 % de los empresarios encuestados determinaron que la adopción de modelos o estándares de calidad es relevante, se indagó respecto a los modelos de calidad en TI implementados o en proceso de implementación y se encontró que el 44 % aún no tiene ejecutado un modelo de calidad, el 20 % se encuentra en desarrollo y el 36 % ya tiene implementado un modelo o estándar de calidad; entre los más implementados se encuentran: CMMI nivel3, ITMARK e ISO 9001 [40]. De lo anterior, es evidente una vez más la necesidad de fortalecer el sector TI en Colombia impulsando a las empresas para que adopten modelos o estándares de calidad y procesos bien definidos en el marco de la mejora continua.

2.1.2. Enfoques ágiles

Las metodologías o enfoques ágiles, generalmente describen un conjunto de principios y valores para el desarrollo de software establecidos en el Manifiesto ágil en los cuales los requerimientos y el producto software como tal evolucionan incremental e iterativamente a partir de un esfuerzo colaborativo entre uno o varios equipos autoorganizados y por lo general multidisciplinarios, los enfoques ágiles proveen planeación adaptativa, desarrollo evolutivo, entregas tempranas de valor, mejora continua y flexibilidad y respuesta al cambio [42], además de esto los enfoques ágiles en general vinculan fuertemente al cliente en la mayor parte del ciclo de vida del producto software.

¹⁴ Pequeñas y Medianas Empresas.

Aunque actualmente los enfoques ágiles son implementados en todo tipo de organizaciones [43]. Son ampliamente utilizados en MiPyMEs alrededor del mundo debido a la facilidad de implementación con respecto a complejidad y costos si se comparan con estándares y enfoques tradicionales [44]. De acuerdo al undécimo reporte anual del estado de los enfoques ágiles¹⁵ el 32% de las organizaciones de la industria del desarrollo de software tiene menos de 100 empleados, el 34% tiene entre 101 y 1000 empleados, el 15% tiene entre 1001 y 5000 empleados y finalmente un 19% de las organizaciones tiene más de 5001 empleados [43]. Lo anterior deja en evidencia que de las organizaciones en la industria del desarrollo de software que implementan enfoques ágiles alrededor del mundo, un 66% son MiPyMes. Además, respecto a los continentes donde están ubicadas estas organizaciones el 88% está conformado por Norteamérica, Europa y Asia y tan sólo un 5% proviene de Latinoamérica, lo cual demuestra la pobre implementación de enfoques ágiles en nuestra región.

Por otra parte, las organizaciones de desarrollo de software alrededor del mundo coinciden en que las razones más relevantes para adoptar enfoques ágiles son las entregas tempranas del producto y la habilidad para enfrentar requisitos y prioridades cambiantes, y que los principales beneficios obtenidos son consecuentemente las entregas tempranas de valor y la visibilidad del proyecto a nivel de equipo, incluso en equipos ágiles distribuidos [43]. Sin embargo a pesar del éxito de la implementación de los enfoques ágiles, el 80% las organizaciones de desarrollo de software se consideran como “*aún madurando*” [43], dejando una gran oportunidad para la mejora de procesos tendiendo a una convergencia entre enfoques tradicionales de disciplina y enfoques ágiles.

2.1.3. SCRUM

SCRUM [12] es uno de los enfoques ágiles más populares en todo el mundo [34], [32]. Fue presentado oficialmente por Ken Schwaber y Jeff Sutherland en 1995 [45], es un framework¹⁶ o marco de trabajo ágil para llevar a cabo soluciones a problemas complejos de manera productiva y creativa generando productos del mayor valor posible [46], está basado en un conjunto de valores, principios y prácticas que proveen los fundamentos para que cada organización le agregue su implementación única [47].

Los roles del proceso están definidos de la siguiente manera: Un *Product Owner* es el representante del cliente ante el equipo de desarrollo o *Development Team*, encargado de comunicar los deseos y necesidades; Un *SCRUM Master* es el experto en SCRUM, encargado de verificar que el framework se está aplicando correctamente, además de ser un solucionador de problemas, comunicador y mediador; finalmente el equipo de desarrollo

¹⁵ 11th Annual State of Agile Report, disponible en: <http://goo.gl/B2To3U>

¹⁶ Un marco de trabajo es un conjunto de componentes que conforman una base sobre la cual se construyen proyectos, generalmente están definidos teniendo en cuenta las mejores prácticas y directrices respecto a la temática de los proyectos que sobre él se construyen.

puede incluir arquitectos, analistas, diseñadores, desarrolladores, testers, entre otros roles relacionados a la implementación del producto software como tal [46] [22].

SCRUM consiste en un proceso iterativo e incremental que divide el proyecto en fases o iteraciones denominadas *Sprints*, los *Sprints* generan entregas de valor que incrementalmente conforman el producto final [34]. Un *Sprint* es llevado a cabo dentro de un lapso de tiempo que generalmente es fijado entre 2 y 4 semanas. La gestión del proceso incluye una actualización continua del estado de los requisitos del producto o *Product Backlog*, los requisitos del producto son generalmente escritos en forma de *Historias de Usuario* que facilitan el entendimiento de los mismos al cliente y sus interesados, sin embargo cada organización puede utilizar el artefacto que considere adecuado para representar los requisitos del producto software. Para cada *sprint*, todo el equipo (*SCRUM Team*) se reúne para seleccionar un conjunto de ítems del Product Backlog que serán implementados en el sprint, conformando así el alcance del sprint, el equipo puede desglosar los ítems en tareas que contienen información más técnica y detallada, posteriormente los ítems y tareas identificadas son auto-asignadas por el equipo de desarrollo, a esta reunión se le llama planeación del sprint [22].

Todos los días durante un sprint el equipo asiste a una corta reunión diaria, típicamente de pie para agilizar la comunicación, los miembros del equipo intercambian información sobre lo que han realizado, lo que esperan realizar y los inconvenientes que han encontrado. Las reuniones diarias no son para reportar el progreso a los superiores, si no para mantener a todo el equipo actualizado acerca de las actividades y los posibles impedimentos que deben ser resueltos, al final del sprint se espera tener un incremento del producto potencialmente entregable [22].

Cuando un sprint termina, se lleva a cabo la reunión de revisión del sprint donde el equipo hace una demostración de las nuevas características implementadas, los errores corregidos u otros artefactos relacionados al producto. Las presentaciones, diapositivas e informes de progreso no son considerados de valor, el énfasis es en la demostración del software funcionando y el valor de negocio para el cliente. Posterior a cada sprint y su revisión, el equipo se reúne sin el cliente para la reunión de retrospectiva del sprint en la que se analizan las cosas que se hicieron bien y las que se pueden mejorar [22].

2.1.3.1. ¿Por qué SCRUM y no XP?

Aunque existen versiones híbridas que incluyen prácticas de SCRUM y XP [32] [24] [4] [43], existen algunas diferencias que hacen a SCRUM el framework más popular a nivel mundial en la actualidad con un 58% de uso según el 11° reporte anual del estado de los enfoques ágiles, muy por encima del segundo enfoque que es un híbrido entre SCRUM y XP con un 10% de uso [43]. XP no es claro en la comunicación con la alta dirección o el cliente, SCRUM define un Product Owner quien es la única interfaz entre el cliente y

el equipo de desarrollo, de esta manera si hay temas contractuales a ajustar el equipo de desarrollo está aislado de ruido y pueden dedicarse a implementar las historias de usuario del Product Backlog provistas por el Product Owner. SCRUM se enfoca más en aspectos de organización y dirección del proyecto mientras que XP se enfoca más en prácticas de Ingeniería, sin embargo SCRUM deja un espacio para definir las prácticas que sean necesarias en cada Planeación del sprint, por ejemplo temas de arquitectura del proyecto. La reunión de retrospectiva del sprint garantiza una reflexión sobre el trabajo realizado y permite mejorar el proceso continuamente, además la reunión diaria de SCRUM garantiza la transparencia y visibilidad del proyecto, y la transferencia continua del conocimiento [32].

2.1.4. ISO 9001

Un sistema de gestión de la calidad puede ser definido en su forma más simple como las políticas y los procedimientos de una organización [24]. La adopción de un sistema de gestión de la calidad es una decisión estratégica que permite a la organización mejorar el desempeño global y conformar una base sólida para llevar a cabo iniciativas de desarrollo sostenible, permitiendo monitorear, medir y mejorar continuamente los procesos internos [30], proporcionando regularmente productos y servicios que satisfacen las necesidades de los clientes [3] [22].

ISO 9000 es una serie de estándares internacionales de propósito general concernientes al establecimiento y mantenimiento de un sistema de gestión de la calidad comprendiendo vocabulario, requerimientos y recomendaciones para la mejora continua, los anteriores pueden ser aplicados desde grandes compañías a muy pequeñas organizaciones [23]. Fue creado en el Reino Unido a finales de los años 90, sin embargo en la actualidad cuenta con más de un millón de organizaciones certificadas alrededor del mundo y tiene cerca de 30 años de experiencia en sistemas de gestión de la calidad, naturalmente la norma ha sido mejorada y actualizada conforme el tiempo y las necesidades de las organizaciones cambian, siendo su actualización más reciente la versión 2015 [48].

El principal objetivo es conseguir la satisfacción del cliente y sus interesados a través de productos consistentes [22], además permite a las organizaciones abordar riesgos y oportunidades asociadas al contexto. Algunas de las características de un sistema de gestión de la calidad conforme a ISO 9001 son: a) el enfoque a procesos incorporando el ciclo PHVA¹⁷, el cual permite a la organización planificar, monitorear y mejorar los procesos internos asegurándose que cuentan con los recursos necesarios; b) el pensamiento basado en riesgos permite a una organización determinar los factores que podrían causar que el sistema de gestión de la calidad se desvíe de los resultados planificados, y así poner en marcha controles preventivos para minimizar los efectos negativos; c) el sistema de gestión de la calidad conforme a ISO 9001 se basa en los principios de la gestión de la calidad: enfoque al cliente, liderazgo, compromiso de las personas, enfoque a procesos, mejora continua, toma de decisiones basadas en evidencia y gestión de las relaciones [3].

Para las organizaciones desarrolladoras de software, ISO 9001 es una gran elección para un sistema de gestión de la calidad, ya que tiene una naturaleza genérica que aplicada a la industria del software permite alcanzar las metas de calidad en los requerimientos del producto durante todas etapas del ciclo de vida incluyendo análisis y diseño, desarrollo, despliegue, y mantenimiento [10]. Además, el alcance de ISO 9001 va más allá del proceso de desarrollo de software abarcando todas las áreas de una organización incluyendo procesos administrativos y comerciales además de operativos, esto lo hace una mejor opción con respecto a otros modelos de calidad como CMM por ejemplo que se enfoca únicamente

¹⁷ Ciclo de Deming: planear, hacer, verificar y actuar.

en los procesos software, ISO 9001 gira en torno a la satisfacción del cliente, mientras que CMM asume que al seguir estrictamente los requisitos de áreas clave, la satisfacción del cliente se verá incrementada [32].

Otro factor que hace a ISO 9001 una gran elección para una organización es que al ser un estándar bien conocido y aceptado alrededor del mundo, permite acceder a nuevos nichos de mercado y atraer potenciales clientes como organizaciones gubernamentales o entidades privadas que requieren un estándar de calidad para los proveedores [22] [28].

En la industria del software existe un concepto erróneo generalizado acerca de que la burocracia y evidencia requerida por ISO 9001 va en contra de las prácticas ágiles de desarrollo de software [30], [32], [23], [22], [24], [28]. Sin embargo, ISO 9001 describe qué se debe hacer para demostrar conformidad más no cómo se debe hacer, dejando esto una gran variedad de implementaciones, particularmente en ISO 9001:2015 los principios de calidad mencionados previamente están muy alineados a las prácticas ágiles de SCRUM. Además, en adición a ISO 9001, el documento ISO 90003 es de gran importancia dado que en él se definen un conjunto de directrices para aplicar los requerimientos del sistema de gestión de la calidad a la industria del desarrollo de software [28] [38].

2.2. Estado del Arte

La ejecución de este proyecto se llevó a cabo tomando como base los siguientes principios: (i) realización de una revisión sistemática de la literatura para conocer el estado del arte con respecto a la integración del estándar ISO 9001 y enfoques ágiles, (ii) la categorización de las propuestas seleccionadas como primarias, identificando: país de origen; modelo, enfoque, estándar y/o metodología analizada; técnica de armonización utilizada; éxito del estudio de caso y tamaño de la organización, (ii) análisis de las propuestas seleccionadas como primarias para la posterior definición de un proceso basado en SCRUM que permita la gestión de proyectos de desarrollo de software conforme al estándar ISO 9001:2015 en MiPyMEs, tomando como referencia el conocimiento y las mejores prácticas encontradas en los estudios primarios seleccionados y utilizando la notación de modelado de procesos BPMN¹⁸ 2.0.

2.3. Revisión sistemática de la literatura

La revisión sistemática es un proceso formal que permite extraer información de la literatura de una manera ordenada y con un método bien definido, reduciendo así la posibilidad de

¹⁸ Notación y Modelado de Procesos de Negocio (Business Process Model and Notation), disponible en: <http://goo.gl/Bdcxg7>

sesgo en la obtención de la información, y con una presentación de la misma de fácil lectura e identificación de tendencias o líneas futuras de investigación. Para este proyecto se siguieron las directrices definidas en [49] para llevar a cabo revisiones sistemáticas de la literatura. A continuación, se presenta un resumen de la revisión sistemática que se llevó a cabo. El documento completo y sus anexos se pueden ver en el **Anexo D**.

2.3.1. Foco de la pregunta

Con la revisión sistemática se busca identificar iniciativas y estudios relacionados en los cuales se definan o propongan métodos, marcos, estrategias y/o soluciones para llevar a cabo la unión, integración, y/o armonización de ISO 9001 y metodologías ágiles de desarrollo de software.

2.3.2. Pregunta de investigación

¿Qué trabajos e iniciativas relacionadas con la implementación de ISO 9001 y metodologías ágiles se han llevado a cabo?

2.3.3. Planeación de la revisión sistemática de la literatura

La revisión sistemática se llevó a cabo con los siguientes parámetros:

(I) la definición del criterio de selección de fuentes fue basado en: (a) la relación de la fuente con el tema de la revisión, (b) la disponibilidad de acuerdo al perfil académico del investigador, (c) la accesibilidad a través de internet y (d) el juicio de expertos en calidad de software y metodologías ágiles;

(II) el lenguaje en el cual se llevará a cabo la búsqueda fue inglés, el lenguaje en el que se realizó el informe fue español;

(III) El proceso de búsqueda y selección de estudios se llevó a cabo a través de un proceso iterativo incremental en el que para cada fuente de búsqueda se aplicaron criterios de inclusión y posteriormente se aplicaron criterios de exclusión a todos los estudios obtenidos.

En la Tabla 1 se muestran las palabras clave y sinónimos utilizados para resolver la pregunta de investigación.

Ítem	Palabra clave	Sinónimo
1	ISO	IEC
2	agile	SCRUM, XP
3	standard	model, framework, technology, method, methodology
4	quality	QA
5	assurance	guarantee, insurance
6	software	product, service, system
7	development	construction, execution
8	implementation	application, utilization, usage, integration, union, harmonization

Tabla 1. Palabras clave relacionadas con la pregunta de investigación.

Combinando los conectores lógicos “AND” y “OR” con las palabras claves identificadas, se realizó una primera búsqueda con el prototipo inicial de cadena de búsqueda sobre una fuente. Obteniendo así las cadenas de búsqueda que se muestran en la Tabla 2.

N°	Cadena de búsqueda básica
1	(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology) AND (quality OR guarantee OR insurance OR assurance OR software OR product OR service OR system OR development OR execution OR construction OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization)
2	(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction) AND (quality OR guarantee OR insurance OR assurance OR software OR product OR service OR system OR development)

3	(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction)
---	---

Tabla 2. Cadenas de búsqueda.

Debido al estado del arte del problema identificado para esta revisión, existen pocos estudios relevantes respecto a la implementación de ISO 9001 y metodologías ágiles, las tendencias muestran que hay variedad de estudios que tratan estándares ISO como ISO/IEC 12207, ISO/IEC 15504 y algunas metodologías ágiles, o referente a la calidad hay variedad de estudios que tratan modelos como CMM, CMMI y metodologías ágiles. Por esa razón se ha decidido utilizar la cadena de búsqueda número 3, que es más simple respecto a 1 y 2, y utiliza menos operadores lógicos AND, permitiendo así la inclusión de estudios que podrían haber sido dejados por fuera con las cadenas 1 y 2.

La cadena de búsqueda seleccionada sin embargo, fue adaptada a cada motor de búsqueda de las fuentes de datos seleccionadas, las adaptaciones realizadas pueden ser encontradas en la revisión sistemática anexa.

2.3.4. Resultados de la revisión sistemática de la literatura

En esta sección, se presentan los hallazgos de la revisión sistemática realizada con el fin de analizar el estado del arte con respecto a la integración del estándar ISO 9001 y enfoques ágiles. Para ello, se definió un formato de extracción de información validado por el director y codirector de este proyecto, que permitió analizar en detalle el contenido relevante para la base del conocimiento de esta investigación. A continuación, en la Tabla 3 se presentan los estudios primarios seleccionados.

N°	Nombre del estudio	Enfoques analizados	Aplicación
1	Achieving ISO 9001 Certification for an XP Company	ISO 9001:2000, XP	Estudio de caso
2	A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process	ISO 29110, Scrum, XP, Upedu	No se evidencia

3	Applying ISO/IEC 12207:2008 with Scrum and Agile Methods	ISO/IEC 12207:2008, Scrum	Análisis de 25 empresas
4	Assessing XP at an European Company	XP	Estudio de caso
5	Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods	No se evidencia	No se evidencia
6	Certifying for CMM Level 2 and ISO 9001 with XP@Scrum	CMM Level 2, ISO 9001:2000, XP@Scrum	Estudio de caso
7	Combining Extreme Programming with ISO 9000	ISO 9001:2000, XP	Estudio de caso
8	Getting ISO 9001 Certified for Software Development Using Scrum and Open Source Tools	ISO 9001:2008, Scrum	Estudio de caso
9	How can Agile and Documentation-Driven Methods be Meshed in Practice	GAMP, FDA, Scrum	Estudios de caso
10	Investigation of The Capability of XP to Support The Requirements of ISO 9001 Software Process Certification	ISO 9001:2008, ISO 90003, ISO 12207, XP	No se evidencia
11	ISO 9001 And Agile Development	ISO 9001:2000, SCRUM/XP	Estudio de caso
12	Reconciling The Irreconcilable A Software Development Approach That Combines Agile with Formal	ISO 9001/ISO 12207/SPICE/ACSS/NASA-SEL, Scrum/XP/ASD	Estudio de caso
13	Requirements of An ISO Compliant XP Tool	ISO 9001:2000, XP	No se evidencia
14	Standards And Agile Software Development	ISO 12207:1995, XP	No se evidencia
15	Structure of A MultiModel Catalog For a Software Projects Management Including Agile And Traditional Practices	PMI/CMMI/ISO 9001/TSP, Scrum/Kanban/XP	No se evidencia
16	The Agile and The Disciplined Software Approaches: Combinable or just Compatible	CMMI/ISO 9001, Scrum/XP	No se evidencia

17	The Application of ISO 9001 to Agile Software Development	ISO 9001, Scrum, XP	No se evidencia
18	The Effect of Moving From A Plan Driven To An Incremental Software Development Approach with Agile Practices	ISO 9001:2000, Scrum, XP	Estudio de caso

Tabla 3. Estudios primarios seleccionados.

El metaanálisis de esta revisión sistemática tuvo como objetivo determinar, entre otros:

- publicaciones por año,
- publicaciones por país,
- porcentaje de estándares/modelos/enfoques/metodologías analizados,
- tipos de empresas,
- factores de éxito y fracaso en la implementación de ISO 9001 y metodologías ágiles y
- técnicas de armonización utilizadas,

La Figura 2 muestra la cantidad de estudios primarios seleccionados por año de publicación desde 2002 hasta 2016.

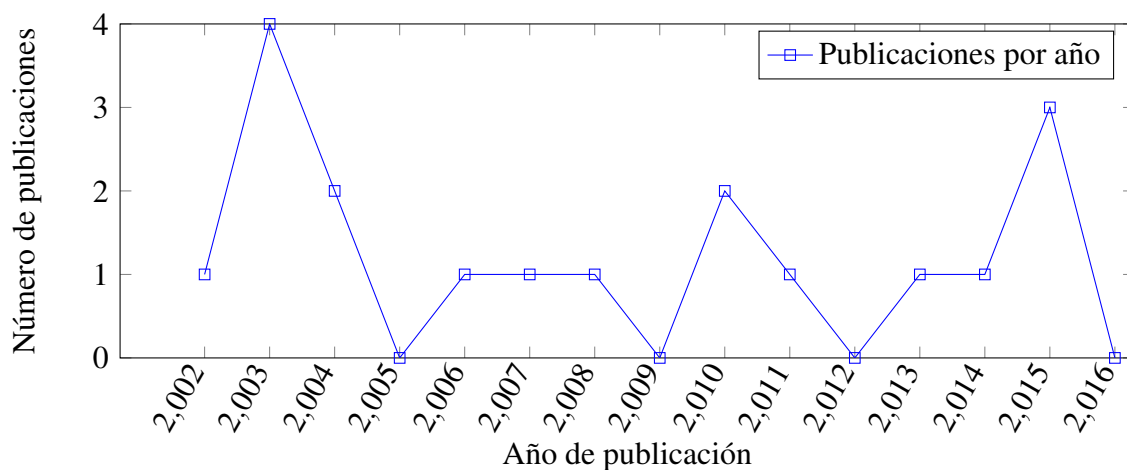


Figura 2. Publicaciones por año.

A continuación, la Figura 3 muestra la cantidad de estudios primarios seleccionados por país.

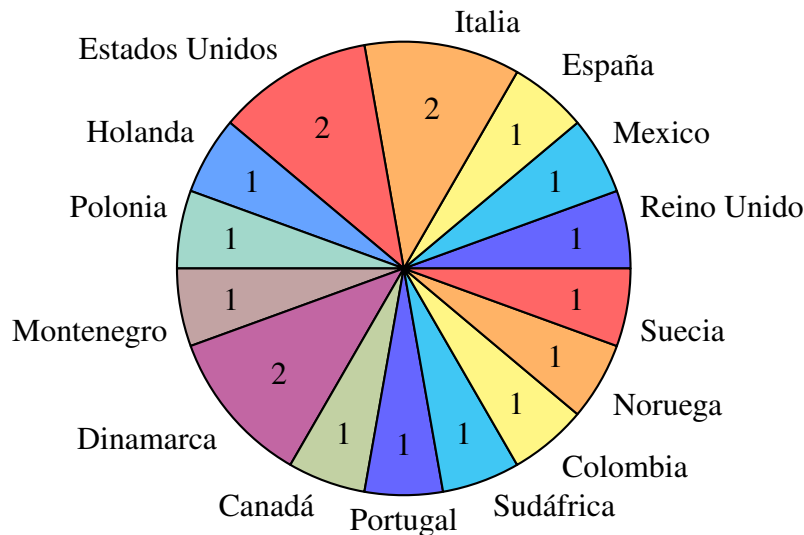


Figura 3. Estudios por país.

La Tabla 4 presenta el porcentaje de modelos, enfoques, estándares y/o metodologías analizadas en todos los estudios primarios seleccionados, de esta tabla es posible observar que el 35.5% de los enfoques analizados fueron estándares de ISO (ISO 90003, ISO 15504, ISO 29110, ISO 12207 e ISO 9001); asimismo, el 7.2% de los enfoques analizados fueron modelos que son aceptados y ampliamente utilizados en la industria del desarrollo de software (CMM¹⁹, CMMI²⁰, PMI y TSP), otro 7.2% de los enfoques fueron estándares referentes a sectores específicos como el aeroespacial (ECSS, NASA-SEL), la manufactura (GAMP) y el estándar de la organización FDA (Food & Drug Administration), finalmente el 50.9% de los enfoques analizados fueron los enfoques ágiles (Upedu, ASD, Kanban, SCRUM y XP).

¹⁹ Por sus siglas en inglés: Capability Maturity Model, disponible en <http://goo.gl/aCoUuS>

²⁰ Por sus siglas en inglés: Capability Maturity Model Integration, disponible en: <http://goo.gl/K6zY7f>

Modelo, enfoque, estándar o metodología analizada	N° de estudios en los cuales se analiza	%
XP	14	78 %
ISO 9001	12	67 %
SCRUM	11	61 %
ISO/IEC 12207	4	22 %
ISO 90003	1	5 %
ISO 29110	1	5 %
SPICE	1	5 %
CMM	1	5 %
CMMI	1	5 %
Kanban	1	5 %
PMBOK	1	5 %
ECSS	1	5 %
NASA-SEL	1	5 %
TSP	1	5 %
Upedu	1	5 %
ASD	1	5 %
GAMP	1	5 %
FDA	1	5 %

Tabla 4. Porcentaje de Modelos, enfoques, estándares y/o metodologías analizadas.

En la Tabla 5 se presenta la información relacionada con los tipos de empresas en las cuales fueron llevados a cabo los estudios de caso, el tipo de empresa ha sido definido de acuerdo a cómo los autores se refieren a las mismas en cada estudio, fue posible identificar 3 tipos de empresa con respecto al tamaño:

1. **Gran Empresa**, el 50 % de los estudios de caso fue realizado en grandes empresas (5 estudios), caracterizadas por tener una proyección global cuyos clientes requerían muy altos niveles de seguridad y fiabilidad en sectores aeroespacial e industrial entre otros.
2. **Mediana Empresa**, el 30 % de los estudios de caso fue realizado en medianas empresas (3 estudios), este tipo de empresa es similar al anterior respecto a la proyección global y requerimientos críticos para sectores específicos, sin embargo en menor medida por el volumen de clientes y el tiempo de vida de la organización.
3. **Pequeña Empresa**, el 20 % de los estudios de caso fue realizado en pequeñas empresas (2 estudios), a diferencia de los 2 tipos anteriores; se caracterizan por desempeñarse en mercados nacionales o regionales y por satisfacer necesidades locales de sectores específicos.

%	Nº de estudios	Tipo de empresa
50 %	5	Gran Empresa
30 %	3	Mediana Empresa
20 %	2	Pequeña Empresa

Tabla 5. Tipos de empresa en los estudios de caso.

Las técnicas de armonización utilizadas en todos los estudios primarios seleccionados se puede ver en la Tabla 6.

Técnica de armonización utilizada	Nº de estudios que la utilizaron	%
Tailoring	5	38.46 %
Mapping	3	23.09 %
Meshing	2	15.38 %
XPMM	1	7.69 %
TickIT	1	7.69 %
Action-Research	1	7.69 %

Tabla 6. Técnicas de Armonización utilizadas.

La Tabla 7 muestra factores de éxito y fracaso para la implementación de la norma ISO 9001 y metodologías ágiles, los autores expresaron algunas situaciones y características a tener en cuenta. Es importante resaltar que los factores identificados no están ordenados de acuerdo a un nivel de importancia, ni de identificación, ni de prioridad. Los factores de éxito se muestran siguiendo un patrón compuesto de: número de factor, enunciado (donde se titula el factor identificado), descripción donde se extiende el concepto para dar mayor claridad en el entendimiento del factor y la respectiva referencia.

Factor 1	
Enunciado	Los procesos, sus actividades y artefactos definidos para la organización deben existir porque contribuyen a la mejora de la calidad y no únicamente para cumplir con el estándar.
Descripción	Hace referencia a que los procesos definidos realmente tengan un valor para la organización, esto debido a que las actividades y artefactos que son definidos únicamente para cumplir con el estándar son nocivos a mediano y largo plazo, el personal tiende a tomar una postura de desagrado y aburrimiento al cumplir con actividades que realmente no aportan a los objetivos de la organización. Siempre se debe dejar espacio para la personalización del proceso y encontrar una forma adecuada de cumplir con el estándar sin agregar actividades sin razón.

Referencia	[30], [32]
Factor 2	
Enunciado	El uso de herramientas informáticas, tanto web, desktop o móviles permiten gestionar mejor, tanto a los procesos como a las personas, además que ayudan a mantener la trazabilidad y los registros de una forma más fácil sin olvidar que ahorran papel.
Descripción	Las herramientas software pueden ayudar a gestionar el proyecto, los procesos, las actividades, el personal, base de conocimiento de soporte o errores conocidos, y pueden reemplazar a la evidencia en papel escrita reduciendo tiempos.
Referencia	[30], [22], [25]
Factor 3	
Enunciado	Un consultor externo es necesario, tener la opinión de un tercero experto permite saber si las cosas se están haciendo bien desde la perspectiva de ISO 9001.
Descripción	No sólo ayuda a tener una opinión imparcial sobre cómo se lleva a cabo el proceso sino también que encargar a un experto de la implementación del proceso permite que el equipo de desarrollo no se vea afectado en labores de modelado y documentación.
Referencia	[30], [32]
Factor 4	
Enunciado	Involucrar las pruebas en todas las partes del ciclo de vida del desarrollo de software donde sea posible, el objetivo es la calidad y asegurarse que el producto correcto es aprobado y el incorrecto es rechazado es fundamental para el estándar ISO 9001.
Descripción	Realizar actividades de pruebas de software no sólo mejora la calidad del producto sino que reduce los costos de soporte y re trabajo, además que ayuda a satisfacer muchos de los requisitos de ISO 9001.
Referencia	[32], [22], [4]
Factor 5	
Enunciado	El trabajo en equipo es esencial, mantener un ambiente de trabajo positivo con personal motivado conduce a una mayor sinergia en la organización.
Descripción	En las prácticas ágiles la transferencia de conocimiento se hace in situ, cara a cara y con comunicación directa, permitiendo así una reducción en los tiempos de aprendizaje y mejorando el entorno de trabajo, esto permite que la sinergia crezca y el equipo funciona de manera más eficiente.
Referencia	[4]

Factor 6	
Enunciado	Utilizar métricas en los diferentes procesos es necesario para identificar oportunidades de mejora y la toma de decisiones al respecto.
Descripción	Lo que no se mide no se controla, para entender el funcionamiento y mejorar los procesos es necesario medirlos.
Referencia	[32], [23]
Factor 7	
Enunciado	La documentación debe ser reducida en orden de mantener la agilidad, pero no puede ser suprimida.
Descripción	Existen varias alternativas para mejorar la gestión de la documentación y conservar las evidencias necesarias para una certificación, por ejemplo el uso de herramientas informáticas o asignar personal específico para esa tarea.
Referencia	[32], [25], [27], [4]
Factor 8	
Enunciado	El talento humano, el éxito en la implementación de un estándar como ISO 9001 y una metodología ágil se logra con una fuerte disciplina y el mejor equipo.
Descripción	
Referencia	[32]
Factor 9	
Enunciado	La falta de disciplina es el principal factor que lleva al fracaso, la agilidad no significa anarquía o indisciplina en ninguna manera.
Descripción	
Referencia	[4], [29]
Factor 10	
Enunciado	La falta de compromiso de todos los involucrados, incluyendo la dirección, el personal de la organización, los clientes y los interesados es una potencial causa de fracaso.
Descripción	Todos los involucrados deben estar comprometidos para tener éxito en la implementación de una metodología ágil, más con ISO 9001. Desde clientes, proveedores e interesados, hasta la alta dirección, todos deben ser conscientes de su labor y llevarla a cabo de la mejor manera.
Referencia	[32], [25], [4]
Factor 11	
Enunciado	La falta de documentación, es una causa de fracaso para la implementación de un sistema de gestión de la calidad conforme a ISO 9001 o estándares similares.

Descripción	Si bien se busca reducir la documentación para mantener la agilidad, la documentación es necesaria y debe ser generada y conservada en la medida necesaria.
Referencia	[29], [4]

Tabla 7. Factores de éxito y fracaso.

Capítulo III. Armonización

En primera instancia, ISO 9001 al ser un estándar de calidad que requiere evidencia constante de los procesos de una organización de desarrollo de software, es aparentemente contraria a SCRUM, la cual se enfoca en la funcionalidad más que la documentación. Se debe tener en cuenta que ISO 9001 define lo que se debe conseguir para ser conforme al estándar, sin embargo no define cómo hacerlo. Con la realización de este proyecto ha sido posible observar que SCRUM comparte algunos conceptos con ISO 9001 respecto a la calidad y satisfacción del cliente de tal manera que puede ser implementado conforme a ISO 9001, para esto, fue necesario llevar a cabo un análisis detallado de los dos enfoques para identificar los conceptos comunes.

Este capítulo presenta la armonización llevada a cabo entre el estándar ISO 9001:2015 y SCRUM. A partir de la cual fue posible definir un proceso ágil para la gestión de proyectos de desarrollo de software, conservando las mejores prácticas de SCRUM y además, siendo conforme a los requisitos del estándar ISO 9001:2015, el proceso diseñado es descrito en detalle en el capítulo IV. Para llevar a cabo la armonización, se tuvo como referencia el conjunto de métodos denominado *HMethods* definido en [35], el cual provee información respecto a *¿Qué hacer?* y *¿Cómo proceder?* con la armonización de múltiples enfoques. *HMethods* define un método de homogeneización *HoMethod* y un método de integración *IMethod* que a su vez utiliza un método de comparación o mapping definido en [50]. Durante la armonización fue posible determinar los puntos en común y las aparentes diferencias entre los requisitos de los capítulos: 8. *Operación*, 9. *Evaluación del desempeño* y 10. *Mejora* del estándar ISO 9001:2015 y el enfoque ágil de desarrollo de software de SCRUM.

3.1. Homogeneización de Enfoques

Con el objetivo de garantizar la fiabilidad de los resultados obtenidos luego de realizar la armonización de los enfoques, se utilizó *HoMethods* [35], un método para la homogeneización de enfoques, cuyo objetivo es proveer una directriz que permita llevar a cabo la homogeneización paso a paso de diferentes enfoques; gestionando además las personas, actividades y pasos involucrados en el método. En ese orden de ideas, para llevar a cabo la homogeneización; *HoMethod* define 2 roles: (I) un intérprete y (II) un revisor como se observa en la Tabla 8.

Abreviatura	Rol	Competencias
I	Intérprete	Responsable del análisis de enfoques, quien implementa las técnicas de armonización. Esta persona debe tener habilidades de abstracción y análisis, además debe ser capaz de relacionar y comparar enfoques.
R	Revisor	Responsable de verificar la fiabilidad de los resultados obtenidos de la armonización.

Tabla 8. Roles definidos por HoMethod [35].

A continuación, se describe en detalle la aplicación de las actividades definidas en *HoMethod* durante la armonización de SCRUM y el estándar ISO 9001:2015.

3.1.1. Adquisición de conocimiento de los enfoques involucrados

HoMethod recomienda llevar a cabo un análisis comparativo desde una perspectiva de alto nivel de los enfoques involucrados, esto con el objetivo de identificar las similitudes o diferencias generales respecto a su naturaleza, objetivo, organización a cargo y factores relativos a la complejidad de aplicación, ver Tabla 9.

Atributos	ISO 9001:2015	SCRUM
Nombre	Sistemas de Gestión de la Calidad. Requisitos	The Definitive Guide to SCRUM: The Rules of the Game
Enfoque	Cualquier organización, todos los aspectos de la organización	Cualquier organización, producción del bien o servicio
Tamaño (Número de páginas)	47	16
Organización a cargo	Organización Internacional para la Estandarización (ISO)	No Aplica
Procesos o prácticas	7 capítulos principales	4 eventos principales
Versión	2015	2017

Tabla 9. Comparación de atributos de ISO 9001:2015 y SCRUM.

3.1.2. Análisis de estructura y terminología

3.1.2.1. ISO 9001

ISO 9001:2015 es un estándar internacional para la implementación de sistemas de gestión de la calidad (SGC) que abarca todos los aspectos de una organización, desde los deseos y necesidades de los clientes y sus partes interesadas, hasta la medición, evaluación y control del desempeño de los procesos internos, la naturaleza genérica del estándar permite la aplicación a la industria del desarrollo de software. Para obtener una certificación es necesario demostrar la conformidad de los procesos de software generando la evidencia necesaria del cumplimiento de los requisitos. El costo de implementación de ISO 9001 va de la mano de capacitaciones de personal, consultores, auditores internos y por supuesto el costo de la certificación como tal y las consecuentes re certificaciones.

La versión en español para Colombia²¹ de ISO 9001 en su versión 2015 está enmarcada en el ciclo PHVA (Planear, Hacer, Verificar, Actuar) que permite constantemente evaluar y mejorar el desempeño del sistema de gestión de la calidad a través de la toma de decisiones con base a la información obtenida del control del mismo. El estándar está conformado por capítulos, que a su vez contienen requisitos, los requisitos contienen literales y en algunos casos los literales contienen numerales. En total son 10 capítulos sin contar la introducción (capítulo 0), de los cuales desde el 1 hasta el 3 se comprende el objeto, campo de aplicación, referencias normativas y los términos y definiciones. Los requisitos de la norma están comprendidos desde el capítulo 4. *Contexto de la Organización* hasta el capítulo 10. *Mejora*. Los requisitos comprendidos entre los capítulos 4 y 7 están relacionados con la organización y el sistema de gestión de la calidad. La genericidad de la norma permite que pueda ser utilizada por cualquier organización que produzca bienes y/o servicios, sin embargo, la armonización se enfocará únicamente en los capítulos: 8. *Operación*, 9. *Evaluación del desempeño* y 10. *Mejora*, ya que son los que se relacionan directamente con el ciclo de vida del desarrollo de software.

Al analizar la sintáxis de algunos de los requisitos se pueden observar verbos en común que hacen referencia al nivel de cumplimiento que el estándar requiere, por ejemplo: a continuación se analiza el requisito 8.1 del estándar ISO 9001:2015 [3]:

“La organización **debe** planificar, implementar y controlar los **procesos** (véase 4.4) necesarios para **cumplir** los **requisitos** para la provisión de **productos y servicios**, y para implementar las acciones determinadas en el capítulo 6, mediante:

- a) la determinación de los requisitos para los productos y servicios;

²¹Norma Técnica Colombiana NTC-ISO 9001:2015. Sistemas de Gestión de la Calidad, Requisitos; provista por el Instituto Colombiano de Normas Técnicas y Certificación ICONTEC.

- b) el establecimiento de criterios para:
 - 1) los procesos;
 - 2) la aceptación de los productos y servicios;
- c) la determinación de los recursos necesarios para lograr la conformidad con los requisitos de productos y servicios;
- d) la implementación del control de los procesos de acuerdo con los criterios;
- e) la determinación, el mantenimiento y la conservación de la información de la información documentada en la extensión necesaria para:
 - 1) tener confianza en que los procesos se han llevado a cabo según lo planificado;
 - 2) demostrar la conformidad de los productos y servicios con sus requisitos;

La **salida** de esta planificación **debe** ser adecuada para las operaciones de la organización.

La organización **debe** controlar los cambios planificados y revisar las consecuencias de los cambios no previstos, tomando acciones para mitigar cualquier efecto adverso, según sea necesario.

La organización **debe** asegurarse de que los procesos contratados externamente estén controlados (véase 8.4).”

Como se puede observar, el verbo *deber* en tiempo presente “*debe*”, es común a través de los diferentes literales y párrafos del requisito e implica una obligatoriedad en el resto del predicado. Así mismo, se identifica la palabra *salida* que corresponde a un producto de trabajo resultante de las actividades obligatorias. En la Tabla 10 se puede observar la sintaxis de identificación de requisitos del estándar ISO 9001:2015 o WER²² de acuerdo a las actividades de HoMethod [35].

²² por sus siglas en inglés: Words that could Establish a Requirement, definido en [35].

Sintaxis (WER)	Description
[El/La/Los] [sujeto] [debe/no debe] [verbo],[verbo],...,[verbo] [para] [objetivo], [objetivo]..[objetivo]	Implica que es obligatorio que el sujeto, que puede ser la organización o uno varios roles; cumplan con las tareas, actividades, procedimientos y/o objetivos descritos por el requerimiento.
requisito	“necesidad o expectativa establecida, generalmente implícita u obligatoria”[51].
proceso	“Un proceso es un conjunto de actividades interrelacionadas que interactúan para transformar entradas en salidas” [51], “Conjunto de políticas, estructuras organizacionales, tecnologías, procedimientos, propósitos, objetivos, y productos de trabajo que son necesarios para diseñar, desarrollar, desplegar y mantener un producto software” [35].
entrada	Producto de trabajo necesario para un proceso.
salida	Producto de trabajo obtenido de un proceso.

Tabla 10. (WER) Sintaxis para identificar los requerimientos en ISO 9001:2015. Fuente: [35]

3.1.2.2. SCRUM

SCRUM por otra parte, es un enfoque ágil para la gestión de proyectos complejos ampliamente usado en el desarrollo de software, el cual se centra en la interacción de individuos, producción de valor de negocio y en garantizar la satisfacción del cliente, a diferencia de ISO 9001, SCRUM no abarca más allá de la gestión de proyectos en una organización. No existe una organización encargada de proveer SCRUM como tal, los creadores Ken Schwaber y Jeff Sutherland decidieron que SCRUM sería libre para cualquier organización que desee implementarlo además los costos de implementación son relativamente bajos, SCRUM es simple de entender pero difícil de aplicar [46], [47].

Al llevar a cabo un análisis de estructura y terminología de la guía de SCRUM de 2017 provista por Jeff Sutherland y Ken Schwaber [12], se pueden identificar las siguientes secciones del documento: (I) Propósito de la guía de SCRUM, (II) Definición de SCRUM, (III) Usos de SCRUM, (IV) Teoría de SCRUM, (V) Valores de SCRUM, (VI) El equipo SCRUM, (VII) Eventos de SCRUM y (VIII) Artefactos de SCRUM. Además, fue posible también concluir que la guía no tiene una sintaxis estándar o común para identificar lo que es un requisito para el cumplimiento de SCRUM, aunque si hay secciones que dan a

entender que algo se debe hacer de cierta manera, la sintáxis no se repite para todos los elementos del enfoque. Jeff y Ken recomiendan aplicar SCRUM de manera completa, incluso sugieren que aplicar parcialmente SCRUM no se debe considerar como SCRUM [12]. Los roles, actividades y artefactos identificados como obligatorios serán descritos en la estructura común de elementos de proceso de SCRUM en las tablas 32, 33 y 34.

3.1.3. Homogeneización de los enfoques usando una plantilla común de elementos de proceso

En esta actividad se llevó a cabo un análisis de la estructura y la terminología de los elementos en común.

Partiendo de la matriz de estructura común de elementos de procesos (CSPE) de alto nivel definida en [35], es posible encontrar los factores comunes entre ISO 9001 y SCRUM, esto con el fin de facilitar la comparación e integración de alto nivel. La correspondencia de los elementos del proceso se soporta con la ayuda de la experiencia y el criterio de investigadores expertos.

Sección	Estereotipos y elementos	ISO 9001:2015	SCRUM
Sección 1: Descripción(SD)	SD1 Grupos de procesos	X	
	SD2 Procesos		
	SD3 Actividades	X	X
	SD4 Tareas		
Sección 2: Roles y Recursos (SRR)	SRR1 Roles		X
	SRR2 Herramientas		
Sección 3: Control (SC)	SC1 Artefactos		X
	SC2 Objetivos	X	X
	SC3 Métricas		X

Tabla 11. Elementos comunes de estructuras de procesos. Fuente: [35]

3.1.3.1. Homogeneización del estándar ISO 9001:2015

La homogeneización de los modelos tiene como objetivo organizar la información de tal manera que se puedan expresar o representar los conceptos en ellos descritos, en un formato común para así poder llevar a cabo una comparación. Para ISO 9001:2015 se extrajeron partes específicas del estándar [3] a una plantilla de estructura común de elementos de proceso (CSPE) definida en [35], la cual fue ajustada de tal manera que se omitió el campo “SD2 Proceso” puesto que ISO 9001:2015 no define procesos específicos, sino grupos de proceso

por capítulo como *Operación, Evaluación del desempeño y Mejora*. Como se evidencia en la Tabla 11, un elemento en común en las estructuras de procesos de los enfoques son las **actividades**, por lo tanto el objetivo de la homogeneización a través de la plantilla es la fácil identificación de **actividades**. A continuación se presenta un extracto de la homogeneización del capítulo 8. *Operación*, en la cual se pueden evidenciar actividades según la estructura y terminología del estándar. Las tablas completas por capítulos 8, 9 y 10 se pueden encontrar como anexo en el **Anexo A**.

SD1. Categoría de proceso	8. Operación
ID	Capítulo 8.
Nombre	Operación.
Propósito	Definido implícitamente.
Descripción	Definida implícitamente
Objetivos	Definidos implícitamente
SD3. Actividades	
<p>“ Requisito 8.1 Planificación y control operacional: La organización debe planificar, implementar y controlar los procesos (véase 4.4) necesarios para cumplir los requisitos para la provisión de productos y servicios, y para implementar las acciones determinadas en el capítulo 6, mediante:</p> <ul style="list-style-type: none"> a) la determinación de los requisitos para los productos y servicios; b) el establecimiento de criterios para <ul style="list-style-type: none"> 1) los procesos; 2) la aceptación de los productos y servicios; c) la determinación de los recursos necesarios para lograr la conformidad con los requisitos de productos y servicios; d) la implementación del control de los procesos de acuerdo con los criterios; e) la determinación, el mantenimiento y la conservación de la información de la información documentada en la extensión necesaria para: 1) tener confianza en que los procesos se han llevado a cabo según lo planificado; 2) demostrar la conformidad de los productos y servicios con sus requisitos; <p>La salida de esta planificación debe ser adecuada para las operaciones de la organización. La organización debe controlar los cambios planificados y revisar las consecuencias de los cambios no previstos, tomando acciones para mitigar cualquier efecto adverso, según sea necesario. La organización debe asegurarse de que los procesos contratados externamente estén controlados (véase 8.4)” [3]</p>	
“ Requisito 8.2.1 Requisitos para los productos y servicios → Comunicación con el cliente... ”	
“ Requisito 8.2.2 Requisitos para los productos y servicios → Determinación de los requisitos para los productos y servicios:... ”	
“ Requisito 8.2.3.1 Requisitos para los productos y servicios → Revisión de los requisitos para los productos y servicios → La organización debe asegurarse de que tiene la capacidad de cumplir los requisitos para los productos y servicios que se van a ofrecer a los clientes.... ”	

“ Requisito 8.2.3.2 Requisitos para los productos y servicios → Revisión de los requisitos para los productos y servicios → La organización debe conservar la información documentada, cuando sea aplicable...”
“ Requisito 8.2.4 Requisitos para los productos y servicios → Cambios en los requisitos para los productos y servicios...”
“ Requisito 8.3.1 Diseño y desarrollo de los productos y servicios → Generalidades...”
“ Requisito 8.3.2 Diseño y desarrollo de los productos y servicios → Planificación del diseño y desarrollo...”
“ Requisito 8.3.3 Diseño y desarrollo de los productos y servicios → Entradas para el diseño y desarrollo...”
“ Requisito 8.3.4 Requisitos para los productos y servicios → Controles del diseño y desarrollo...”
“ Requisito 8.3.5 Requisitos para los productos y servicios → Salidas del diseño y desarrollo...”
“ Requisito 8.3.6 Requisitos para los productos y servicios → Cambios del diseño y desarrollo...”
“ Requisito 8.3.6 Requisitos para los productos y servicios → Cambios del diseño y desarrollo...”
“ Requisito 8.4.1 Control de los procesos, productos y servicios suministrados externamente → Generalidades...”
“ Requisito 8.4.2 Control de los procesos, productos y servicios suministrados externamente → Tipo y alcance del control...”
“ Requisito 8.4.3 Control de los procesos, productos y servicios suministrados externamente → Información para los proveedores externos...”
“ Requisito 8.5.1 Producción y provisión del servicio → Control de la producción y provisión del servicio...”
“ Requisito 8.5.2 Producción y provisión del servicio → Identificación y trazabilidad...”
“ Requisito 8.5.3 Producción y provisión del servicio → Propiedad perteneciente a los clientes o proveedores externos...”
“ Requisito 8.5.3 Producción y provisión del servicio → Propiedad perteneciente a los clientes o proveedores externos...”
“ Requisito 8.5.3 Producción y provisión del servicio → Propiedad perteneciente a los clientes o proveedores externos...”
“ Requisito 8.5.4 Producción y provisión del servicio → Preservación...”
“ Requisito 8.5.5 Producción y provisión del servicio → Actividades posteriores a la entrega...”
“ Requisito 8.5.6 Producción y provisión del servicio → Control de los cambios...”
“ Requisito 8.6 Liberación de los productos y servicios...”
“ Requisito 8.7.1 Control de las salidas no conformes → La organización debe asegurarse de que las salidas que no sean conformes con sus requisitos se identifican y se controlan para prevenir su uso o entrega no intencionada...”
“ Requisito 8.7.2 Control de las salidas no conformes → La organización debe conservar información documentada...”

Tabla 12. Extracto de Homogeneización del Capítulo 8. Operación, de ISO 9001:2015.

3.1.3.2. Homogeneización del enfoque ágil SCRUM

Dado que SCRUM no define oficialmente categorías o grupos de proceso como se evidenció en la Tabla 11, las descripciones de las actividades, roles y artefactos descritos en la guía oficial de 2017 [12] serán presentados en una versión ajustada de la plantilla de homogeneización propuesta en [35], dicha tabla tendrá el atributo “SD1. Categoría del elemento” en lugar de “SD1. Categoría de proceso”. Las categorías de elementos evidenciados en SCRUM corresponden a Roles, Eventos y Artefactos. De manera análoga a la homogeneización de ISO 9001:2015 en la tabla 31, se omite el campo “SD2. Proceso” y se orienta la plantilla a la identificación de **actividades** por cada categoría de elemento. A continuación se presenta un extracto de la homogeneización del enfoque ágil SCRUM, las tablas completas de la homogeneización por cada categoría de proceso se pueden encontrar en el **Anexo A**.

SD1. Categoría de elemento	Roles
ID	The SCRUM Team
Nombre	El Equipo SCRUM
Propósito	“Los equipos de SCRUM entregan productos iterativa e incrementalmente, maximizando oportunidades de retroalimentación. Entregas incrementales de producto “Hecho” aseguran que una versión potencialmente útil de producto funcional está siempre disponible.”
Descripción	“El equipo SCRUM consta de un Product Owner, el Equipo de Desarrollo y un SCRUM Master. Los equipos SCRUM son auto organizados y multifuncionales. Equipos auto organizados escogen la mejor forma de conseguir sus objetivos de trabajo, en lugar de ser dirigidos directamente por terceros fuera del equipo. Equipos multifuncionales tienen todas las competencias necesarias para lograr los objetivos de trabajo sin depender de otros que no son parte del equipo. El modelo del equipo SCRUM está diseñado para optimizar la flexibilidad, creatividad, y productividad. El equipo SCRUM ha probado ser por sí mismo incrementalmente efectivo para todos los usos de SCRUM y para cualquier trabajo complejo.”
Objetivos	Definidos implícitamente
SRR1. Roles	

Rol 1. El Product Owner: “ El Product Owner es el responsable de maximizar el valor del producto resultante del trabajo del Equipo de Desarrollo. La forma en que se lleva a cabo puede variar dependiendo de la organización, el equipo SCRUM e individuos.

El Product Owner es la única persona responsable de gestionar el **Product Backlog**. La gestión del **Product Backlog** incluye:

- Expresar claramente los ítems del **Product Backlog**;
- Ordenar los ítems en el **Product Backlog** para lograr metas y objetivos de la mejor manera;
- Asegurar que el **Product Backlog** es visible, transparente, y claro para todos, y muestra en lo que el equipo SCRUM trabajará, y
- Asegurar que el Equipo de Desarrollo entienda los ítems del **Product Backlog** al nivel necesario

El Product Owner puede llevar a cabo el trabajo mencionado previamente, o puede hacerlo el Equipo de Desarrollo. De cualquier manera, el Product Owner sigue siendo el responsable.

El Product Owner es una persona, no un comité. El Product Owner puede representar los deseos de un comité en el **Product Backlog**, pero aquellos que quieran modificar la prioridad de los ítems del **Product Backlog** deben dirigirse al Product Owner.

Para que el Product Owner tenga éxito, toda la organización debe respetarlo y respetar sus decisiones. Las decisiones del Product Owner son visibles en el contenido y ordenamiento del **Product Backlog**. Nadie puede forzar al Equipo de Desarrollo a trabajar a partir de otro conjunto de requerimientos.”

Tabla 13. Extracto de la homogeneización de los roles de SCRUM.

3.2. Comparación de enfoques

A través de la comparación de los enfoques se pudo determinar la relación entre los elementos de cada propuesta. Esta comparación permitió definir si un enfoque define o contiene los elementos del otro. Para llevar a cabo esta actividad se utilizó el método de *Mapping* definido en [50], que fue adaptado de tal manera que permitió efectuar una comparación desde una perspectiva de alto nivel entre los factores comunes identificados en la tabla 11. A continuación se describen las actividades realizadas para efectuar la comparación de los enfoques.

3.2.1. Diseño de la comparación

Esta actividad implica las siguientes tareas: (I) establecer las entidades que serán comparadas, (II) definir el grado de la relación, (III) establecer la direccionalidad de la comparación y (IV) definir una plantilla para llevar a cabo la comparación.

3.2.1.1. Establecer las entidades a ser comparadas en función de las necesidades de investigación

Durante la homogeneización fue posible identificar los elementos de proceso que son comunes entre los enfoques analizados. ISO 9001:2015 define 10 capítulos. De los cuales del 1 al 3 son introductorios al estándar, los capítulos 4, 5, 6 y 7 se consideran específicos por organización ya que su implementación depende de las necesidades de la misma. Y finalmente se consideraron relevantes para este proyecto los capítulos 8. *Operación*, 9. *Evaluación del desempeño* y 10. *Mejora* mapeados con el mismo nombre a grupos de proceso en la tablas que describen sus requisitos, respectivamente 29, 30, y 31. Cada requisito es el enunciado de una o más **actividades** que se deben realizar, sin embargo, no se dice cómo realizarlas. SCRUM por su parte, define roles, eventos y artefactos. En SCRUM cada evento comprende una serie de **actividades** que involucran roles y artefactos. Por lo tanto, las entidades a ser comparadas entre los enfoques, son las **actividades** que se deben llevar a cabo según los enunciados de los numerales en ISO 9001:2015, y las actividades definidas en los eventos de SCRUM, llevadas a cabo por uno o más roles y cuyas salidas son artefactos de SCRUM. La comparación de alto nivel está dada por capítulos de ISO 9001:2015 vs SCRUM, mientras que en un nivel más bajo se comparan numerales de capítulos de ISO 9001:2015 vs eventos, roles y/o artefactos de SCRUM.

3.2.1.2. Definir el grado de relación

De la Tabla 10, se pueden identificar las actividades en los requisitos que componen un capítulo de ISO 9001:2015; e.g. El requisito 8.1 del capítulo 8 está compuesto por actividades como:

- “La organización debe planificar, implementar y controlar los procesos necesarios para la provisión de productos y servicios y para implementar las acciones determinadas en el capítulo 6 mediante la determinación de los requisitos para los productos y servicios.”
- “La organización debe planificar, implementar y controlar los procesos necesarios para la provisión de productos y servicios y para implementar las acciones determinadas en el capítulo 6 mediante la determinación de los recursos necesarios para lograr la conformidad con los requisitos de los productos y servicios.”

De lo anterior se puede decir que un capítulo CH_i de ISO 9001:2015 está compuesto por un número R de requisitos, y cada requisito R_j tiene un conjunto de actividades. Asimismo, la comparación o relación debe llevarse a cabo a nivel de capítulo de ISO 9001:2015 vs SCRUM como un todo, y a nivel del requisito de un capítulo de ISO 9001:2015 vs Evento, Rol y/o Artefacto de SCRUM. De tal manera que si al analizar las actividades de un requisito en ISO 9001:2015 se determina que éstos pueden ser cumplidos o encuentran

equivalencia en un Evento, Rol y/o Artefacto de SCRUM, el requisito tendrá una relación de 1 a 1 con SCRUM, es decir que la *conformidad de SCRUM* C_s con respecto al requisito R_j será igual a 1, si las actividades del requisito R_j no pueden ser cumplidas por un Evento, Rol y/o Artefacto de SCRUM la *conformidad de SCRUM* C_s con respecto al requisito R_j será igual a 0. Posteriormente, la sumatoria de las conformidades de cada requisito del capítulo CH_i será dividida entre el total de requisitos R del capítulo CH_i de ISO 9001:2015; por lo tanto, el grado de relación entre un capítulo de ISO 9001:2015 y SCRUM será expresado porcentualmente como la *conformidad de SCRUM* C_s con respecto al capítulo CH_i como se muestra en la Figura 4:

$$C_sCH_i = \frac{\sum_{j=1}^R C_sR_j}{R}$$

tal que,

C_sCH_i es la conformidad de SCRUM con respecto al capítulo i de ISO 9001:2015

C_sR_j es la conformidad de SCRUM con respecto al requisito j

R es la cantidad de requisitos del capítulo i

Figura 4. Grado de relación de la comparación. Fuente: propia.

De igual manera, para expresar la conformidad de SCRUM con respecto a un capítulo de ISO 9001 C_sCH_i se utilizará la escala en el estándar ISO/IEC 15504[52] para evaluar el cumplimiento de atributos de proceso, también llamada *Escala de 4 puntos (N-P-L-F)*²³ como se muestra en la Tabla 14.

Etiqueta	(%) Conformidad de SCRUM con respecto al Capítulo de ISO 9001:2015
Conformidad Total.	SCRUM soporta entre 85.1 % y el 100 % de los numerales del capítulo.
Conformidad Alta.	SCRUM soporta entre el 50.1 % y el 85 % de los numerales del capítulo.
Conformidad Parcial.	SCRUM soporta entre el 15.1 % y el 50 % de los numerales del capítulo.
No Conforme.	SCRUM soporta entre el 0 % y el 15 % de los numerales del capítulo.

Tabla 14. Nomenclatura de comparación. Fuente: propia.

²³Not achieved, Partially achieved, Largely achieved, Fully achieved

3.2.1.3. Fijar la direccionalidad de la comparación

Teniendo en cuenta que ISO 9001 es un enfoque tradicional que define 36 requerimientos de obligatorio cumplimiento, y que SCRUM es un enfoque ágil compuesto de eventos, roles y artefactos que generalmente son adaptados por las organizaciones para darle su implementación única. Se tomará como base a ISO 9001 con el objetivo de buscar conformidad con los requerimientos en los diferentes elementos de SCRUM.

3.2.1.4. Definir una plantilla de comparación

La plantilla de comparación para esta actividad se definió tomando como referencia la plantilla definida en [35] y teniendo en cuenta la direccionalidad de la comparación definida anteriormente.

	Eventos				Roles			Artefactos			
Requisito	Sprint Planning	Daily SCRUM	Sprint Review	Sprint Retrospective	Product Owner	SCRUM Master	Development Team	Product Backlog	Sprint Backlog	Incremento	Definition of Done
8.1											
8.2											
...											
n											

Tabla 15. Plantilla de Comparación. Fuente: propia.

3.2.2. Comparación entre ISO 9001:2015 y SCRUM

A continuación se lleva a cabo la comparación de los enfoques analizados en este proyecto. Tomando como base a ISO 9001:2015 hacia la conformidad de sus requisitos en los eventos, roles y artefactos de SCRUM. La comparación se llevó a cabo analizando todas y cada una de las actividades de obligatorio cumplimiento encontradas en los requisitos de los capítulos 8, 9 y 10 de ISO 9001:2015 basándose en la Tabla 10 y su posible equivalencia o cumplimiento durante cada evento de SCRUM y con los respectivos artefactos y/o roles.

3.2.2.1. Comparación de entidades

Para llevar a cabo la comparación, se utilizó el método de *Mapping* en [50], en el cuál se toman elementos de dos enfoques con un determinado nivel de abstracción, y se aplica la teoría de conjuntos para realizar una unión, intersección o complemento entre los elementos. Para el caso específico de este proyecto, el *Intérprete* certificado como SCRUM Master, auditor interno de ISO 9001:2015, y además tutor en el estudio de caso, llevó a cabo el análisis de las 396 posibles relaciones entre los 36 requisitos de los capítulos 8,9 y 10 de ISO 9001:2015 y los 11 elementos de proceso de SCRUM. Identificando un total de 140 relaciones, que fueron a su vez validadas y depuradas por 3 *Revisores*, 2 de ellos director y codirector de este proyecto con una vasta experiencia en armonización de modelos, y una tercera revisora experta en sistemas de gestión de la calidad. Las tablas 16, 17 y 18 se limitan a mostrar cada relación 1 a 1 entre un requisito de ISO 9001:2015 y SCRUM a través de la etiqueta *rel_i*, donde *i* es un consecutivo único para cada relación, el detalle de cada relación puede ser encontrado en el **Anexo C**. Los diferentes eventos, roles y artefactos serán etiquetados con acrónimos para cada columna con el objetivo de mejorar la distribución de espacio, sin embargo, la etiqueta de color se mantendrá como en la Tabla 15.

Capítulo 8											
Requisito	SP	DS	SRV	SRP	PO	SM	DT	PB	SB	I	DD
8.1	<i>rel1</i>	<i>rel2</i>	<i>rel3</i>	<i>rel4</i>	<i>rel5</i>	<i>rel6</i>	<i>rel7</i>	<i>rel8</i>	<i>rel9</i>		<i>rel10</i>
8.2.1			<i>rel11</i>		<i>rel12</i>			<i>rel13</i>		<i>rel14</i>	
8.2.2											
8.2.3.1											
8.2.3.2											
8.2.4	<i>rel20</i>		<i>rel21</i>		<i>rel22</i>	<i>rel23</i>	<i>rel24</i>	<i>rel25</i>			
8.3.1	<i>rel26</i>	<i>rel27</i>	<i>rel28</i>	<i>rel29</i>	<i>rel30</i>	<i>rel31</i>	<i>rel32</i>	<i>rel33</i>	<i>rel34</i>	<i>rel35</i>	<i>rel36</i>
8.3.2	<i>rel37</i>	<i>rel38</i>	<i>rel39</i>	<i>rel40</i>	<i>rel41</i>	<i>rel42</i>	<i>rel43</i>	<i>rel44</i>	<i>rel45</i>	<i>rel46</i>	<i>rel47</i>
8.3.3	<i>rel48</i>				<i>rel49</i>			<i>rel50</i>	<i>rel51</i>	<i>rel52</i>	<i>rel53</i>
8.3.4	<i>rel54</i>		<i>rel55</i>		<i>rel56</i>	<i>rel57</i>	<i>rel58</i>	<i>rel59</i>	<i>rel60</i>	<i>rel61</i>	<i>rel62</i>
8.3.5			<i>rel63</i>		<i>rel64</i>	<i>rel65</i>	<i>rel66</i>	<i>rel67</i>	<i>rel68</i>	<i>rel69</i>	<i>rel70</i>
8.3.6	<i>rel71</i>		<i>rel72</i>		<i>rel73</i>			<i>rel74</i>			
8.4.1											
8.4.2											
8.4.3											
8.5.1											
8.5.2			<i>rel75</i>		<i>rel76</i>	<i>rel77</i>	<i>rel78</i>	<i>rel79</i>	<i>rel80</i>	<i>rel81</i>	
8.5.3											
8.5.4			<i>rel82</i>		<i>rel83</i>	<i>rel84</i>	<i>rel85</i>	<i>rel86</i>	<i>rel87</i>	<i>rel88</i>	
8.5.5											

8.5.6											
8.6			rel89		rel90	rel91		rel92	rel93	rel94	rel95
8.7.1	rel96		rel97		rel98	rel99	rel100	rel101	rel102		rel103
8.7.2			rel105		rel106	rel107	rel108	rel109	rel110		

Tabla 16. Comparación de actividades de los requisitos del capítulo 8 de ISO 9001:2015 con SCRUM.

A continuación se calcula la conformidad de SCRUM con respecto al capítulo 8 de ISO 9001:2015 dado por C_sCH_8 de acuerdo a la fórmula de la Figura4. Para el capítulo 8 hay un total de 24 requisitos, por tanto el valor de i es 8 y R es 24 como se muestra a continuación.

$$C_sCH_8 = \frac{\sum_{j=1}^{24} C_sR_j}{24}$$

Para los requisitos 8.2.2, 8.2.3.1, 8.2.3.2, 8.4.1, 8.4.2, 8.4.3, 8.5.1, 8.5.3, 8.5.5 y 8.5.6 no se encontró ninguna relación para el cumplimiento de sus actividades en los eventos, roles y artefactos de SCRUM, por tanto, la conformidad de SCRUM C_s para estos requisitos es 0, en los demás requisitos si se encontraron relaciones que permiten el cumplimiento de sus actividades, por tanto la conformidad de SCRUM C_s será igual a 1, finalmente la conformidad de SCRUM con respecto al capítulo 8 de ISO 9001:2015 está dada por la siguiente ecuación.

$$C_sCH_8 = \frac{(10 * 0) + (14 * 1)}{24}$$

$$C_sCH_8 = \frac{14}{24}$$

$$C_sCH_8 = 58\%$$

Figura 5. Conformidad de SCRUM con respecto al capítulo 8 de ISO 9001:2015 C_sC_8 .

Capítulo 9											
Requisito	SP	DS	SRV	SRP	PO	SM	DT	PB	SB	I	DD
9.1.1						rel112	rel113	rel114	rel115		
9.1.2			rel116	rel117							

9.1.3			<i>rel118</i>	<i>rel119</i>	<i>rel120</i>	<i>rel121</i>	<i>rel122</i>	<i>rel123</i>	<i>rel124</i>		
9.2.1											
9.2.2											
9.3.1											
9.3.2			<i>rel125</i>		<i>rel126</i>	<i>rel127</i>		<i>rel128</i>	<i>rel129</i>		
9.3.3											

Tabla 17. Comparación de actividades de los requisitos del capítulo 9 de ISO 9001:2015 con SCRUM.

A continuación se calcula la conformidad de SCRUM con respecto al capítulo 9 de ISO 9001:2015 dado por C_sCH_9 de acuerdo a la fórmula de la Figura 4. Para el capítulo 9 hay un total de 8 requisitos, por tanto el valor de i es 9 y R es 8 como se muestra a continuación.

$$C_sCH_9 = \frac{\sum_{j=1}^8 C_s R_j}{8}$$

Para los requisitos 9.2.1, 9.2.2, 9.3.1 y 9.3.3 no se encontró ninguna relación para el cumplimiento de sus actividades en los eventos, roles y artefactos de SCRUM, por tanto, la conformidad de SCRUM C_s para estos requisitos es 0, en los demás requisitos si se encontraron relaciones que permiten el cumplimiento de sus actividades, por tanto la conformidad de SCRUM C_s será igual a 1, finalmente la conformidad de SCRUM con respecto al capítulo 9 de ISO 9001:2015 está dada por la siguiente ecuación.

$$C_sCH_9 = \frac{(4 * 0) + (4 * 1)}{8}$$

$$c_sCH_9 = \frac{4}{8}$$

$$c_sCH_9 = 50\%$$

Figura 6. Conformidad de SCRUM con respecto al capítulo 9 de ISO 9001:2015 C_sCH_9 .

Capítulo 10											
Requisito	SP	DS	SRV	SRP	PO	SM	DT	PB	SB	I	DD
10.1											
10.2.1	<i>rel130</i>				<i>rel131</i>	<i>rel132</i>	<i>rel133</i>	<i>rel134</i>	<i>rel135</i>	<i>rel136</i>	

10.2.2				rel137	rel138		rel139	rel140			
10.3											

Tabla 18. Comparación de actividades de los requisitos del capítulo 10 de ISO 9001:2015 con SCRUM.

A continuación se calcula la conformidad de SCRUM con respecto al capítulo 10 de ISO 9001:2015 dado por C_sCH_{10} de acuerdo a la fórmula de la Figura 4. Para el capítulo 10 hay un total de 4 requisitos, por tanto el valor de i es 10 y R es 4 como se muestra a continuación.

$$C_sCH_{10} = \frac{\sum_{j=1}^4 C_s R_j}{4}$$

Para los requisitos 10.1 y 10.3 no se encontró ninguna relación para el cumplimiento de sus actividades en los eventos, roles y artefactos de SCRUM, por tanto, la conformidad de SCRUM C_s para estos requisitos es 0, en los demás requisitos si se encontraron relaciones que permiten el cumplimiento de sus actividades, por tanto la conformidad de SCRUM C_s será igual a 1, finalmente la conformidad de SCRUM con respecto al capítulo 10 de ISO 9001:2015 está dada por la siguiente ecuación.

$$C_sCH_{10} = \frac{(2 * 0) + (2 * 1)}{4}$$

$$C_sCH_{10} = \frac{2}{4}$$

$$C_sCH_{10} = 50\%$$

Figura 7. Conformidad de SCRUM con respecto al capítulo 10 de ISO 9001:2015 C_sCH_{10} .

Capítulo IV. Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015

Este capítulo presenta los diferentes procesos propuestos y su interacción para soportar la gestión de proyectos ágiles basándose en SCRUM, y en un sistema de gestión de calidad conforme a los requisitos de los capítulos 8, 9, y 10 del estándar ISO 9001:2015; Incluyendo un análisis de aquellos requisitos por capítulo para los cuales no se encontró relación alguna con SCRUM, y las alternativas o procesos genéricos que cualquier organización puede implementar para lograr la conformidad con el estándar.

4.1. Análisis de la conformidad entre SCRUM e ISO 9001:2015

A continuación, se analizan los resultados de la armonización para identificar los requisitos que SCRUM no puede satisfacer en primera instancia del estándar ISO 9001:2015. Asimismo, se presentan algunas consideraciones y alternativas para lograr la conformidad. El análisis de conformidad se organiza igual que la estructura de capítulos descritos por la norma ISO 9001:2015.

4.1.1. Capítulo 8. Operación

Con relación al capítulo 8, se encontraron las siguientes no conformidades:

- Como se observa en la Tabla 16, los requisitos 8.2.2, 8.2.3.1 y 8.2.3.2 que corresponden a la *determinación y revisión de los requisitos para los productos y servicios* de la norma ISO 9001:2015, no se identificó una relación con los elementos de SCRUM, ésto, debido a que SCRUM no define eventos para la definición de los requisitos del producto, más bien se asume que los requisitos existen y alimentan el artefacto *Product Backlog*. Por lo tanto, para conseguir conformidad con el estándar, es necesario que la organización defina un proceso para la definición y revisión de los requisitos para los productos y servicios.
- Asimismo, en la Tabla 16 se puede evidenciar que los requisitos 8.4.1, 8.4.2 y 8.4.3 de la norma ISO 9001:2015 y que corresponden al *control de los procesos, productos y servicios suministrados externamente*, no se relacionaron con los elementos de SCRUM; esto debido a que SCRUM no aborda el tema de servicios externos. Por lo tanto, para conseguir conformidad con el estándar, la organización debe definir los procesos necesarios para controlar los productos y servicios suministrados externamente donde sea aplicable.

- También se puede observar en la Tabla 16 que el requisito 8.5.1 correspondiente al *control de la producción y provisión del servicio*, no encontró ninguna relación con los elementos de SCRUM, ya que aunque un Sprint proporciona entregables funcionales de producto o servicio en el artefacto Incremento, no aborda la forma en que se hace provisión del servicio a los clientes, incluyendo factores como la infraestructura, los recursos necesarios de seguimiento y medición, el personal, actividades de liberación y posteriores a la entrega entre otros. Por lo tanto para conseguir conformidad con el estándar, la organización debe definir un proceso adecuado para la gestión de la provisión del servicio.
- El requisito 8.5.3 de la norma ISO 9001:2015, relacionada con *la propiedad perteneciente a los clientes o proveedores externos*, tampoco encontró una relación con los elementos de SCRUM como se evidencia en la Tabla 16, ya que aunque el Product Owner es el responsable de la propiedad intelectual del cliente al gestionar los requisitos para los productos y servicios, el estándar abarca mucho más con respecto a la propiedad perteneciente a los clientes o proveedores externos, incluyendo identificación, verificación y protección durante la provisión y prestación del servicio, es decir que la organización debe definir un proceso con respecto a la seguridad de la información de la propiedad perteneciente a los clientes o proveedores externos durante la provisión y prestación del servicio.
- Finalmente, los requisitos 8.5.5 y 8.5.6 de la norma ISO 9001:2015, correspondientes a las *actividades posteriores a la entrega y control de cambios respectivamente*, no se encontró relación con los elementos de SCRUM, ya que naturalmente SCRUM se enfoca en la construcción del producto más no en el mantenimiento y prestación del servicio. Por lo tanto, la organización debe dar cumplimiento a estos requisitos a través de un proceso para la gestión de la provisión del servicio.

4.1.2. Capítulo 9. Evaluación del desempeño

Como se observa en la Tabla 17, los requisitos 9.2.1, 9.2.2, 9.3.1 y 9.3.3 de la norma y los correspondientes a la *realización de auditorías internas y revisiones por la dirección*, no se relacionaron con los elementos de SCRUM, incluso, aunque SCRUM provee insumos relacionados con la satisfacción del cliente y el desempeño del proceso de diseño y desarrollo, esto, debido a que naturalmente SCRUM no tiene eventos que mencionan la realización de dichas actividades. La organización debe definir los procesos necesarios con respecto a la realización de auditorías internas y revisiones por la dirección para cumplir con el estándar.

4.1.3. Capítulo 10. Mejora

En la Tabla 18 se puede observar que los requisitos 10.1 y 10.3 de la norma, correspondientes a las *oportunidades de mejora* y *mejora continua*, no se relacionaron con los elementos de SCRUM, incluso, aunque el evento *SCRUM Retrospective* aborda la autoevaluación y mejora continua del proceso de diseño y desarrollo, el estándar requiere un proceso formal para determinar y seleccionar las oportunidades de mejora del sistema de gestión de la calidad.

4.2. Consideraciones y alternativas para lograr la conformidad en los requisitos no relacionados

Con relación al requisito 8.7 de la norma, la cual está relacionada con *el control de las salidas no conformes*, la organización debe incluir en el *Sprint* actividades de pruebas de software para dar cumplimiento al estándar [38]. Por lo tanto, se han definido actividades genéricas de pruebas de software en el proceso *Fabricación del código* (ver ítem 22) de la propuesta de proceso integrado.

Teniendo en cuenta el análisis de conformidad realizado (ver ítem 4.1), para que una organización pueda dar conformidad al estándar ISO 9001:2015 en los requisitos no relacionados durante la comparación con los elementos del enfoque ágil SCRUM, la organización debe definir los siguientes procesos:

- **Proceso de adquisición de requisitos de software:** Este proceso debe tener como salida el conjunto de requisitos revisados y aprobados para los productos y servicios, incluyendo los legales y reglamentarios aplicables.
- **Proceso de gestión de los productos y servicios suministrados externamente:** Donde sea aplicable, la organización debe definir los procesos necesarios para el control de los productos y servicios suministrados externamente de acuerdo al requisito 8.4 de la norma.
- **Proceso de producción y provisión del servicio:** Donde sea aplicable, la organización debe definir los procesos necesarios para abordar temas como el despliegue y mantenimiento del producto software, incluyendo los recursos necesarios de infraestructura, seguridad de la información, personal de atención al cliente, entre otros.

Además de lo mencionado anteriormente, la organización debe definir los procesos necesarios para dar cumplimiento a los requisitos del capítulo 9. *Evaluación del desempeño*, correspondientes a la realización de auditorías internas y revisiones por la dirección, al igual

que los requisitos del capítulo 10. *Mejora*, correspondiente a las oportunidades de mejora y mejora continua.

4.3. Proceso integrado para la gestión de proyectos ágiles con SCRUM e ISO 9001:2015

Teniendo en cuenta los resultados de la revisión sistemática de la literatura (ver ítem 2.3.4), la armonización realizada a través de la homogenización y comparación de actividades de los enfoques SCRUM e ISO 9001:2015 en las Tablas 16, 17 y 18, y el análisis de los requisitos que no encontraron conformidad en SCRUM (ver ítem 4.2); se presenta a continuación, la propuesta del proceso integrado para la gestión de proyectos de software basado en SCRUM e ISO 9001:2015. Se utilizó una versión ajustada de la plantilla de proceso de COMPETISOFT en [53]. Para los ítems del Product Backlog se utilizaron *Historias de Usuario*, sin embargo, la organización puede utilizar el artefacto que mejor se adapte a sus necesidades. En las descripciones de los procesos se conservan algunos nombres originales en inglés de los elementos de proceso, debido a que son ampliamente utilizados y entendidos. Una guía electrónica que permite observar los modelos de procesos y sus descripciones de forma más amigable al lector está disponible <http://artemisa.unicauca.edu.co/~miguelmorcillo/public>.

4.3.1. Proceso de Diseño y Desarrollo

Definición General del proceso	
Proceso	Diseño y Desarrollo
Propósito	El propósito de este proceso es soportar la gestión de proyectos ágiles de desarrollo de software en MiPyMEs utilizando SCRUM y ofreciendo conformidad con el estándar ISO 9001:2015
Descripción	Este es el proceso de más alto nivel en la propuesta de proceso integrado, contiene al resto de subprocessos definidos en este proyecto y está dividido en 3 grandes etapas: <i>Diseño</i> , <i>Desarrollo</i> y <i>Revisión</i> . Interactúa con los procesos definidos en el ítem 4.2 correspondientes a <i>Proceso de adquisición de requisitos de software</i> y <i>Proceso de producción y provisión del servicio</i> respectivamente antecesor y sucesor. Asimismo contiene a las actividades: <i>Sprint Review</i> , <i>Sprint Retrospective</i> . Y a las actividades de toma de decisión: <i>¿Producto terminado?</i> y <i>¿Hay solicitudes de cambio?</i>
Objetivos	Definidos implícitamente
Indicadores	

Responsabilidad	<i>SCRUM Team</i>		
Subprocesos	<i>Planeación del Sprint, Ejecución del Sprint, Producción y provisión del servicio</i>		
Entradas	Nombre	Fuente	
	<i>Product Backlog de Alto Nivel</i>	<i>Proceso de adquisición de requisitos de software</i>	
Salidas	Nombre	Destino	
	<i>Incremento del Producto</i>	<i>Producción y provisión del servicio</i>	
Roles involucrados	Abreviatura	Rol	Descripción
	SM	SCRUM Master	ver Tabla 32
	PO	Product Owner	ver Tabla 32
	DT	Development Team	ver Tabla 32
Actividades			
Rol	Descripción		
A1. Sprint Planning, etapa 1.			
Entradas	<i>Product Backlog de alto nivel</i>		
SM,PO,DT	Es la etapa 1 del proceso de <i>Planeación del Sprint</i> en el que intervienen el <i>Product Owner</i> , el <i>Development Team</i> y el <i>SCRUM Master</i> . Este proceso se enfoca en la documentación detallada de historias de usuario y la definición del alcance del sprint, recibe como entrada el <i>Product Backlog de alto nivel</i> y cuando se llega de un sprint anterior, recibe un <i>Product Backlog detallado</i> y posibles solicitudes de cambio provenientes de la revisión del último sprint, el resultado es un <i>Product Backlog detallado</i> ajustado según cambios solicitados		
Salidas	<i>Product Backlog detallado</i>		
A2. Sprint Planning, etapa 2.			
Entradas	<i>Product Backlog detallado</i>		
SM,PO,DT	Es la etapa 2 del proceso de <i>Planeación del Sprint</i> en el que intervienen el <i>Development Team</i> y el <i>SCRUM Master</i> . Este proceso se enfoca en el diseño y plan de trabajo para las historias de usuario definidas en el alcance del sprint, recibe como entrada un <i>Product Backlog detallado</i> y da como resultado un <i>Sprint Backlog</i> en la herramienta de gestión y artefactos de diseño actualizados.		
Salidas	<i>Sprint Backlog, Artefactos de diseño</i>		
A3. Ejecución del Sprint.			
Entradas	<i>Sprint Backlog, Artefactos de diseño</i>		

SM,PO,DT	Es el proceso de <i>Ejecución del Sprint</i> en el que intervienen el <i>Product Owner</i> , el <i>Development Team</i> y el <i>SCRUM Master</i> . Este proceso recibe como entrada un <i>Sprint Backlog</i> y artefactos de diseño, el resultado es un <i>Incremento del Producto</i> probado y aprobado para presentar al cliente.
Salidas	<i>Incremento del Producto</i>
A4. Sprint Review.	
Entradas	<i>Incremento del Producto</i>
SM,PO,DT	El <i>SCRUM Team</i> presenta al cliente el <i>Incremento del producto</i> , resultante de la <i>Ejecución del Sprint</i> . El cliente proporciona retroalimentación acerca de su satisfacción con el incremento presentado diligenciando la <i>Revisión del sprint</i> . Además el cliente y sus interesados deciden si el <i>Incremento del producto</i> debe ser puesto en producción (lanzamiento), si hay lanzamiento el proceso continúa con el proceso de <i>Producción y Provisión del Servicio</i> . Si no, el proceso continúa con la actividad <i>Sprint Retrospective</i> .
Salidas	<i>Revisión del Sprint</i>
A5. Producción y provisión del servicio.	
Entradas	<i>Incremento del producto</i>
SM,PO,DT	Este proceso debe estar definido por la organización, corresponde a las actividades de despliegue y mantenimiento posteriores a la entrega.
Salidas	
A6. Sprint Retrospective.	
Entradas	
SM,PO,DT	El <i>SCRUM Team</i> conformado por el <i>Product Owner</i> , el <i>SCRUM Master</i> y el <i>Development Team</i> se reúnen para reflexionar acerca de la iteración que ha terminado, se habla sobre las buenas prácticas realizadas y las que se pueden mejorar. En esta actividad se documentan las <i>Lecciones aprendidas</i> en una herramienta de gestión para las mismas.
Salidas	<i>Lecciones aprendidas</i>
A7. ¿Producto Terminado?	
Entradas	<i>Product Backlog</i>
SM,PO,DT	En esta actividad se determina si el producto ha sido terminado, en tal caso el proceso de <i>Diseño y desarrollo</i> finaliza, de lo contrario el proceso continúa con la actividad <i>¿Hay solicitudes de cambio?</i> .
Salidas	<i>Sprint Backlog</i>
A8. ¿Hay solicitudes de cambio?	
Entradas	<i>Revisión del sprint</i>

SM,PO,DT	En esta actividad se verifica si hay solicitudes de cambio generadas en la última Revisión del Sprint, en tal caso el flujo del proceso vuelve a comenzar desde la etapa 1 de la <i>Planeación del Sprint</i> , en caso contrario el proceso vuelve a comenzar desde la etapa 2 de la <i>Planeación del Sprint</i> .
Salidas	<i>Product Backlog detallado, Solicitud de Cambio</i>
Flujo BPMN	

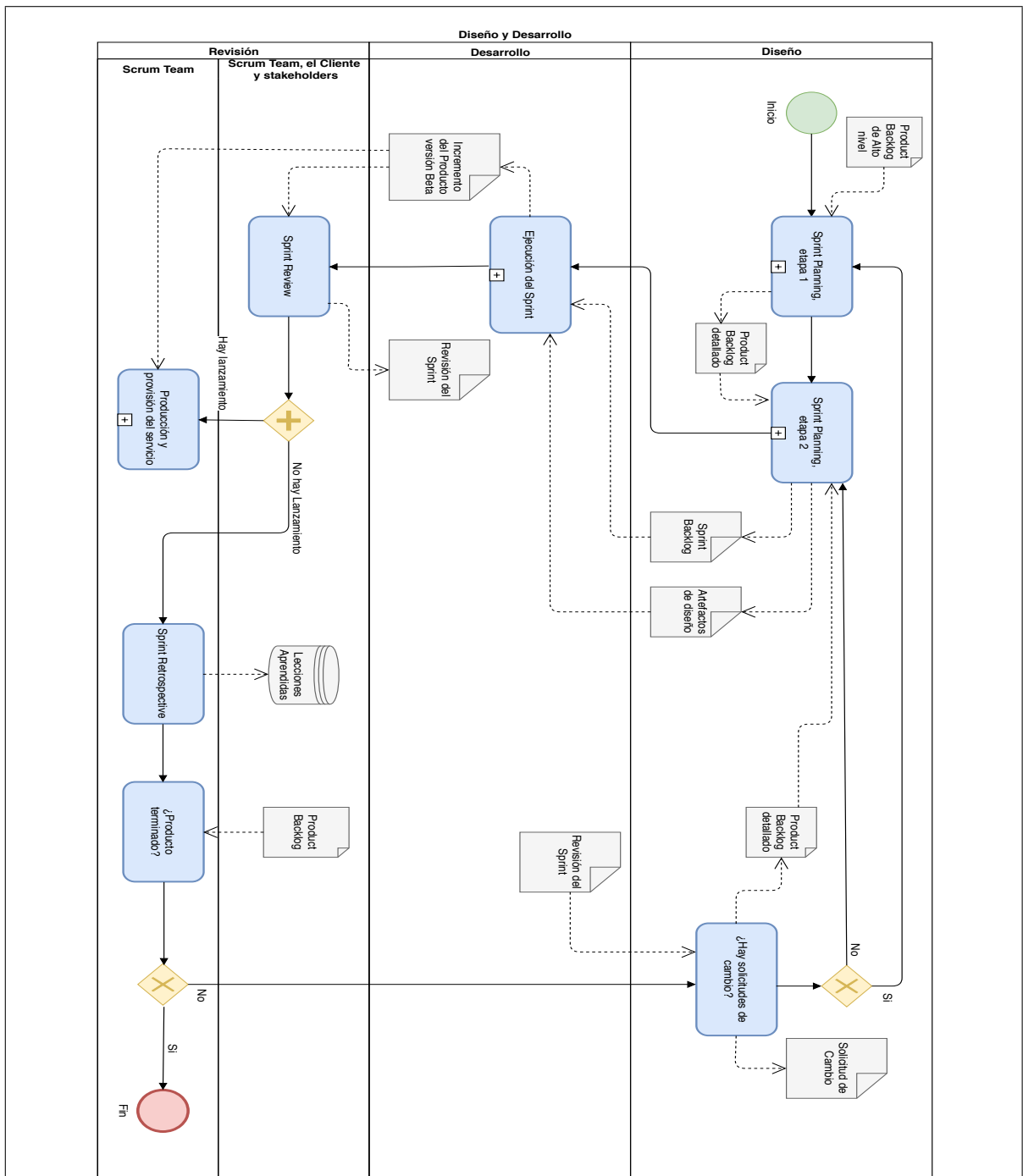


Tabla 19. Proceso de Diseño y Desarrollo

4.3.2. Proceso de Planeación del Sprint

Definición General del proceso			
Proceso	Planeación del Sprint		
Propósito	El propósito de este proceso es profundizar en la definición de los ítems del <i>Product Backlog</i> , y priorizarlos y estimarlos para definir el alcance del sprint.		
Descripción	Este proceso contiene todas las actividades para el diseño del producto y está dividido en 2 etapas. La etapa 1 incluye la participación del <i>Product Owner</i> que representa los deseos y necesidades del cliente, contribuyendo a la mejor definición de los ítems del <i>Product Backlog</i> y priorizando por valor de negocio, de esta manera el <i>Product Owner</i> y el <i>Development Team</i> negocian el alcance del <i>Sprint</i> finalizando la etapa 1. La etapa 2 no incluye al <i>Product Owner</i> ya que las actividades que se realizan son más técnicas y de autogestión del <i>Development Team</i> y el <i>SCRUM Master</i> , la etapa 2 finaliza cuando se ha definido el <i>Sprint Backlog</i> .		
Objetivos	Definidos implícitamente		
Indicadores			
Responsabilidad	<i>SCRUM Team</i>		
Subprocesos			
Entradas	Nombre	Fuente	
	<i>Product Backlog de Alto Nivel</i>	<i>Proceso de adquisición de requisitos de software</i>	
Salidas	Nombre	Destino	
	<i>Sprint Backlog</i>	<i>Ejecución del Sprint</i>	
Roles involucrados	Abreviatura	Rol	Descripción
	SM	SCRUM Master	ver Tabla 32
	PO	Product Owner	ver Tabla 32
	DT	Development Team	ver Tabla 32
	PPGC	Personal del proceso de gestión comercial	Son los encargados de negociar contractualmente con el cliente.
Actividades			
Rol	Descripción		
A1. ¿Es el primer Sprint?			
Entradas	<i>Product Backlog de alto nivel, Product Backlog de alto detallado Solicitud de Cambio</i>		

SM,PO,DT	Dependiendo del contexto el <i>Product Owner</i> decide el camino a seguir en este proceso, si es el primer sprint el proceso continúa con la actividad <i>Detallar Product Backlog</i> tomando como entrada el <i>Product Backlog de alto nivel</i> . Si no es el primer sprint el proceso continúa con la actividad <i>Revisar si los cambios son viables</i> tomando como entrada el <i>Product Backlog detallado</i> y las posibles Solicitudes de cambio.
Salidas	
A2. Detallar Product Backlog.	
Entradas	<i>Product Backlog de alto nivel</i>
SM,PO,DT	El <i>Product Owner</i> y el <i>Developmet Team</i> parten del <i>Product Backlog de alto nivel</i> para comenzar a detallar y extender los ítems generando así el <i>Product Backlog Detallado</i> .
Salidas	<i>Product Backlog detallado</i>
A3. ¿Los cambios son viables?	
Entradas	<i>Contrato Solicitud de Cambio</i>
SM,PO,DT	El <i>Product Owner</i> y el personal del proceso de gestión comercial revisan las solicitudes de cambio provenientes de una <i>Sprint Review</i> inmediatamente anterior para determinar si los cambios tienen lugar dentro del contrato acordado. Si los cambios son viables, el proceso continúa con la actividad <i>Ajustar requerimientos</i> . Si no, continúa con la actividad <i>Ajustar contrato</i> .
Salidas	<i>Ajustes a los requerimientos</i>
A4. Ajustar Contrato.	
Entradas	<i>Contrato</i>
PPGC	Cuando el cliente ha solicitado cambios que el <i>Product Owner</i> y el proceso de gestión comercial consideran fuera del contrato. Este último es responsable de ajustar el contrato para llegar a un acuerdo con respecto a los cambios solicitados.
Salidas	<i>Contrato</i>
A5. Ajustar Product Backlog.	
Entradas	<i>Ajustes a los requerimientos</i>
SM,PO,DT	El <i>Product Owner</i> realiza los ajustes necesarios a los requerimientos como consecuencia de una <i>Solicitud de cambios</i> , modificando los ítems que lo requieran. El resultado es un <i>Product Backlog de alto nivel</i> actualizado.
Salidas	<i>Product Backlog actualizado</i>
A6. Priorizar Product Backlog.	
Entradas	<i>Product Backlog actualizado</i>

SM,PO,DT	Una vez que se tiene el <i>Product Backlog actualizado</i> . El <i>Product Owner</i> hace una re priorización de ítems de acuerdo al valor de negocio de cada una.
Salidas	<i>Product Backlog priorizado</i>
A7. Estimar Historias de Usuario y priorizar por complejidad.	
Entradas	<i>Product Backlog detallado</i>
DT	El <i>Development Team</i> lleva a cabo una estimación para determinar la complejidad de los ítems del <i>Product Backlog</i> y posteriormente se priorizan según esa variable.
Salidas	<i>Product Backlog estimado</i>
A8. Definir alcance del Sprint.	
Entradas	<i>Product Backlog estimado</i>
PO,DT	El <i>Product Owner</i> y el <i>Development Team</i> definen el alcance del sprint seleccionando los ítems del <i>Product Backlog</i> que se van a trabajar, dependiendo del valor de negocio, la complejidad y la velocidad del equipo. El resultado es el <i>Sprint Backlog</i> en la herramienta de gestión, con los ítems que se trabajarán en el <i>Sprint</i> .
Salidas	<i>Sprint Backlog</i>
A9. Crear sprint en la herramienta de gestión.	
Entradas	<i>Sprint Backlog</i>
SM	El <i>SCRUM Master</i> crea el sprint en la herramienta de gestión, incluyendo los ítems acordados en la actividad anterior y las fechas de inicio y final del sprint. Desde esta actividad en adelante durante la iteración, el <i>Sprint Backlog</i> se gestionará en la herramienta de gestión web.
Salidas	<i>Sprint Backlog</i>
A10. Resolver dudas de implementación de Historias de Usuario.	
Entradas	<i>Sprint Backlog</i>
DT	El <i>Development Team</i> manifiesta y resuelve las dudas de implementación de los ítems del <i>Sprint Backlog</i> que por su naturaleza o complejidad necesitan información adicional respecto al diseño o implementación.
Salidas	<i>Sprint Backlog</i>
A11. Definir o actualizar artefactos de diseño.	
Entradas	<i>Sprint Backlog</i>

DT	El <i>Development Team</i> define o actualiza los artefactos de diseño que sean necesarios para enriquecer los ítems del <i>Sprint Backlog</i> , y para conservar información documentada del diseño de los productos y servicios. Entre los artefactos de diseño pueden estar diagramas o artefactos UML, entre otros.
Salidas	<i>Artefactos de diseño</i>
A12. Auto asignar historias de usuario.	
Entradas	<i>Sprint Backlog</i>
DT	El equipo de desarrollo auto gestionado, asigna y distribuye las responsabilidades de historias de usuario entre los miembros del equipo.
Salidas	<i>Sprint Backlog</i>
Flujo BPMN(El modelo del proceso se puede ver más grande en el Anexo B.)	

4.3.3. Proceso de Ejecución del Sprint

Definición General del proceso			
Proceso	Ejecución del Sprint		
Propósito	El propósito de este proceso es construir el <i>Incremento del Producto</i> utilizando las prácticas ágiles de SCRUM y la rigurosidad de ISO 9001:2015 para garantizar la calidad y la satisfacción del cliente.		
Descripción	Este proceso define un flujo general para la ejecución de un sprint, que dura por lo general entre 2 y 4 semanas. Contiene actividades de seguimiento como la reunión diaria, solución de impedimentos, pruebas de caja negra y además contiene al sub proceso <i>Fabricación del Código</i> que encapsula todo lo relativo al desarrollo del producto software.		
Objetivos	Definidos implícitamente		
Indicadores	Velocidad de equipo		
Responsabilidad	<i>SCRUM Team</i>		
Subprocesos	<i>Fabricación del Código</i>		
Entradas	Nombre	Fuente	
	<i>Sprint Backlog</i>	<i>Proceso de Planeación del Sprint</i>	
Salidas	Nombre	Destino	
	<i>Incremento del Producto</i>	<i>Proceso de Diseño y Desarrollo</i>	
Roles involucrados	Abreviatura	Rol	Descripción
	SM	SCRUM Master	ver Tabla 32
	PO	Product Owner	ver Tabla 32
	DT	Development Team	ver Tabla 32
Actividades			
Rol	Descripción		
A1. Participar/asistir a reunión diaria.			
Entradas			
SM,PO,DT	El <i>Product Owner</i> , el <i>SCRUM Master</i> y el <i>Development Team</i> se reúnen cada día durante 15 minutos. El <i>Product Owner</i> puede asistir pero no debe participar, esto para mantener al equipo libre de presiones y enfocado totalmente en el <i>Sprint Backlog</i> . En cada reunión los miembros del equipo responden a las preguntas ¿Qué hiciste desde la última reunión? ¿Qué harás hasta la próxima reunión? ¿Qué impedimentos has encontrado?. El SCRUM Master documenta la reunión en el <i>Acta de reunión</i> .		
Salidas	<i>Acta de reunión Impedimentos</i>		

A2. Proceso de Fabricación del Código.	
Entradas	<i>Sprint Backlog</i>
DT	El proceso de <i>Fabricación del Código</i> recibe como entrada un <i>Sprint Backlog</i> en la herramienta de gestión, que se va actualizando a lo largo del <i>Sprint</i> . Cuando ha finalizado, el producto de salida es un <i>Incremento del producto</i> listo para pruebas del Product Owner.
Salidas	<i>Incremento del producto</i>
A3. Monitorear desempeño y progreso.	
Entradas	<i>Sprint Backlog</i>
SM	El <i>SCRUM Master</i> monitorea el desempeño y progreso del proceso <i>Fabricación del código</i> a través de la herramienta de gestión, tal herramienta puede proveer indicadores como: <ul style="list-style-type: none"> ➤ El porcentaje del proyecto completado, ➤ La velocidad del equipo y ➤ El porcentaje del Sprint completado, entre otros.
Salidas	<i>Indicador</i>
A4. Verificar que la metodología se cumple.	
Entradas	
SM	El <i>SCRUM Master</i> debe verificar en todo momento que la metodología se esté cumpliendo por parte de todos los roles, estando siempre presente acompañando al equipo durante el sprint, resolviendo dudas de la metodología, verificando que las actividades sean desempeñadas y corrigiendo el rumbo cuando es necesario.
Salidas	
A5. Controlar y solucionar impedimentos.	
Entradas	<i>Impedimentos</i>
SM	El <i>SCRUM Master</i> debe velar por el bienestar del equipo de desarrollo, resolviendo y documentando los impedimentos encontrados y alimentando la base del conocimiento.
Salidas	
A6. ¿Sprint terminado?	
Entradas	<i>Sprint Backlog</i>
DT	El <i>Development Team</i> revisa constantemente el <i>Sprint Backlog</i> en la herramienta de gestión para verificar si hay tareas pendientes, cuando el <i>Sprint Backlog</i> no tiene más tareas el proceso continúa con la actividad <i>Realizar pruebas de caja negra</i> , si el <i>Sprint Backlog</i> tiene ítems por implementar, el proceso vuelve al inicio.
Salidas	

A7. Realizar pruebas de caja negra.	
Entradas	<i>Incremento del Producto, Sprint Backlog</i>
PO	Cuando el trabajo del <i>Sprint Backlog</i> se ha terminado, el <i>Development Team</i> entrega un incremento del producto listo para pruebas de caja negra al <i>Product Owner</i> , quien realiza las pruebas de cada ítem del <i>Sprint Backlog</i> sobre el <i>Incremento del producto</i> , como resultado de las pruebas el <i>Product Owner</i> diligencia los <i>Resultados de pruebas</i>
Salidas	<i>Resultados de pruebas</i>
A8. ¿Producto aprobado?	
Entradas	<i>Resultados de pruebas</i>
PO	El <i>Product Owner</i> decide si el producto es aprobado o si se deben corregir errores. Cuando existen errores, se vuelve al inicio del proceso y el <i>Development Team</i> actualiza el <i>Sprint Backlog</i> en la herramienta de gestión. En el caso contrario, el proceso de <i>Ejecución del Sprint</i> finaliza produciendo un incremento del producto aprobado por el <i>Product Owner</i> .
Salidas	<i>Incremento del Producto</i>
Flujo BPMN (El modelo del proceso se puede ver más grande en el Anexo B)	

4.3.4. Proceso de Fabricación del Código

Definición General del proceso			
Proceso	Fabricación del código		
Propósito	El propósito de este proceso es definir de manera genérica las actividades necesarias para construir código de manera organizada, sistemática y realizando las pruebas necesarias para garantizar la calidad y el buen funcionamiento del producto.		
Descripción	Este proceso describe las prácticas básicas durante el desarrollo de software con respecto al control de versiones, integración y pruebas conforme al estándar ISO 9001:2015 [38], cada organización puede ajustarlo para adaptarlo a sus necesidades.		
Objetivos	Definidos implícitamente		
Indicadores			
Responsabilidad	<i>Development Team</i>		
Subprocesos			
Entradas	Nombre		Fuente
	<i>Sprint Backlog</i>		<i>Proceso de Ejecución del Sprint</i>
Salidas	Nombre		Destino
	<i>Incremento del Producto</i>		<i>Proceso de Ejecución del Sprint</i>
Roles involucrados	Abreviatura	Rol	Descripción
	DT	Development Team	ver Tabla 32
Actividades			
Rol	Descripción		
A1. Desarrollar.			
Entradas	<i>Sprint Backlog, Documentos de referencia</i>		
DT	El <i>Development Team</i> implementa las historias de usuario del sprint descritas en el Sprint Backlog, dando lugar a nuevas funcionalidades. Esta actividad puede recibir como entrada documentos que la organización defina con respecto a buenas prácticas o directrices para el desarrollo.		
Salidas	<i>Funcionalidad</i>		
A2. Guardar progreso en repositorio.			
Entradas	<i>Funcionalidad</i>		

DT	Se recomienda que cada desarrollador guarde el progreso en una rama específica para la funcionalidad que desarrolla, asumiendo que la organización utiliza un sistema de control de versiones, esto para evitar pérdidas de trabajo en caso de mal funcionamiento del equipo de cómputo.
Salidas	<i>Funcionalidad</i>
A3. Realizar pruebas unitarias.	
Entradas	<i>Funcionadlidad</i>
DT	El equipo de desarrollo, o bien cada desarrollador debe realizar pruebas unitarias de cada nueva funcionalidad, el flujo del proceso no continuará hasta que las pruebas sean satisfactorias.
Salidas	<i>Funcionalidad</i>
A4. ¿Trabajo terminado?	
Entradas	<i>Funcionalidad</i>
DT	Cada vez que se ha terminado y probado una funcionalidad, el equipo de desarrollo revisa, actualiza y verifica el progreso del <i>Sprint Backlog</i> . Si se han terminado todas las funcionalidades, se continúa con la actividad <i>Integrar nuevas funcionalidades</i> , si aún hay trabajo pendiente el flujo del proceso vuelve a la actividad <i>Desarrollar</i>
Salidas	<i>Funcionalidades</i>
A5. Integrar nuevas funcionalidades.	
Entradas	<i>Funcionalidades</i>
DT	Cuando se han terminado de implementar las nuevas funcionalidades, el <i>Development Team</i> procede a integrarlas (merge) con el <i>Incremento del Producto</i> del sprint anterior, generando así el <i>Incremento del producto</i> para el sprint actual. Se recomienda que la organización utilice una rama de “desarrollo” para el <i>Incremento del Producto</i> .
Salidas	<i>Incremento del Producto</i>
A6. Realizar pruebas de integración.	
Entradas	<i>Incremento del Producto</i>
DT	El <i>Development Team</i> realiza las pruebas de integración, necesarias para garantizar que las nuevas funcionalidades son estables y además no afectan el funcionamiento de funcionalidades anteriores. Cuando existen errores, el flujo del proceso continúa con la actividad <i>Solucionar problemas de integración</i> , si las pruebas son satisfactorias el proceso continúa con la actividad <i>Realizar pruebas de caja blanca</i> .
Salidas	<i>Incremento del Producto</i>
A7. Solucionar problemas de integración.	
Entradas	<i>Incremento del Producto</i>

DT	El equipo de desarrollo identifica los errores provocados por la integración, se corrigen inmediatamente o se consulta la base del conocimiento para buscar solución a problemas conocidos. Si los problemas son desconocidos se debe documentar el nuevo error y su solución.
Salidas	<i>Incremento del Producto, Base del conocimiento</i>
A8. Realizar pruebas de caja blanca.	
Entradas	<i>Incremento del Producto, Recomendaciones o estándares</i>
DT	El <i>Tester</i> realiza pruebas de caja blanca, seguridad y rendimiento, tomando como referencia los documentos que la organización considere necesarios respecto a recomendaciones o estándares. Si se encuentran errores es necesario volver a la actividad <i>Desarrollar</i> donde el <i>Development Team</i> debe hacer los ajustes necesarios para corregir los errores encontrados y volver a recorrer el camino hasta esta actividad.
Salidas	<i>Resultados de pruebas</i>
A9. Realizar pruebas de usabilidad.	
Entradas	<i>Incremento del Producto, Recomendaciones o estándares</i>
DT	El Tester, una vez que aprueba el <i>Incremento del Producto</i> con respecto a calidad, seguridad y rendimiento del código, procede a realizar pruebas de usabilidad tomando como referencia los documentos que la organización considere necesarios sobre usabilidad. Si se encuentran errores de usabilidad, se debe volver a la actividad de <i>Desarrollar</i> en la que se llevan a cabo las correcciones necesarias y posteriormente se recorrer el camino hasta esta actividad.
Salidas	<i>Resultados de pruebas</i>
A10. Crear o actualizar artefactos técnicos.	
Entradas	
DT	Una vez que se tiene una versión del incremento del producto con todas las funcionalidades integradas que ha sido probada y aprobada. El equipo de desarrollo crea o actualiza los artefactos técnicos que se consideren necesarios.
Salidas	<i>Artefactos técnicos</i>
Flujo BPMN (El modelo del proceso se puede ver más grande en el Anexo B)	

Capítulo V. Estudio de caso

El *proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015* presentado en este proyecto fue adaptado y aplicado a una pequeña empresa de la región que estaba en busca de la mejora de sus procesos de desarrollo de software. Esta empresa estaba certificada en ISO 9001:2015 para uno de sus procesos, y querían incluir el proceso de desarrollo de software en el alcance del sistema de gestión de la calidad, utilizando a su vez el enfoque ágil SCRUM. Se llevó a cabo un estudio de caso, siguiendo el protocolo de la plantilla en [37]. Este capítulo describe el estudio de caso en términos de diseño, sujeto, procedimientos de campo y análisis de resultados.

5.1. Diseño

La principal pregunta de investigación que este estudio de caso buscaba resolver era: *¿Es el proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015 adaptable, útil y práctico para gestionar proyectos de software?* Otras preguntas de investigación adicionales fueron: (I) *¿el esfuerzo involucrado se adapta a pequeñas empresas?* (II) *¿el proceso propuesto realmente da a las pequeñas empresas la capacidad de aplicar agilidad a sus procesos de desarrollo a través de SCRUM?* Con estas preguntas se buscaba identificar si el proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015 funciona en la práctica y si se adapta a la realidad de las pequeñas empresas.

Las medidas utilizadas en este proyecto fueron: (I) El esfuerzo utilizado durante la adaptación del proceso propuesto a las necesidades de la organización y (II) la conformidad y percepción de los actores durante la aplicación del proceso a un proyecto piloto. También se tuvo en cuenta los beneficios descritos por el personal de la organización en general.

5.2. Sujetos y unidades de análisis

La organización en la que se realizó el estudio de caso comenzó una mejora de procesos de desarrollo de software partiendo del levantamiento de sus procesos actuales en la notación BPMN, y posteriormente realizando la adaptación y aplicación del *proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015* propuesto en este proyecto. La adaptación se llevó a cabo durante 4 meses con el acompañamiento de un *Tutor* asignado, quien es el autor de este documento, SCRUM Master certificado, y que posteriormente obtendría la certificación de auditor interno en ISO 9001:2015; además el *Tutor* estuvo bajo seguimiento y dirección permanente de un *Supervisor*, quien es el director de este proyecto, SCRUM Master certificado, auditor interno en ISO 9001:2008 e ISO

9001:2015, y además cuenta con una vasta experticia en armonización de múltiples modelos. Posterior a la adaptación y partiendo del proceso propuesto en este proyecto, la organización comenzó un proyecto de desarrollo aplicando los procedimientos definidos. Las unidades de análisis fueron la adaptación del proceso propuesto a las necesidades de la organización. Y la aplicación de los procesos definidos a un proyecto de desarrollo desde la captura de requisitos hasta su puesta en producción y mantenimiento.

Para preservar la anonimidad, de aquí en adelante se denominará a la organización objeto del estudio de caso como *La empresa*, la Tabla 23 describe algunas propiedades de la organización que participa en el estudio de caso.

Nombre	País	Empleados	Edad	Mercado	Áreas principales de actividad profesional
<i>La empresa</i>	Colombia	26(12) ²⁴	20 años	Nacional	Software de gestión documental electrónica de archivo

Tabla 23. Características de la organización objeto del estudio de caso. Fuente: [35]

El gerente de *La empresa* decidió optar por la mejora de sus procesos de desarrollo de software aplicando SCRUM e ISO 9001:2015 para dar apoyo sistemático a la consolidación y crecimiento de la organización, buscando incluir el proceso de desarrollo de software en el alcance del sistema de gestión de la calidad y consiguiendo así la capacidad de proporcionar regularmente productos y servicios software de alta calidad que satisfacen los requisitos del cliente, mejorando constantemente y abordando riesgos y oportunidades alineados con los objetivos de la organización.

El *Tutor* asignado estuvo encargado del acompañamiento a *La empresa*, durante la adaptación del proceso integrado en los procesos de software de la organización, reuniéndose a diario con personal específico de la misma, adicionalmente el *Tutor* se reunió semanalmente con el *Supervisor* para obtener retroalimentación con respecto a la correcta aplicación de los nuevos procesos.

5.3. Procedimiento de campo y recolección de datos

El procedimiento de campo y recolección de datos en este estudio de caso se llevó a cabo en dos grandes etapas: (I) la adaptación del proceso propuesto en este proyecto a las necesidades específicas de *La empresa*, en la cual se generaron procesos documentados para

²⁴ Total de empleados de la organización (empleados involucrados en desarrollo y mantenimiento de software).

las actividades de gestión de proyectos de desarrollo de software (se utilizó la notación BPMN para los modelos de procesos, ya que permitiría una definición estandarizada y mantenible en el tiempo) y (II) la aplicación de los procesos documentados a un proyecto de desarrollo de software en *La empresa*. En las siguientes secciones se presenta un resumen general del trabajo realizado durante las etapas de adaptación y aplicación.

5.3.1. Etapa de adaptación

Previo a la etapa de adaptación, fue necesario llevar a cabo un diagnóstico de los procesos de *La empresa* para evaluar el correcto uso de SCRUM en proyectos de desarrollo. Esta etapa generó como producto varios procesos en la notación BPMN que fueron necesarios para comenzar la adaptación, el diagnóstico de los procesos de la *La empresa* fue realizado por un ingeniero de sistemas con experiencia en modelado de procesos, también bajo la dirección del *Supervisor*.

La adaptación del proceso integrado propuesto en este proyecto a las necesidades de *La empresa* fue realizada durante un periodo de 4 meses, entre junio y octubre de 2016, a través de reuniones diarias de 4 horas entre el *Tutor* designado y personal específico de *La empresa*. Los insumos para la adaptación fueron los procesos levantados en la etapa de diagnóstico mencionada previamente.

Durante la primera semana de la etapa de adaptación, se llevó a cabo un entrenamiento en el enfoque ágil SCRUM para todo el personal de la organización a cargo del *Supervisor* y con el acompañamiento del *Tutor*, entre los participantes se encontraban miembros del equipo de desarrollo, líderes de calidad y la gerencia. Posterior al entrenamiento, el *Tutor* designado se reunió con personal específico de la organización para definir los nuevos procesos de desarrollo de software, de igual manera el *Tutor* designado se reunió también por separado semanalmente con el *Supervisor* para validar los nuevos procesos que se documentaban, como producto del ajuste de los procesos legados de *La empresa*. A continuación se describe el objetivo y el personal con el que se llevaron a cabo las reuniones por mes durante la etapa de adaptación.

5.3.1.1. Mes 1

Durante el primer mes, el *Tutor* designado se reunió principalmente con los líderes de desarrollo de *La empresa* para indagar sobre las prácticas que se realizaban hasta ese momento y tomando como referencia el proceso integrado propuesto en este proyecto para ajustar y definir las nuevas prácticas. Eventualmente asistieron a las reuniones personal de calidad, la gerencia y el *Supervisor* para retroalimentar los procesos que se estaban generando, obteniendo así las primeras versiones de los nuevos procesos para la gestión de proyectos software.

5.3.1.2. Mes 2

Durante el segundo mes, el *Tutor* designado se reunió inicialmente con personal de calidad, incluyendo una consultora en ISO 9001:2015 contratada por *La empresa* para obtener retroalimentación desde un punto de vista del estándar y para comenzar a documentar los nuevos procesos en un formato de especificación de procedimientos. Posteriormente, volvieron a las reuniones los líderes de desarrollo para ajustar los procesos y sus procedimientos con base en la retroalimentación de parte del personal de calidad, y para trabajar en los formatos requeridos por los nuevos procesos. Durante este segundo mes también hubo asistencia eventual del *Supervisor* para validar los nuevos procesos, procedimientos y formatos generados.

5.3.1.3. Mes 3

Durante el tercer mes, el *Tutor* designado se reunió con el personal de calidad para analizar y generar la documentación adyacente a los nuevos procesos y referente al sistema de gestión de la calidad como formatos, procedimientos y recomendaciones; además se realizaron reuniones para documentar los perfiles y responsabilidades requeridos por los nuevos procesos. Finalmente, también se realizaron reuniones con líderes de desarrollo, líderes de calidad y desarrolladores para validar y retroalimentar los nuevos procesos y su documentación.

5.3.1.4. Mes 4

Durante el cuarto y último mes, *La empresa* analizó la posibilidad de incluir el proceso de desarrollo de software en el alcance del sistema de gestión de la calidad para la auditoría de re certificación que se iba a realizar en el mes de septiembre, sin embargo, se tomó la decisión de no extender el alcance del sistema ya que la documentación de los nuevos procesos era muy reciente y podía estar sujeta a cambios o modificaciones. Durante el resto del mes se llevaron a cabo ajustes finales a los nuevos procesos y se publicó la documentación asociada, así mismo se llevaron a cabo actividades de socialización de los nuevos procesos al personal de *La empresa* incluyendo personal de desarrollo, calidad y la gerencia.

La Tabla 24 muestra la equivalencia entre el *proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015* propuesto en este proyecto y el conjunto de procesos definidos en *La empresa* producto de la etapa de adaptación, se puede observar que *La empresa* aplicó todos los elementos de proceso propuestos por este proyecto.

Elementos principales del proceso propuesto en este proyecto	Equivalencia en elementos definidos en procesos de <i>La empresa</i> estudio de caso
Roles y responsabilidades	Total
<i>Proceso de adquisición de requisitos de software</i>	Total
<i>Proceso de gestión de los productos y servicios suministrados externamente</i>	No Aplica
<i>Proceso de Diseño y Desarrollo</i>	Total
<i>Proceso de Planeación del Sprint</i>	Total
<i>Proceso de Ejecución del Sprint</i>	Total
<i>Proceso de Fabricación del Código</i>	Total

Tabla 24. Equivalencia entre el proceso integrado propuesto en este proyecto y el proceso definido por *La empresa* estudio de caso. Fuente: propia.

5.3.2. Etapa de aplicación

Debido a que *La empresa* para finales de 2016 no estaba desarrollando nuevo software sino manteniendo y dando soporte a sus principales productos, la aplicación de los nuevos procesos que surgieron de la etapa de adaptación se realizó entre marzo y julio de 2017 para un proyecto piloto que buscaba solucionar necesidades de atención y gestión de solicitudes de servicio de los clientes de *La empresa*.

Se buscó aplicar los procesos definidos en la etapa de adaptación, sin embargo por disponibilidad de recursos no fue posible aplicar todos los elementos de los nuevos procesos de desarrollo ya que el personal designado para llevar a cabo el proyecto piloto fueron 1 *Product Owner*, 1 *SCRUM Master/Desarrollador* y 2 interesados. La Tabla 25 muestra la equivalencia entre el proceso definido en la etapa de adaptación en 5.3.1 y el proceso aplicado para la gestión del proyecto software.

Elementos principales del proceso definido en el ítem 5.3.1	Equivalencia en elementos aplicados para la gestión del proyecto piloto
Roles y responsabilidades	Parcial, ya que los roles de SCRUM Master y Equipo de Desarrollo fueron compartidos por una sola persona, dejando falencias en miembros del equipo de desarrollo esenciales del proceso como personal especializado en pruebas.
<i>Proceso de adquisición de requisitos de software</i>	Total
<i>Proceso de gestión de los productos y servicios suministrados externamente</i>	No Aplica
<i>Proceso de Diseño y Desarrollo</i>	Parcial, ya que al ser el Equipo de Desarrollo de una sola persona, no hubo diferentes puntos de vista en cuanto a planeación e implementación de arquitectura y diseño del software.
<i>Proceso de Planeación del Sprint</i>	
<i>Proceso de Ejecución del Sprint</i>	
<i>Proceso de Fabricación del Código</i>	

Tabla 25. Equivalencia entre el proceso definido por *La empresa* y el proceso aplicado en el proyecto piloto. Fuente: propia.

A continuación se describen de manera general algunas de las actividades realizadas durante la aplicación de los nuevos procesos.

5.3.2.1. Inicio del proyecto

El proyecto software comenzó a mediados del mes de marzo de 2017, de acuerdo a los procesos definidos en 5.3.1 se siguió inicialmente el proceso definido por la organización para la captura de requisitos de software que consistía de reuniones entre los 4 miembros del equipo, siendo los interesados dos usuarios finales de la aplicación que explicaron en detalle las necesidades y requisitos para el producto. La definición de los requisitos de software terminó a inicios del mes de abril con un total de 20 requisitos. Posteriormente el *SCRUM Master* con la ayuda del *Product Owner* creó las historias de usuario en la herramienta de gestión de la organización generando así el *Product Backlog de alto nivel* con un total de 27 historias de usuario.

5.3.2.2. Ejecución del proyecto

Las planeaciones del sprint no fueron realizadas ya que el equipo de desarrollo era de una sola persona y el desarrollador no tenía dudas de implementación. Se llevaron a cabo 4 sprints entre los meses de abril y julio de 2017, cada sprint con su respectiva reunión de

Sprint Review entre el *Product Owner* y el *SCRUM Master*, sin embargo no se realizaron retrospectivas ya que el *SCRUM Master* y el *Development Team* eran la misma persona. Durante el sprint 1 comprendido entre el 4 y 21 de abril de 2017, Se implementaron 8 historias de usuario, el *Product Owner* aceptó todas las historias durante la reunión de *Sprint Review* haciendo comentarios y sugerencias menores. Durante el sprint 2 comprendido entre el 26 de abril y 8 de mayo se implementaron 4 historias de usuario, el *Product Owner* aceptó todas las historias de usuario durante la reunión de *Sprint Review* igual que antes haciendo comentarios y sugerencias menores. El sprint 3 se realizó durante el 19 de mayo y el 5 de junio, se implementaron 6 historias de usuario que fueron aceptadas en su totalidad por el *Product Owner* durante la reunión de *Sprint Review*. Finalmente, el sprint 4 se realizó entre el 15 de junio y el 4 de julio donde se implementaron 9 historias de usuario que fueron aceptadas por el *Product Owner* durante la reunión de *Sprint Review*. A continuación, en la Figura 8 se muestra el reporte del sprint 4 en la herramienta de gestión utilizada para el proyecto piloto.

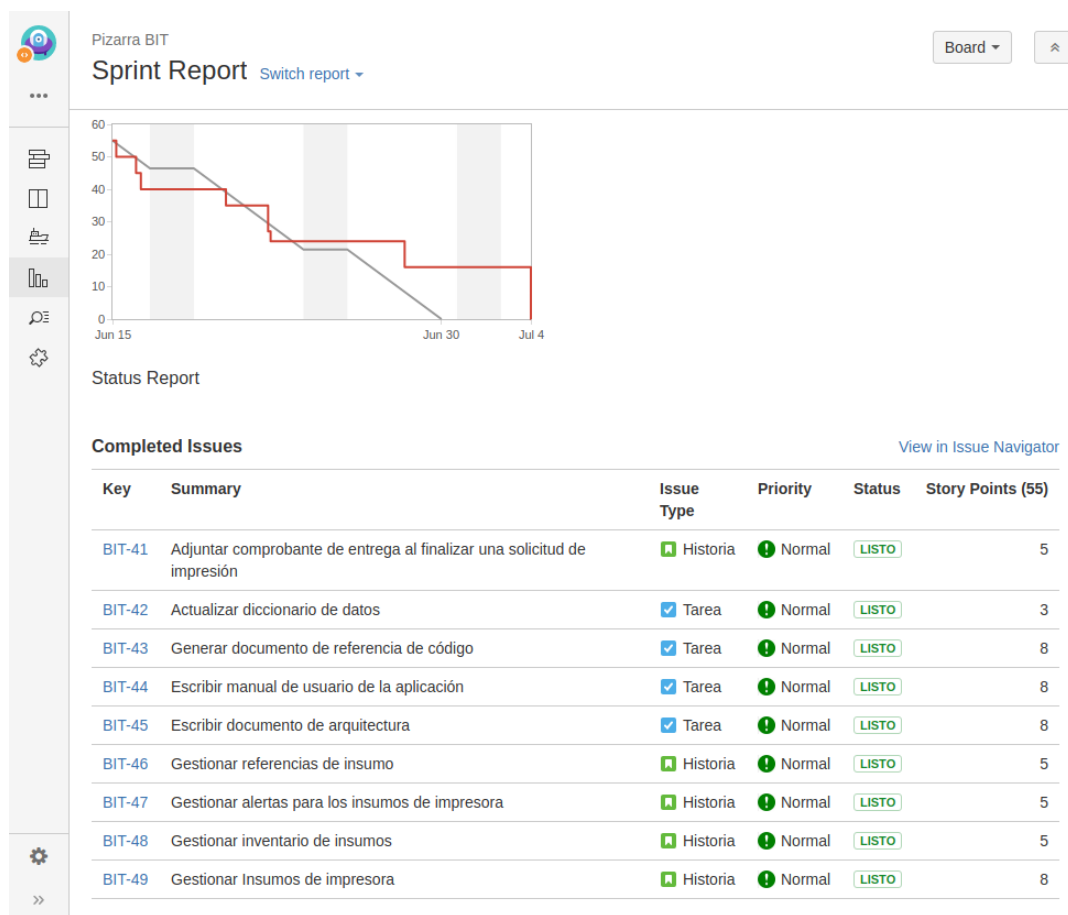


Figura 8. Captura de pantalla del reporte del sprint 4 en la herramienta de gestión. Fuente: propia.

5.3.2.3. Puesta en marcha

El producto fue puesto en producción a comienzos del mes de julio de 2017 para 8 clientes, entre ellos 2 internos (usuarios finales empleados de *La empresa*) y 6 externos (usuarios finales clientes de *La empresa*). Hasta noviembre de 2018 y después de 1 año y 4 meses en producción, se han presentado tan solo 3 solicitudes de soporte relacionadas con correcciones de datos que se ingresaron mal.

5.4. Análisis de resultados y lecciones aprendidas

5.4.1. Esfuerzo

Con respecto al esfuerzo en horas/hombre realizado por *La empresa* en la etapa de adaptación, se programaron 84 días de reuniones de los cuales 12 no se realizaron debido a falta de tiempo del personal o falta de actividades pendientes, sin embargo al obtener un promedio de asistentes en todas las 84 reuniones programadas se obtuvo un total de 3.04 asistentes incluyendo al tutor designado, de esta manera y teniendo en cuenta la intensidad horaria (4 horas por día) se obtuvo un esfuerzo total aproximado de 1021 horas/hombre para la etapa de adaptación.

La Figura 9 muestra el esfuerzo en horas/hombre realizado por personal específico de la organización durante los meses de julio, agosto, septiembre y octubre; en los cuales se llevó a cabo la etapa de adaptación, incluyendo los roles de *Supervisor* y *Tutor*. Se puede observar un mayor esfuerzo por parte del equipo de desarrollo, los líderes de desarrollo y el *Tutor*, mientras que el menor esfuerzo fue realizado por el *Supervisor* y la gerencia.

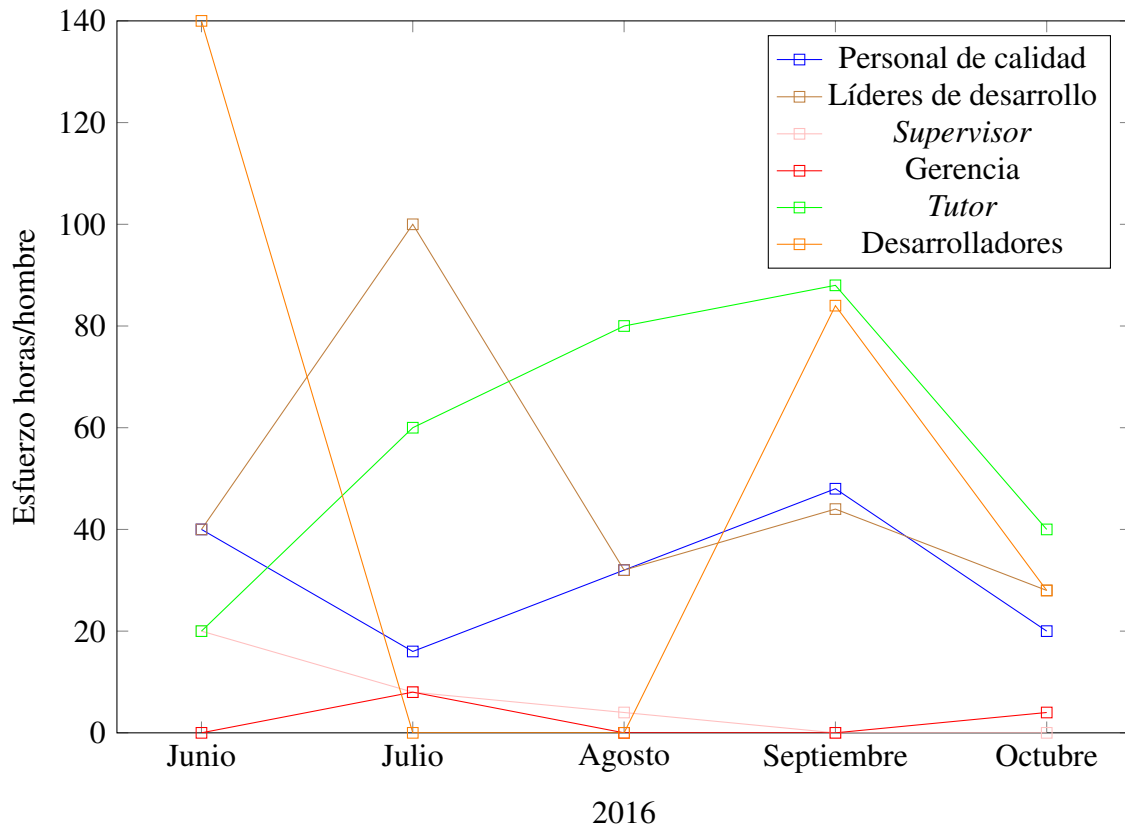


Figura 9. Esfuerzo de *La empresa* durante la etapa de adaptación. Fuente: propia.

5.4.2. Encuesta de conformidad con el proceso

Para evaluar la percepción y conformidad del personal que participó en el proyecto piloto con respecto al proceso aplicado, se llevó a cabo una encuesta de evaluación en la que se preguntó por el nivel de conformidad con respecto a un conjunto de premisas, para expresar la conformidad se utilizó la escala de *Likert* como se muestra a continuación:

Escala	Descripción
5	Muy de acuerdo / Muy bien
4	De acuerdo / Bien
3	Indeciso / Regular
2	En desacuerdo / Mal
1	Muy en desacuerdo / Muy mal

Tabla 26. Escala de *Likert* para la encuesta de evaluación. Fuente: propia.

A continuación, la Tabla 27 presenta las premisas de la encuesta de evaluación que buscaban responder a las preguntas descritas en el diseño del estudio de caso (ver ítem 5.1)

N°	Premisa
1	El proceso aplicado en su organización contiene todos los elementos necesarios para gestionar un proyecto de software de manera ágil y generando un producto de calidad.
2	Las actividades y subprocesos son de fácil comprensión y aplicación para la gestión de proyectos ágiles.
3	Las actividades de cada sub proceso permiten cumplir los objetivos del mismo.
4	Las actividades del subproceso de planeación de un sprint son suficientes para obtener un Sprint Backlog adecuado.
5	Las actividades del subproceso de ejecución de un sprint son suficientes para obtener un incremento del producto adecuado.
6	Las actividades del subproceso para la fabricación del código son suficientes para codificar de manera organizada y controlada.
7	Las evidencias generadas por el proceso son suficientes para dar soporte a los requisitos de ISO 9001:2015.
8	La documentación generada por el proceso es suficiente para cumplir con el estándar, pero no demasiada como para afectar negativamente la agilidad.
9	Las actividades del proceso permiten la mejora continua.
10	Las actividades del proceso permiten construir un producto que satisface los deseos y necesidades del cliente y sus interesados.
11	El proceso aplicado en su organización está al alcance de las MiPyMEs en cuanto a esfuerzo tanto en horas/hombre como económico.

Tabla 27. Premisas de la encuesta de evaluación del proceso. Fuente: propia.

La encuesta de evaluación se realizó a los 4 participantes del proyecto piloto donde se aplicó el proceso definido en la etapa de adaptación, correspondientes al *Product Owner*, el *SCRUM Master/Desarrollador* y los 2 interesados. Los resultados de la encuesta se evidencian en la Figura 10, de la que se puede decir que en su mayoría los participantes del proceso quedaron muy conformes con el proceso en general, es de notar la premisa número **8**, *La documentación generada por el proceso es suficiente para cumplir con el estándar, pero no demasiada como para afectar negativamente la agilidad*, ya que es la única que presenta una respuesta de un nivel 3 de conformidad correspondiente a *Indeciso/Regular* y la única que no tiene ninguna respuesta de nivel 5 de conformidad correspondiente a *Muy de acuerdo/Muy bien*, en comparación con las otras premisas.

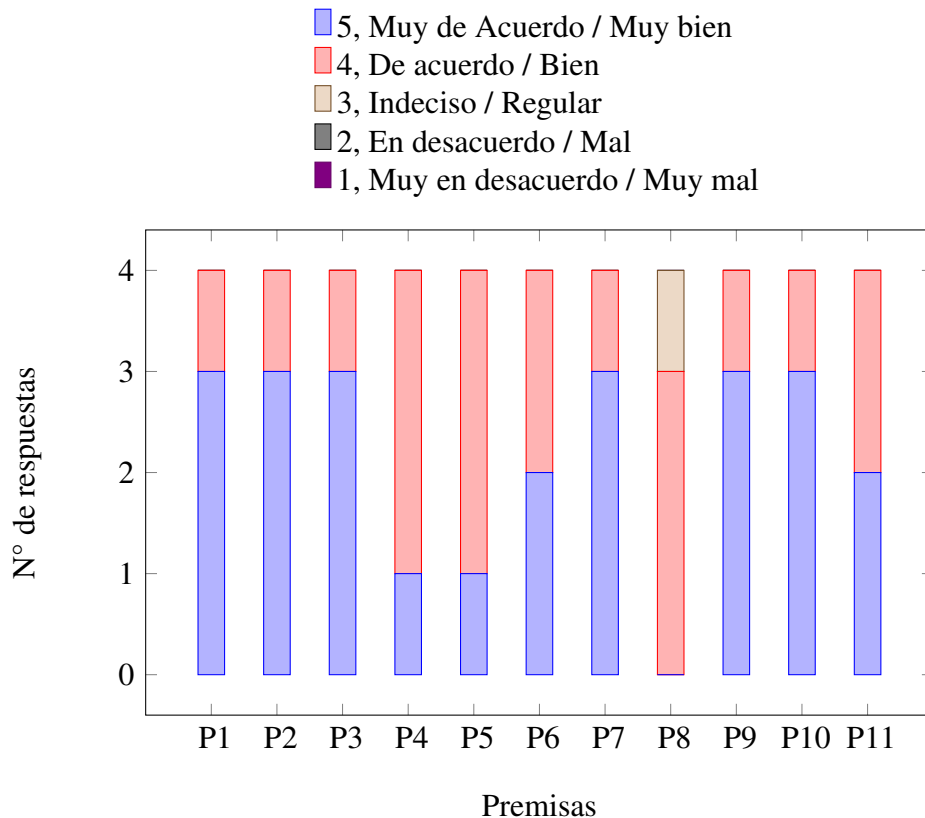


Figura 10. Resultados de la encuesta de evaluación del proceso. Fuente: propia.

5.4.3. Factores de éxito y fracaso identificados

Además de las etapas de adaptación y aplicación, La Tabla 28 muestra los factores de éxito y fracaso identificados por el *Tutor*, *Supervisor* y personal de *La empresa* durante el estudio de caso, se puede observar que el *Factor 3* correspondiente a la inclusión de un consultor externo se identificó durante la etapa de adaptación, sin embargo no se mantuvo para la etapa de aplicación; de manera análoga el *Factor 4* se identificó en la etapa de adaptación puesto que en los procesos documentados se definieron actividades de pruebas de software en todos sus niveles, sin embargo en la etapa de aplicación debido que el equipo de desarrollo era una sola persona, la falta de personal de pruebas no permitió aplicar las actividades definidas; se excluye el Factor 8 por motivos de confidencialidad, para ver en mayor detalle los factores de éxito y fracaso a tener en cuenta para cualquier organización, ver la Tabla 7.

N° Factor	Descripción	¿Identificado en Adaptación?	¿Identificado en Aplicación?
1	Los procesos, sus actividades y artefactos definidos para la organización deben existir porque contribuyen a la mejora de la calidad y no únicamente para cumplir con el estándar.	Si	Si
2	El uso de herramientas informáticas, tanto web, desktop o móviles permiten gestionar mejor, tanto a los procesos como a las personas, además que ayudan a mantener la trazabilidad y los registros de una forma más fácil sin olvidar que ahorran papel.	Si	Si
3	Un consultor externo es necesario, tener la opinión de un tercero experto permite saber si las cosas se están haciendo bien desde la perspectiva de ISO 9001.	Si	No
4	Involucrar las pruebas en todas las partes del ciclo de vida del desarrollo de software donde sea posible, el objetivo es la calidad y asegurarse que el producto correcto es aprobado y el incorrecto es rechazado es fundamental para el estándar ISO 9001.	Si	No
5	El trabajo en equipo es esencial, mantener un ambiente de trabajo positivo con personal motivado conduce a una mayor sinergia en la organización.	Si	Si
6	Utilizar métricas en los diferentes procesos es necesario para identificar oportunidades de mejora y la toma de decisiones al respecto.	Si	Si
7	La documentación debe ser reducida en orden de mantener la agilidad, pero no puede ser suprimida.	Si	Si
9	La falta de disciplina es el principal factor que lleva al fracaso, la agilidad no significa anarquía o indisciplina en ninguna manera.	No	No

10	La falta de compromiso de todos los involucrados, incluyendo la dirección, el personal de la organización, los clientes y los interesados es una potencial causa de fracaso.	No	No
11	La falta de documentación, es una causa de fracaso para la implementación de un sistema de gestión de la calidad conforme a ISO 9001 o estándares similares.	No	No

Tabla 28. Factores de éxito y fracaso en el estudio de caso durante las etapas de Adaptación y Aplicación. Fuente: propia.

5.4.4. Beneficios y lecciones aprendidas

La empresa contaba con 12 empleados involucrados en actividades de desarrollo o mantenimiento de productos software, sin embargo debido a que la gerencia participó en el proceso de adaptación, se tiene en cuenta para el cálculo del esfuerzo. Por lo tanto en un rango de 84 días con una intensidad laboral de 8 horas al día, se estima un aproximado de 8736 horas/hombre, de las cuales 1021 fueron empleadas para adaptar la propuesta de este proyecto a los procesos existentes en *La empresa*, es decir que durante los 84 días laborales se invirtió el 11.69 % del esfuerzo.

Dos de los factores de éxito y fracaso identificados fueron el *Factor 5* correspondiente al trabajo en equipo, y el *Factor 10* correspondiente al compromiso de todos los involucrados, puesto que todos estuvieron siempre dispuestos a contribuir con la mejora de los procesos, tanto desde la perspectiva de los desarrolladores como la del personal de calidad. Esto fue clave a ya que el progreso de la adaptación no tuvo mayores impedimentos y contribuyó a una reducción del esfuerzo estimado.

Con respecto a la etapa aplicación del proceso no es posible generalizar el esfuerzo ya que como se puede observar en la Tabla 25 el Equipo de Desarrollo era de una sola persona, y el esfuerzo empleado por los participantes depende en gran medida del *Factor 8* correspondiente a la experticia o nivel de conocimiento (ver Tabla 7). Sin embargo aún con las limitaciones de recursos, el proyecto fue culminado satisfactoriamente dando como resultado un producto que satisface los requisitos del cliente y sus interesados.

De acuerdo a los resultados de la encuesta de conformidad (ver Figura 10), se puede decir que el 100% de quienes participaron en la aplicación del proceso a un proyecto piloto, están “Muy de Acuerdo” con que el proceso aplicado contiene todos los elementos necesarios para gestionar un proyecto de desarrollo de software de manera ágil y generando un producto de calidad. De igual manera el 100% de los participantes están “Muy de Acuerdo” con que las documentación generada por el proceso permite dar soporte a los requisitos de

ISO 9001:2015.

Finalmente, es que de notar que el 75 % de quienes participaron en la aplicación del proceso están “De acuerdo” con que la documentación generada por el proceso es suficiente para cumplir con el estándar pero no demasiada como para afectar la agilidad, mientras que el 25 % restante está indeciso. Esto concuerda con lo que se dice en [27], referente a que los enfoques ágiles y tradicionales son compatibles, pero no combinables de tal forma que al integrarlos, uno de los dos enfoques o los dos deben ser extendidos o acotados, reflejándose en una disminución de agilidad o rigurosidad en enfoques ágiles o tradicionales respectivamente.

5.5. Plan de validación

Puesto que la armonización fue realizada por personas certificadas como SCRUM Master, tanto el *Tutor* como el *Supervisor*. Puede haber un sesgo en las relaciones identificadas durante la armonización. Sin embargo se tomaron medidas para reducir este riesgo ya que el *Supervisor* también estaba certificado como auditor interno de ISO 9001:2008, y posteriormente tanto el *Supervisor* como el *Tutor* obtuvieron la certificación de auditor interno de ISO 9001:2015. Adicionalmente, se contó con la validación de una Ingeniera externa experta en sistemas de gestión de la calidad con ISO 9000 y con el codirector de este proyecto, experto en la armonización de múltiples modelos.

Las relaciones de la armonización se soportaron con el estandar ISO 90003:2014, para validar el cumplimiento de los requisitos en la Industria del desarrollo de software.

5.6. Limitaciones

Debido a que el estudio de caso se realizó en una sola organización, es necesario que se lleven a cabo más evaluaciones para poder obtener resultados objetivos puesto que debido al tiempo compartido entre el personal de la organización, el tutor y el supervisor; puede haber un sesgo a la hora de evaluar el proceso.

Como se puede observar en la tabla 25, hubo limitaciones de recursos humanos para aplicar el proceso definido en su máxima extensión, esto impidió realizar todas las actividades propuestas y aunque no se vio reflejado en la calidad del producto final, no permite llevar a cabo una auditoría de ISO 9001:2015, o una evaluación del cumplimiento de SCRUM.

Capítulo VI. Conclusiones y trabajos futuros

Este capítulo presenta un análisis de los objetivos de investigación y la forma en que se cumplieron, además de las conclusiones obtenidas y los posibles trabajos futuros que surgieron durante la realización de este proyecto.

6.1. Análisis de objetivos de investigación

Para la realización de este proyecto se definieron un conjunto de objetivos (ver ítem 1.2) que fueron cumplidos sistemáticamente siguiendo las actividades definidas en el ítem 1.3. A continuación se presenta un resumen del cumplimiento de cada uno de los objetivos propuestos.

- OE1.** *Realizar una revisión sistemática de la literatura acerca de la integración de prácticas ágiles y el estándar ISO 9001, lo cual permitirá identificar las propuestas, soluciones y herramientas definidas por otros autores.*

El Capítulo II. Marco teórico y estado del arte en su ítem 2.2 y específicamente el **Anexo D** describe en detalle los elementos, la planeación, ejecución y análisis de resultados de la revisión sistemática de la literatura acerca de la integración del estándar ISO 9001 y enfoques ágiles. Con la cual fue posible la obtención de los estudios considerados como primarios para este proyecto y el análisis de los mismos identificando enfoques analizados, técnicas de armonización utilizadas, factores de éxito y fracaso, propuestas y estudios de caso realizados en la integración de enfoques tradicionales y enfoques ágiles, entre otros hallazgos.

- OE2.** *Identificar y definir las relaciones entre SCRUM y el capítulo 8. Operación del estándar ISO 9001:2015 a través de la comparación a bajo nivel de abstracción utilizando como interfaz el estándar ISO 90003:2014 y las técnicas de armonización de modelos propuestas en [35].*

El Capítulo III. Armonización describe en detalle las actividades realizadas para armonizar el estándar ISO 9001:2015 en sus capítulos 8. *Operación*, 9. *Evaluación del desempeño* y 10. *Mejora* con el enfoque ágil SCRUM. Utilizando las técnicas de armonización en [35] para homogeneizar los elementos de proceso de múltiples enfoques y realizando una comparación de bajo nivel a través del método de *mapping* propuesto en [50]. Finalmente, fue posible identificar la conformidad de SCRUM C_s ²⁵ con respecto a cada uno de los capítulos basándose en las relaciones identificadas entre los requisitos del estándar, con las actividades, roles y artefactos de SCRUM.

²⁵Conformidad de SCRUM

- OE3.** *Definir un proceso integrado para la gestión de proyectos ágiles que permita a las micro, pequeñas y medianas empresas desarrollar productos y servicios de software a través de un enfoque ágil y conforme a los requisitos de la norma ISO 9001:2015.*

El Capítulo IV. Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015 presenta el conjunto de procesos definidos que permiten gestionar proyectos ágiles de desarrollo de software dando soporte a los requisitos del estándar ISO 9001:2015 en MiPyMEs. Partiendo del macroproceso **Diseño y Desarrollo** que abarca los subprocesos: **Planeación del sprint** para satisfacer los requisitos del *Diseño* durante la *Operación*, **Ejecución del sprint** para satisfacer los requisitos del *Desarrollo* durante la *Operación*, y también incluyendo actividades de SCRUM como *Sprint Review* y *Scrum Retrospective* para satisfacer requisitos de *Evaluación del desempeño* y *Mejora*, entre otros.

- OE4.** *Generar una guía electrónica en BPMN utilizando la herramienta Bizagi Modeler que permita conocer los elementos del Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015 y esté disponible vía web.*

Para la facilidad de visualización del proceso propuesto en este proyecto, incluyendo los modelos de proceso en la notación BPMN, sus descripciones en la plantilla de COMPETISOFT[53], y la homogeneización de los enfoques SCRUM e ISO 9001:2015, se desarrolló una guía electrónica con una interfaz amigable al lector, disponible online en <http://artemisa.unicauca.edu.co/~miguelmorcillo/public>

- OE5.** *Evaluar la propuesta llevando a cabo un estudio de caso en una empresa de software de la región.*

El Capítulo V. Estudio de caso describe los elementos, etapas e indicadores del estudio de caso realizado en una PyME de la región.

6.2. Conclusiones

La revisión sistemática llevada a cabo en este proyecto permitió identificar las propuestas existentes con respecto a la integración del estándar ISO 9001 y enfoques ágiles, del análisis de las mismas se obtuvieron recursos fundamentales para la propuesta presentada como lo fueron: (I) 14 factores de éxito y fracaso, (II) la importancia de las pruebas de software para satisfacer requisitos de calidad y, (III) el hecho de que enfoques tradicionales y ágiles pueden ser complementarios a pesar que en primera instancia parecen contradictorios. Entre otros. Además, se pudo evidenciar que existía una falencia en la literatura con respecto a la armonización de Scrum e ISO 9001 que llevara a cabo una comparación o mapping de bajo nivel de los elementos de proceso.

Se llevó a cabo una armonización de modelos utilizando las técnicas definidas en [35],

partiendo de una homogeneización que permitió representar la información de los enfoques de tal manera que podía ser comparada, se llevó a cabo un mapping entre los elementos de proceso de los enfoques homogeneizados, cuyo resultado dejó en evidencia que SCRUM puede ser una forma de implementar ISO 9001:2015 en la gestión de proyectos de software, finalmente la armonización permitió definir un conjunto de procesos genéricos que dan soporte a la gestión de proyectos de desarrollo de software con los principios ágiles de SCRUM junto a la disciplina y rigurosidad del estándar ISO 9001:2015.

La evaluación a través de un estudio de caso permitió identificar el esfuerzo necesario requerido por una organización para adaptar el proceso propuesto a sus necesidades, específicamente la organización objeto del estudio utilizó aproximadamente un 11.7% del esfuerzo disponible durante 4 meses de adaptación, siendo un valor aceptable para MiPyMEs. Además se obtuvo un resultado satisfactorio en la aplicación del mismo a un entorno real con un proyecto de desarrollo de software, aún cuando por limitaciones de recursos no fue posible aplicar el proceso en su máxima extensión.

Del estudio de caso también se puede decir que el personal que participó en la aplicación del proceso quedó muy conforme con los resultados, y están de acuerdo en que el proceso permite gestionar proyectos ágiles de desarrollo de software de manera ágil, útil y práctica.

6.3. Trabajos futuros

Durante la realización de este proyecto fue posible identificar una serie de vías de trabajo futuro descritas a continuación:

Como se menciona en los factores de éxito y fracaso (ver ítem 7), una herramienta web, o móvil que permita gestionar los procesos es un factor clave para lograr los objetivos a la hora de integrar enfoques ágiles estándares de calidad. Asimismo, los requisitos definidos en [54] proveen un punto de partida para el desarrollo de tal herramienta.

Como se mencionó en las limitaciones del estudio de caso (ver ítem 5.6) El *Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015* debe ser validado en más estudios de caso para garantizar que los indicadores pueden ser generalizados.

Asimismo, los proyectos de desarrollo de software que sean gestionados utilizando el proceso propuesto deben ser auditados con respecto al estándar ISO 9001:2015, esto para garantizar que en efecto el *Proceso integrado para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015* ofrece conformidad con el estándar.

Finalmente y de manera análoga, los proyectos de desarrollo de software que sean gestionados utilizando el proceso propuesto deben ser evaluados con respecto a la implementación de SCRUM, esto permitirá garantizar que en efecto el *Proceso integrado*

para la gestión de proyectos ágiles basado en SCRUM e ISO 9001:2015 mantiene la agilidad y los beneficios de SCRUM.

Referencias

- [1] C. L. Tapia Urbina, “PSP/TSP, SCRUM e ISO 9126 y 90003 en combinación como herramientas de trabajo para la construcción en pequeños y medianos proyectos de software”, Tesis doct., Universidad Autónoma de Querétaro, 2009.
- [2] S. Galvan, M. Mora, R. V. O’Connor, F. Acosta y F. Alvarez, “A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process”, en *Procedia Computer Science*, vol. 64, Elsevier Masson SAS, 2015, págs. 188-195, ISBN: 18770509 (ISSN). DOI: 10.1016/j.procs.2015.08.480. dirección: <http://dx.doi.org/10.1016/j.procs.2015.08.480>.
- [3] ISO/TC 176/SC 2 Quality Systems, “ISO 9001:2015 Quality management systems – Requirements”, vol. 5, 2015.
- [4] K. Petersen y C. Wohlin, “The effect of moving from a plan-driven to an incremental software development approach with agile practices: An industrial case study”, *Empirical Software Engineering*, vol. 15, n.º 6, págs. 654-693, 2010, ISSN: 13823256. DOI: 10.1007/s10664-010-9136-6.
- [5] IEEE, *Iso/Iec 12207*, 2008. dirección: <https://www.iso.org/standard/43447.html>.
- [6] I. J. S. 7, *Iso/Iec 15504*, 2013. dirección: <https://www.iso.org/standard/61492.html>.
- [7] SEI, *CMMI*. dirección: <https://www.sei.cmu.edu/cmmi/>.
- [8] R Woodward, *The Organisation for Economic Co-operation and Development (OECD)*, ép. Global Institutions. Taylor & Francis, 2009, ISBN: 9780203875773. dirección: <https://books.google.com.co/books?id=I851I1hsMfAC>.
- [9] A. F. Bustamante, J. A. Hincapié y G. P. Gasca-Hurtado, “Structure of a multi-model catalog for software projects management including agile and traditional practices”, en *Advances in Intelligent Systems and Computing*, vol. 405, 2016, págs. 87-97, ISBN: 9783319262833. DOI: 10.1007/978-3-319-26285-7_8.
- [10] M. Qasaimeh y A. Abran, “Investigation of the capability of XP to support the requirements of ISO 9001 software process certification”, en *8th ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2010*, 2010, págs. 239-247, ISBN: 9780769540757. DOI: 10.1109/SERA.2010.38.
- [11] R. V. O. Connor y C. Y. Laporte, “Towards the Provision of Assistance for Very Small Entities in Deploying Software Lifecycle Standards”, en *ACM International Conference Proceeding Series*, ACM, 2010, págs. 4-7, ISBN: 9781450302814. DOI: 10.1145/1961258.1961259.

- [12] K. Schwaber y J. Sutherland, *The Definitive Guide to Scrum: The Rules of the Game*, Available at <http://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>, 2017.
- [13] M. Poppendieck y T. Poppendieck, *Lean Software Development: An Agile Toolkit*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2003, ISBN: 0321150783.
- [14] S. Amber, *Agile Unified Process*. dirección: <https://goo.gl/Pp2Yih>.
- [15] *KanBan Fundamentals*. dirección: <http://www.scrumhub.com/kanban-fundamentals/>.
- [16] B. Roberts, *Dynamic Systems Development Method*, 2001. dirección: <https://www.agilebusiness.org/what-is-dsdm>.
- [17] A. Cockburn, *Crystal Clear a Human-powered Methodology for Small Teams*, First. Addison-Wesley Professional, 2004, ISBN: 0201699478.
- [18] *Refactoring Extreme Programming*, 2005. DOI: 10.1002/stvr.315. dirección: <http://www.extremeprogramming.org/>.
- [19] J. A. Highsmith III, *Adaptive Cockburn Software Development: A Collaborative Approach to Managing Complex Systems*. New York, NY, USA: Dorset House Publishing Co., Inc., 2000, ISBN: 0-932633-40-4.
- [20] C. Larman, *Agile and Iterative Development: A Manager's Guide*. Pearson Education, 2003, ISBN: 0131111558.
- [21] B. Boehm y R. Turner, "Balancing agility and discipline: evaluating and integrating agile and plan-driven methods", *Proceedings. 26th International Conference on Software Engineering*, págs. 2-3, 2004, ISSN: 0270-5257. DOI: 10.1109/ICSE.2004.1317503. dirección: <https://goo.gl/jm1sza>.
- [22] T. Popović, "Getting ISO 9001 certified for software development using scrum and open source tools: a case study", *Tehnicki vjesnik - Technical Gazette*, vol. 22, n.º 6, págs. 1633-1640, 2015, ISSN: 13303651. DOI: 10.17559/TV-20140704180948. dirección: http://hrcak.srce.hr/index.php?show=clanak{\&}id{_}clanak{_}jezik=219986.
- [23] J. Nawrocki, M. Jasiński, B. Walter y A. Wojciechowski, "Combining extreme programming with ISO 9000", *EurAsia-ICT 2002: Information and Communication Technology*, vol. 23, págs. 786-794, 2002, ISSN: 16113349. DOI: 10.1007/3-540-36087-5_91. dirección: <http://www.springerlink.com/index/3F3AKBRVP9NGKLVQ.pdf>.

- [24] B. McMichael y M. Lombardi, "ISO 9001 and agile development", en *Proceedings - AGILE 2007*, 2007, págs. 262-265, ISBN: 0769528724. DOI: 10 . 1109 / AGILE . 2007 . 36. dirección: <https://www.scopus.com/inward/record.uri?eid=2-s2.0-46449138772{\&}partnerID=40{\&}md5=199a8facc31c53205d621995d046be38>.
- [25] J. G.A.O. B. Da Silva y P. R. Da Cunha, "Reconciling the irreconcilable? A software development approach that combines Agile with Formal", en *Proceedings of the Annual Hawaii International Conference on System Sciences*, vol. 9, 2006, ISBN: 0769525075. DOI: 10 . 1109/HICSS.2006.408.
- [26] D. G. Kourie y B. W. Watson, "Standards and Agile Software Development", en *Saicsit 2003*, 2003, págs. 178-188.
- [27] L. Heeager, "The Agile and the Disciplined Software Approaches: Combinable or Just Compatible?", en *Information Systems Development SE - 4*, 2013, págs. 35-49, ISBN: 978-1-4614-4950-8. DOI: 10 . 1007 / 978 - 1 - 4614 - 4951 - 5 _ 4. dirección: http://dx.doi.org/10.1007/978-1-4614-4951-5{_}4.
- [28] T. Stålhane y G. K. Hanssen, "The application of ISO 9001 to agile software development", en *Product-Focused Software Process Improvement*, 2008, págs. 371-385, ISBN: 978-3-540-69564-6. DOI: 10 . 1007/978-3-540-69566-0_30.
- [29] L. T. Heeager, "How Can Agile and Documentation-Driven Methods be Meshed in Practice ?", en *Agile Processes in Software Engineering and Extreme Programming*, 2014, págs. 62-77.
- [30] G. Wright, "Achieving ISO 9001 Certification for an XP Company", *Extreme Programming and Agile Methods-XP/Agile Universe 2003*, vol. 2753, págs. 43-50, 2003, ISSN: 03029743. DOI: 10 . 1007 / 978 - 3 - 540 - 45122 - 8 _ 6. dirección: <http://www.springerlink.com/index/VPL6VM9HT9FFNME3.pdf>.
- [31] ISO, "ISO Survey of Certifications 2014.", inf. téc., 2015, págs. 1-2.
- [32] C. Vriens, "Certifying for CMM Level 2 and ISO9001 with XP@Scrum", *Proceedings of the Agile Development Conference, ADC 2003*, págs. 120-124, 2003. DOI: 10 . 1109/ADC.2003.1231461.
- [33] O. Murru, R. Deias y G. Mugheddu, "Assessing XP at a European Internet company", *IEEE Software*, vol. 20, n.º 3, págs. 37-43, 2003, ISSN: 07407459. DOI: 10 . 1109/MS.2003.1196318.
- [34] E. Irrazabal, F. Vásquez, R. Díaz y J. Garzás, "Applying ISO / IEC 12207 : 2008 with SCRUM and Agile Methods", *Communications in Computer and Information Science*, vol. 155 CCIS, págs. 169-180, 2011.
- [35] P. César, "A Framework to Support the Harmonization between Multiple Models and Standards", Tesis doct., Institute of Information Technologies & Systems, University of Castilla-La Mancha, 2012.

- [36] T Wood-Harper, “Research Methods in Information Systems: Using Action Research”, en *Information Systems*, Amsterdam, 1985, págs. 169-191.
- [37] P. Brereton, B. Kitchenham, D. Budgen y Z. Li, “Using a Protocol Template for Case Study Planning”, en *Proceedings of the 12th International Conference on Evaluation and Assessment in Software Engineering*, ép. EASE’08, Italy: BCS Learning & Development Ltd., 2008, págs. 41-48. dirección: <https://goo.gl/n4BzqZ>.
- [38] ISO/IEC JTC 1/SC 7, “ISO/IEC 90003:2014 Software engineering – Guidelines for the application of ISO 9001:2008 to computer software”, vol. 3, 2014.
- [39] Icetex, *TERCERA CONVOCATORIA DE COMPETENCIAS TRANSVERSALES*, Disponible en internet, 2015. dirección: <https://goo.gl/FUuh6f>.
- [40] Fedesoft, *Caracterización del sector Teleinformática, Software y TI en Colombia 2015*, 2015. dirección: <http://fedesoft.org/noticias-fedesoft/disponible-estudio-de-caracterizacion-de-la-industria-del-software-colombiano>.
- [41] Unimedios, *Promedio de vida de microempresas en Latinoamérica*, Disponible en <http://agenciadenoticias.unal.edu.co/detalle/article/promedio-de-vida-de-microempresas-en-latinoamerica-es-de-12-anos.html>, 2014.
- [42] A. Alliance, *What is Agile?*, Available at <https://www.agilealliance.org/agile101>, 2017.
- [43] V. One, *11th annual State Of Agile Report*, 2017. dirección: <https://explore.versionone.com/state-of-agile/versionone-11th-annual-state-of-agile-report-2>.
- [44] G. Coleman y R. O’Connor, “Investigating software process in practice: A grounded theory perspective”, vol. 81, págs. 772-784, mayo de 2008.
- [45] K. Schwaber y J. Sutherland, “OOPSLA ’95 workshop proceedings”, en *Business object design and implementation*, The University of Michigan, 1995, pág. 118.
- [46] —, *The Definitive Guide to Scrum: The Rules of the Game*, Available at <https://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf>, 2013.
- [47] R. Kenneth, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley, 2012.
- [48] I. C. on conformity assessment”, *ISO 9001 - data per country and sector - 1993 to 2016*, 2016. dirección: <https://isotc.iso.org/livelink/livelink?func=11{\&}objId=18808772>.
- [49] B. Kitchenham y S Charters, *Guidelines for performing Systematic Literature Reviews in Software Engineering*, 2007.

- [50] F. J. Pino, M. T. Baldassarre, M. Piattini y G. Visaggio, “Harmonizing maturity levels from CMMI-DEV and ISO/IEC 15504”, *Journal of Software Maintenance and Evolution: Research and Practice*, vol. 22, n.º 4, págs. 279-296, DOI: 10.1002/spip.437. eprint: <https://onlinelibrary.wiley.com/doi/pdf/10.1002/spip.437>. dirección: <https://onlinelibrary.wiley.com/doi/abs/10.1002/spip.437>.
- [51] ISO/TC 176/SC 1 Concepts and terminology, “ISO 9000:2015 Quality management systems – Fundamentals and vocabulary”, vol. 4, 2015.
- [52] ISO/IEC JTC 1/SC 7, “ISO/IEC 15504, Software Process Improvement and Capability Determination”, 2013.
- [53] H. Oktaba, F. García, M. Piattini, F. Ruiz, F. J. Pino y C. Alquicira, “Software Process Improvement: The Competisoft Project”, *Computer*, vol. 40, 2007.
- [54] M. Melis, W. Ambu, S. Pinna y K. Mannaro, “Requirements of an ISO compliant XP tool”, en *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3092, 2004, págs. 266-269, ISBN: 9783540221371.

Anexos

Anexo A. Tablas de Homogeneización

SD1. Categoría de proceso	8. Operación
ID	Capítulo 8.
Nombre	Operación.
Propósito	Definido implícitamente.
Descripción	Definida implícitamente
Objetivos	Definidos implícitamente
SD3. Actividades	
<p>“ Requisito 8.1 Planificación y control operacional: La organización debe planificar, implementar y controlar los procesos (véase 4.4) necesarios para cumplir los requisitos para la provisión de productos y servicios, y para implementar las acciones determinadas en el capítulo 6, mediante:</p> <ul style="list-style-type: none"> a) la determinación de los requisitos para los productos y servicios; b) el establecimiento de criterios para <ul style="list-style-type: none"> 1) los procesos; 2) la aceptación de los productos y servicios; c) la determinación de los recursos necesarios para lograr la conformidad con los requisitos de productos y servicios; d) la implementación del control de los procesos de acuerdo con los criterios; e) la determinación, el mantenimiento y la conservación de la información de la información documentada en la extensión necesaria para: 1) tener confianza en que los procesos se han llevado a cabo según lo planificado; 2) demostrar la conformidad de los productos y servicios con sus requisitos; <p>La salida de esta planificación debe ser adecuada para las operaciones de la organización. La organización debe controlar los cambios planificados y revisar las consecuencias de los cambios no previstos, tomando acciones para mitigar cualquier efecto adverso, según sea necesario. La organización debe asegurarse de que los procesos contratados externamente estén controlados (véase 8.4)”</p>	
<p>“ Requisito 8.2.1 Requisitos para los productos y servicios → Comunicación con el cliente: La comunicación con los clientes debe incluir:</p> <ul style="list-style-type: none"> a) proporcionar la información relativa a los productos y servicios; b) tratar las consultas, los contratos o los pedidos, incluyendo los cambios; c) obtener la retroalimentación de los clientes relativa a los productos y servicios, incluyendo las quejas de los clientes d) manipular o controlar la propiedad del cliente e) establecer los requisitos específicos para las acciones de contingencia, cuando sea pertinente” 	

“ Requisito 8.2.2 Requisitos para los productos y servicios → Determinación de los requisitos para los productos y servicios:

Cuando se determinan los requisitos para los productos y servicios que se van a ofrecer a los clientes, la organización debe asegurarse de que:

- a) los requisitos para los productos y servicios se definen, incluyendo:
 - 1) cualquier requisito legal y reglamentario aplicable;
 - 2) aquellos considerados necesarios por la organización
- b) la organización puede cumplir con las declaraciones acerca de los productos y servicios que ofrece.”

“ Requisito 8.2.3.1 Requisitos para los productos y servicios → Revisión de los requisitos para los productos y servicios → La organización debe asegurarse de que tiene la capacidad de cumplir los requisitos para los productos y servicios que se van a ofrecer a los clientes. La organización debe llevar a cabo una revisión antes de comprometerse a suministrar productos y servicios a un cliente, para incluir:

- a) los requisitos especificados por el cliente, incluyendo los requisitos para las actividades de entrega y las posteriores a la misma;
 - b) los requisitos no establecidos por el cliente, pero necesarios para el uso especificado o previsto, cuando sea conocido;
 - c) los requisitos especificados por la organización;
 - d) los requisitos legales y reglamentarios aplicables a los productos y servicios;
 - e) las diferencias existentes entre los requisitos del contrato o pedido y los expresados previamente.
- La organización debe asegurarse de que, cuando se cambien los requisitos para los productos y servicios, la información documentada pertinente sea modificada, y de que las personas pertinentes sean conscientes de los requisitos modificados”

“ Requisito 8.2.3.2 Requisitos para los productos y servicios → Revisión de los requisitos para los productos y servicios → La organización debe conservar la información documentada, cuando sea aplicable:

- a) sobre los resultados de la revisión;
- b) sobre cualquier requisito nuevo para los productos y servicios”

“ Requisito 8.2.4 Requisitos para los productos y servicios → Cambios en los requisitos para los productos y servicios:

La organización debe asegurarse de que, cuando se cambien los requisitos para los productos y servicios, la información documentada pertinente sea modificada, y de que las personas pertinentes sean conscientes de los requisitos modificados.”

“ Requisito 8.3.1. Diseño y desarrollo de los productos y servicios → Generalidades:

La organización debe establecer, implementar y mantener un proceso de diseño y desarrollo que sea adecuado para asegurarse de la posterior provisión de productos y servicios.”

“ Requisito 8.3.2. Diseño y desarrollo de los productos y servicios → Planificación del diseño y desarrollo:

Al determinar las etapas y controles para el diseño y desarrollo, la organización debe considerar:

- a) la naturaleza, duración y complejidad de las actividades de diseño y desarrollo;
- b) las etapas del proceso requeridas, incluyendo las revisiones del diseño y desarrollo aplicables;
- c) las actividades requeridas de verificación y validación del diseño y desarrollo;
- d) las responsabilidades y autoridades involucradas en el proceso de diseño y desarrollo;
- e) las necesidades de recursos internos y externos para el diseño y desarrollo de los productos y servicios;
- f) la necesidad de controlar las interfaces entre las personas que participan activamente en el proceso de diseño y desarrollo;
- g) la necesidad de la participación activa de los clientes y usuarios en el proceso de diseño y desarrollo;
- h) los requisitos para la posterior provisión de productos y servicios;
- i) el nivel de control del proceso de diseño y desarrollo esperado por los clientes y otras partes interesadas pertinentes;
- j) la información documentada necesaria para demostrar que se han cumplido los requisitos del diseño y desarrollo”

“ Requisito 8.3.3. Diseño y desarrollo de los productos y servicios → Entradas para el diseño y desarrollo :

La organización debe determinar los requisitos esenciales para los tipos específicos de productos y servicios a diseñar y desarrollar. La organización debe considerar:

- a) los requisitos funcionales y de desempeño;
- b) la información proveniente de actividades previas de diseño y desarrollo similares;
- c) los requisitos legales y reglamentarios;
- d) normas o códigos de prácticas que la organización se ha comprometido a implementar
- e) las consecuencias potenciales de fallar debido a la naturaleza de los productos y servicios

Las entradas deben ser adecuadas para los fines del diseño y desarrollo, estar completas y sin ambigüedades.

Las entradas del diseño y desarrollo contradictorias deben resolverse.

La organización debe conservar la información documentada sobre las entradas del diseño y desarrollo”

“ Requisito 8.3.4. Diseño y desarrollo de los productos y servicios → Controles del diseño y desarrollo:

La organización debe aplicar controles al proceso de diseño y desarrollo para asegurarse de que:

- a) se definen los resultados a lograr;
- b) se realizan revisiones para evaluar la capacidad de los resultados del diseño y desarrollo;
- c) se realizan actividades de verificación para asegurarse de que las salidas del diseño y desarrollo cumplen los requisitos de las entradas;
- d) se realizan actividades de validación para asegurarse de que los productos y servicios resultantes satisfacen los requisitos para su aplicación especificada o uso previsto;
- e) se toma cualquier acción necesaria sobre los problemas determinados durante las revisiones, o las actividades de verificación y validación;
- f) se conserva la información documentada de estas actividades

NOTA: Las revisiones, la verificación y validación del diseño y desarrollo tienen propósitos distintos. Pueden realizarse de forma separada o en cualquier combinación, según sea idóneo para los productos y servicios de la organización.”

“ Requisito 8.3.5. Diseño y desarrollo de los productos y servicios → Salidas del diseño y desarrollo:

La organización debe asegurarse de que las salidas del diseño y desarrollo:

- a) cumplen los requisitos de las entradas;
- b) son adecuadas para los procesos posteriores para la provisión de productos y servicios;
- c) incluyen o hacen referencia a los requisitos de seguimiento y medición, cuando sea apropiado, y a los criterios de aceptación;
- d) especifican las características de los productos y servicios que son esenciales para su propósito previsto y su provisión segura y correcta.

La organización debe conservar información documentada sobre las salidas del diseño y desarrollo ”

“ Requisito 8.3.6. Diseño y desarrollo de los productos y servicios → Cambios del diseño y desarrollo:

La organización debe identificar, revisar y controlar los cambios hechos durante el diseño y desarrollo de los productos y servicios, o posteriormente en la medida necesaria para asegurarse de que no haya un impacto adverso en la conformidad con los requisitos:

La organización debe conservar la información documentada sobre:

- a) los cambios del diseño y desarrollo;
- b) los resultados de las revisiones;
- c) la autorización de los cambios;
- d) las acciones tomadas para prevenir los impactos adversos”

“ Requisito 8.4.1 Control de los procesos, productos y servicios suministrados externamente →

Generalidades:

La organización debe asegurarse de que los procesos, productos y servicios suministrados externamente son conformes a los requisitos.

La organización debe determinar los controles a aplicar a los procesos, productos y servicios suministrados externamente cuando:

- a) los productos y servicios de proveedores externos están destinados a incorporarse dentro de los propios productos y servicios de la organización
- b) los productos y servicios son proporcionados directamente a los clientes por proveedores externos en nombre de la organización;
- c) un proceso, o una parte de un proceso, es proporcionado por un proveedor externo como resultado de una decisión de la organización.

La organización debe determinar y aplicar criterios para la evaluación, la selección, el seguimiento del desempeño y la reevaluación de los proveedores externos, basándose en su capacidad para proporcionar procesos o productos y servicios de acuerdo con los requisitos. La organización debe conservar la información documentada de estas actividades y de cualquier acción necesaria que surja de las evaluaciones.”

“ Requisito 8.4.2 Control de los procesos, productos y servicios suministrados externamente →

Tipo y alcance del control:

La organización debe asegurarse de que los procesos, productos y servicios suministrados externamente no afectan de manera adversa a la capacidad de la organización de entregar productos y servicios conformes de manera coherente a sus clientes

La organización debe:

- a) asegurarse de que los procesos suministrados externamente permanecen dentro del control de su sistema de gestión de la calidad;
- b) definir los controles que pretende aplicar a un proveedor externo y los que pretende aplicar a las salidas resultantes;
- c) tener en consideración:
 - 1) el impacto potencial de los procesos, productos y servicios suministrados externamente en la capacidad de la organización de cumplir regularmente los requisitos del cliente y los legales y reglamentarios aplicables;
 - 2) la eficacia de los controles aplicados por el proveedor externo;
- d) determinar la verificación, u otras actividades necesarias para asegurarse de que los procesos, productos y servicios suministrados externamente cumplen los requisitos.”

“ Requisito 8.4.3 Control de los procesos, productos y servicios suministrados externamente →

Información para los proveedores externos:

La organización debe asegurarse de la adecuación de los requisitos antes de su comunicación al proveedor externo.

La organización debe comunicar a los proveedores externos sus requisitos para:

- a) los procesos, productos y servicios a proporcionar;
- b) la aprobación de:
 - 1) productos y servicios;
 - 2) métodos, procesos y equipos;
 - 3) la liberación de productos y servicios;
- c) la competencia, incluyendo cualquier calificación requerida de las personas;
- d) las interacciones del proveedor externo con la organización;
- e) el control y el seguimiento del desempeño del proveedor externo a aplicar por parte de la organización;
- f) las actividades de verificación o validación que la organización, o su cliente, pretende llevar a cabo en las instalaciones del proveedor externo.”

“ Requisito 8.5.1 Producción y provisión del servicio → Control de la producción y provisión del servicio:

La organización debe implementar la producción y provisión del servicio bajo condiciones controladas.

Las condiciones controladas deben incluir, cuando sea aplicable:

- a) la disponibilidad de información documentada que defina:
 - 1) las características de los productos a producir, los servicios a prestar, o las actividades a desempeñar;
 - 2) los resultados a alcanzar;
- b) la disponibilidad y el uso de los recursos de seguimiento y medición adecuados;
- c) la implementación de actividades de seguimiento y medición en las etapas apropiadas para verificar que se cumplen los criterios para el control de los procesos o sus salidas, y los criterios de aceptación para los productos y servicios;
- d) el uso de la infraestructura y el entorno adecuado para la operación de los procesos;
- e) la designación de personas competentes, incluyendo cualquier calificación requerida;
- f) la validación y revalidación periódica de la capacidad para alcanzar los resultados planificados de los procesos de producción y de prestación del servicio, cuando las salidas resultantes no pueden verificarse mediante actividades de seguimiento o medición posteriores;
- g) la implementación de acciones para prevenir los errores humanos;
- h) la implementación de actividades de liberación, entrega y posteriores a la entrega.”

“ Requisito 8.5.2 Producción y provisión del servicio → Identificación y trazabilidad:

La organización debe utilizar los medios apropiados para identificar las salidas, cuando sea necesario, para asegurar la conformidad de los productos y servicios.

La organización debe identificar el estado de las salidas con respecto a los requisitos de seguimiento y medición a través de la producción y prestación del servicio.

La organización debe controlar la identificación única de las salidas cuando la trazabilidad sea un requisito, y debe conservar la información documentada necesaria para permitir la trazabilidad.”

“ Requisito 8.5.3 Producción y provisión del servicio → Propiedad perteniente a los clientes o proveedores externos:

La organización debe cuidar la propiedad perteniente a los clientes o a proveedores externos mientras esté bajo el control de la organización o esté siendo utilizado por la misma.

La organización debe identificar, verificar, proteger y salvaguardar la propiedad de los clientes o de los proveedores externos suministrada para su utilización o incorporación dentro de los productos y servicios.

Cuando la propiedad de un cliente o de un proveedor externo se pierda, deteriore, o de algún otro modo se considere inadecuada para su uso, la organización debe informar de esto al cliente o proveedor externo y conservar la información documentada sobre lo ocurrido.

NOTA: La propiedad de un cliente o de un proveedor externo puede incluir materiales, componentes, herramientas y equipos, instalaciones, propiedad intelectual y datos personales.”

“ Requisito 8.5.4 Producción y provisión del servicio → Preservación:

La organización debe preservar las salidas durante la producción y prestación del servicio, en la medida necesaria para asegurarse de la conformidad con los requisitos.

NOTA: La preservación puede incluir la identificación, la manipulación, el control de la contaminación, el embalaje, el almacenamiento, la transmisión de la información o el transporte, y la protección.”

“ Requisito 8.5.5 Producción y provisión del servicio → Actividades posteriores a la entrega:

La organización debe cumplir con los requisitos para las actividades posteriores a la entrega asociadas con los productos y servicios.

Al determinar el alcance de las actividades posteriores a la entrega que se requieren, la organización debe considerar:

- a) los requisitos legales y reglamentarios;
- b) las consecuencias potenciales no deseadas asociadas a sus productos y servicios;
- c) la naturaleza, el uso y la vida útil prevista de sus productos y servicios
- d) los requisitos del cliente;
- e) la retroalimentación del cliente

NOTA: Las actividades posteriores a la entrega pueden incluir acciones cubiertas por las condiciones de la garantía, obligaciones contractuales como servicios de mantenimiento, y servicios suplementarios como el reciclaje o la disposición final.”

<p>“ Requisito 8.5.6 Producción y provisión del servicio → Control de los cambios: La organización debe revisar y controlar los cambios para la producción o la prestación del servicio, en la extensión necesaria para asegurarse de la continuidad en la conformidad con los requisitos. La organización debe conservar información documentada que describa los resultados de la revisión de los cambios, las personas que autorizan el cambio y de cualquier acción necesaria que surja de la revisión.”</p>
<p>“ Requisito 8.6 Liberación de los productos y servicios: La organización debe implementar las disposiciones planificadas, en las etapas adecuadas, para verificar que se cumplen con los requisitos de los productos y servicios. La liberación de los productos y servicios al cliente no debe llevarse a cabo hasta que se hayan completado satisfactoriamente las disposiciones planificadas, a menos que sea aprobado de otra manera por una autoridad pertinente y, cuando sea aplicable, por el cliente. La organización debe conservar la información documentada sobre la liberación de los productos y servicios. La información documentada debe incluir:</p> <ul style="list-style-type: none"> a) evidencia de la conformidad con los criterios de aceptación; b) trazabilidad a las personas que autorizan la liberación;”
<p>“ Requisito 8.7.1 Control de las salidas no conformes → La organización debe asegurarse de que las salidas que no sean conformes con sus requisitos se identifican y se controlan para prevenir su uso o entrega no intencionada. La organización debe tomar las acciones adecuadas basándose en la naturaleza de la no conformidad y en su efecto sobre la conformidad de los productos y servicios. Esto se debe aplicar también a los productos y servicios no conformes detectados después de la entrega los productos, durante o después de la provisión de los servicios. La organización debe tratar las salidas no conformes de una o más de las siguientes maneras:</p> <ul style="list-style-type: none"> a) corrección b) separación, contención, devolución o suspensión de provisión de productos y servicios; c) información al cliente; d) obtención de autorización para su aceptación bajo consecuencia <p>Debe verificarse la conformidad con los requisitos cuando se corrigen las salidas no conformes.”</p>
<p>“ Requisito 8.7.2 Control de las salidas no conformes → La organización debe conservar la información documentada que:</p> <ul style="list-style-type: none"> a) describa la no conformidad; b) describa las acciones tomadas; c) describa todas las consecuencias obtenidas; d) identifique la autoridad que decide la acción con respecto a la no conformidad.”

Tabla 29. Homogeneización del Capítulo 8. Operación, de ISO 9001:2015.

SD1. Categoría de proceso	9. Evaluación del desempeño
ID	Capítulo 9.
Nombre	Evaluación del desempeño.
Propósito	Definido implícitamente.
Descripción	Definida implícitamente
Objetivos	Definidos implícitamente
SD3. Actividades	
<p>“ Requisito 9.1.1 Seguimiento, medición, análisis y evaluación → Generalidades: La organización debe determinar:</p> <ul style="list-style-type: none"> a) qué necesita seguimiento y medición; b) los métodos de seguimiento, medición, análisis y evaluación necesarios para asegurar los resultados válidos; c) cuándo se deben llevar a cabo el seguimiento y la medición; d) cuándo se deben analizar y evaluar los resultados del seguimiento y la medición. <p>La organización debe evaluar el desempeño y la eficacia del sistema de gestión de la calidad. La organización debe conservar la información documentada apropiada como evidencia de los resultados.”</p>	
<p>“ Requisito 9.1.2 Seguimiento, medición, análisis y evaluación → Satisfacción del cliente: La organización debe realizar el seguimiento de las percepciones de los clientes del grado en que se cumplen sus necesidades y expectativas. La organización debe determinar los métodos para obtener, realizar el seguimiento y revisar esta información. NOTA: los ejemplos de seguimiento de las percepciones del cliente pueden incluir las encuestas al cliente, la retroalimentación del cliente sobre los productos y servicios entregados, las reuniones con los clientes, el análisis de las cuotas de mercado, las felicitaciones, las garantías utilizadas y los informes de agentes comerciales.”</p>	
<p>“ Requisito 9.1.3 Seguimiento, medición, análisis y evaluación → Análisis y evaluación: La organización debe analizar y evaluar los datos y la información apropiados que surgen por el seguimiento y la medición. Los resultados del análisis deben utilizarse para evaluar:</p> <ul style="list-style-type: none"> a) la conformidad de los productos y servicios; b) el grado de satisfacción del cliente; c) el desempeño y la eficacia del sistema de gestión de la calidad; d) si lo planificado se ha implementado de forma eficaz; e) la eficacia de las acciones tomadas para abordar los riesgos y oportunidades; f) el desempeño de los proveedores externos; g) la necesidad de mejoras en el sistema de gestión de la calidad <p>NOTA: Los métodos para analizar los datos pueden incluir técnicas estadísticas.”</p>	

“ Requisito 9.2.1 Auditoría interna → La organización debe llevar a cabo auditorías internas a intervalos planificados para proporcionar información acerca de si el sistema de gestión de la calidad:

- a) es conforme con:
 - 1) los requisitos propios de la organización para su sistema de gestión de la calidad;;
 - 2) los requisito de esta Norma Internacional;
- b) se implementa y se mantiene eficazmente

NOTA: Los métodos para analizar los datos pueden incluir técnicas estadísticas.”

“ Requisito 9.2.2 Auditoría interna → La organización debe:

- a) planificar, establecer, implementary mantener uno o varios programas de auditoría que incluyan la frecuencia, los métodos, las responsabilidades, los requisitos de planificación y la elaboración de informes, que deben tener en consideración la importancia de los procesos involucrados, los cambios que afecten a la organización y los resultados de las auditorías previas;
- b) definir los criterios de la auditoría y el alcance para cada auditoría
- c) seleccionar los auditores y llevar a cabo auditorías para asegurarse de la objetividad y la imparcialidad del proceso de auditoría;
- d) asegurarse de que los resultados de las auditorías se informen a la dirección pertinente;
- e) realizar las correcciones y tomar las acciones correctivas adecuadas sin demora injustificada;
- f) conservar información documentada como evidencia de la implementación del programa de auditoría y de los resultados de las auditorías.

NOTA: Véase la Norma ISO 19011 a modo de orientación.”

“ Requisito 9.3.1 Revisión por la dirección → Generalidades:

La alta dirección debe revisar el sistema de gestión de la calidad de la organización a intervalos planificados, para asegurarse de su conveniencia, adecuación, eficacia, y alineación continuas con la dirección estratégica de la organización.”

<p>“ Requisito 9.3.2 Revisión por la dirección → Entradas de la revisión por la dirección:</p> <p>La revisión por la dirección debe planificarse y llevarse a cabo incluyendo consideraciones sobre:</p> <ul style="list-style-type: none"> a) el estado de las acciones de las revisiones por la dirección previas; b) los cambios en las cuestiones externas e internas que sean pertinentes al sistema de gestión de la calidad; c) la información sobre el desempeño y la eficacia del sistema de gestión de la calidad, incluidas las tendencias relativas a : <ul style="list-style-type: none"> 1) la satisfacción del cliente y la retroalimentación de las partes interesadas pertinentes; 2) el grado en que se han logrado los objetivos de calidad; 3) el desempeño de los procesos y conformidad de los productos y servicios; 4) las no conformidades y acciones correctivas; 5) los resultados de seguimiento y medición; 6) los resultados de las auditorías; 7) el desempeño de los proveedores externos; d) la adecuación de los recursos; e) la eficacia de las acciones tomadas para abordar los riesgos y las oportunidades (véase 6.1); f) las oportunidades de mejora <p>NOTA: Véase la Norma ISO 19011 a modo de orientación.”</p>
<p>“ Requisito 9.3.3 Revisión por la dirección → Salidas de la revisión por la dirección:</p> <p>Las salidas de la revisión por la dirección deben incluir las decisiones y acciones relacionadas con:</p> <ul style="list-style-type: none"> a) las oportunidades de mejora; b) cualquier necesidad de cambio en el sistema de gestión de la calidad; c) las necesidades de recursos; <p>La organización debe conservar información documentada como evidencia de los resultados de las revisiones por la dirección.</p> <p>NOTA: Véase la Norma ISO 19011 a modo de orientación.”</p>

Tabla 30. Homogeneización del Capítulo 9. Evaluación del desempeño, de ISO 9001:2015.

SD1. Categoría de proceso	10. Mejora
ID	Capítulo 10.
Nombre	Mejora.
Propósito	Definido implícitamente.
Descripción	Definida implícitamente
Objetivos	Definidos implícitamente
SD3. Actividades	

<p>“ Requisito 10.1 Mejora → Generalidades:</p> <p>La organización debe determinar y seleccionar las oportunidades de mejora e implementar cualquier acción necesaria para cumplir los requisitos del cliente y aumentar la satisfacción del cliente. Éstas deben incluir:</p> <ul style="list-style-type: none"> a) mejorar los productos y servicios para cumplir los requisitos, así como considerar las necesidades y expectativas futuras; b) corregir, prevenir o reducir los efectos no deseados; c) mejorar el desempeño y la eficacia del sistema de gestión de la calidad. <p>NOTA: Los ejemplos de mejora pueden incluir corrección, acción correctiva, mejora continua, cambio abrupto, innovación y reorganización”</p>
<p>“ Requisito 10.2.1 Mejora → No conformidad y acción correctiva → Cuando ocurra una no conformidad, incluida cualquiera originada por quejas, la organización debe:</p> <ul style="list-style-type: none"> a) reaccionar ante la no conformidad y, cuando sea aplicable: <ul style="list-style-type: none"> 1) tomar acciones para controlarla y corregirla; 2) hacer frente a las consecuencias; b) evaluar la necesidad de acciones para eliminar las causas de la no conformidad, con el fin de que no vuelva a ocurrir ni ocurra en otra parte, mediante: <ul style="list-style-type: none"> 1) la revisión y el análisis de la no conformidad; 2) la determinación de las causas de la no conformidad; 3) la determinación de si existen no conformidades similares, o que potencialmente pueda ocurrir; c) implementar cualquier acción necesaria; d) revisar la eficacia de cualquier acción correctiva tomada; e) si fuera necesario, actualizar los riesgos y las oportunidades determinados durante la planificación; y f) si fuera necesario, hacer cambios al sistema de gestión de la calidad. <p>Las acciones correctivas deben ser apropiadas a los efectos de las no conformidades encontradas.”</p>
<p>“ Requisito 10.2.2 Mejora → No conformidad y acción correctiva → La organización debe conservar información documentada como evidencia de:</p> <ul style="list-style-type: none"> a) la naturaleza de las no conformidades y cualquier acción tomada posteriormente; b) los resultados de cualquier acción correctiva.”
<p>“ Requisito 10.3 Mejora → Mejora continua:</p> <p>La organización debe mejorar continuamente la conveniencia, adecuación y eficacia del sistema de gestión de la calidad.</p> <p>La organización debe considerar los resultados del análisis y la evaluación, y las salidas de la revisión por la dirección, para determinar si hay necesidades u oportunidades que deben considerarse como parte de la mejora continua.”</p>

Tabla 31. Homogeneización del capítulo de *10. Mejora*, de ISO 9001 en los elementos de proceso de la estructura común.

SD1. Categoría de elemento	Roles
ID	The SCRUM Team
Nombre	El Equipo SCRUM
Propósito	“ Los equipos de SCRUM entregan productos iterativa e incrementalmente, maximizando oportunidades de retroalimentación. Entregas incrementales de producto “Hecho” aseguran que una versión potencialmente útil de producto funcional está siempre disponible.”
Descripción	“ El equipo SCRUM consta de un Product Owner, el Equipo de Desarrollo y un SCRUM Master. Los equipos SCRUM son auto organizados y multifuncionales. Equipos auto organizados escogen la mejor forma de conseguir sus objetivos de trabajo, en lugar de ser dirigidos directamente por terceros fuera del equipo. Equipos multifuncionales tienen todas las competencias necesarias para lograr los objetivos de trabajo sin depender de otros que no son parte del equipo. El modelo del equipo SCRUM está diseñado para optimizar la flexibilidad, creatividad, y productividad. El equipo SCRUM ha probado ser por sí mismo incrementalmente efectivo para todos los usos de SCRUM y para cualquier trabajo complejo.”
Objetivos	Definidos implícitamente
SRR1. Roles	
<p>Rol 1. El Product Owner: “ El Product Owner es el responsable de maximizar el valor del producto resultante del trabajo del Equipo de Desarrollo. La forma en que se lleva a cabo puede variar dependiendo de la organización, el equipo SCRUM e individuos.</p> <p>El Product Owner es la única persona responsable de gestionar el Product Backlog. La gestión del Product Backlog incluye:</p> <ul style="list-style-type: none"> ■ Expresar claramente los ítems del Product Backlog; ■ Ordenar los ítems en el Product Backlog para lograr metas y objetivos de la mejor manera; ■ Asegurar que el Product Backlog es visible, transparente, y claro para todos, y muestra en lo que el equipo SCRUM trabajará, y ■ Asegurar que el Equipo de Desarrollo entienda los ítems del Product Backlog al nivel necesario <p>El Product Owner puede llevar a cabo el trabajo mencionado previamente, o puede hacerlo el Equipo de Desarrollo. De cualquier manera, el Product Owner sigue siendo el responsable.</p> <p>El Product Owner es una persona, no un comité. El Product Owner puede representar los deseos de un comité en el Product Backlog, pero aquellos que quieran modificar la prioridad de los ítems del Product Backlog deben dirigirse al Product Owner.</p> <p>Para que el Product Owner tenga éxito, toda la organización debe respetarlo y respetar sus decisiones. Las decisiones del Product Owner son visibles en el contenido y ordenamiento del Product Backlog. Nadie puede forzar al Equipo de Desarrollo a trabajar a partir de otro conjunto de requerimientos.”</p>	

“ Rol 2. El Equipo de Desarrollo:

El Equipo de Desarrollo consiste de profesionales quienes realizan el trabajo de entregar un **incremento** potencialmente liberable de producto **“Hecho”** al final de cada **Sprint**. Un **incremento “Hecho”** es requerido en el **Sprint Review**. Solo los miembros del Equipo de Desarrollo crean el **incremento**.

Los Equipos de Desarrollo son estructurados y empoderados por la organización para gestionar su propio trabajo. La sinergia resultante optimiza la eficiencia y efectividad.

Los Equipos de Desarrollo tienen las siguientes características:

- Son auto organizados. Nadie (ni el **SCRUM Master**) le dice al Equipo de Desarrollo como convertir el **Product Backlog** en incrementos de funcionalidad potencialmente liberables;
- Los Equipos de Desarrollo son multifuncionales, con todas las habilidades necesarias para crear un **incremento** del producto;
- SCRUM no reconoce títulos para los miembros del Equipo de Desarrollo, no importa el trabajo que esté llevando a cabo cada persona;
- SCRUM no reconoce sub equipos en el Equipo de Desarrollo, sin importar los dominios que deben ser dirigidos como pruebas, arquitectura, operaciones, lógica del negocio, y;
- Individuos del Equipo de Desarrollo pueden tener habilidades especializadas y areas de enfoque, pero la responsabilidad pertenece al Equipo de Desarrollo como un todo.

Tamaño del Equipo de Desarrollo: Un tamaño óptimo del Equipo de Desarrollo suficientemente pequeño para permanecer ágil y suficientemente grande para completar trabajo significativo en un **Sprint**. Menos de 3 miembros en el Equipo de Desarrollo pueden encontrar restricciones respecto a las habilidades durante el **Sprint**, causando imposibilidad de entregar un **incremento** potencialmente liberable. Tener más de 9 miembros requiere demasiada coordinación. Equipos de Desarrollo muy grandes generan demasiada complejidad para que un proceso empírico sea útil. Los roles de **Product Owner** y **SCRUM Master** no están incluidos en esta cuenta a menos que estén llevando a cabo trabajo del **Sprint Backlog**.”

Rol 3. El SCRUM Master:

“ El SCRUM Master es responsable de promover y apoyar al Equipo SCRUM a seguir SCRUM como está definido en la guía, esto ayudando a todos a entender la teoría de SCRUM, las prácticas, reglas y valores.

El SCRUM Master es un líder servidor para el Equipo SCRUM. Ayuda a aquellos por fuera del Equipo SCRUM a entender cuáles de sus interacciones con el Equipo SCRUM son de ayuda y cuáles no lo son. El SCRUM Master ayuda a todos a cambiar estas interacciones para maximizar el valor creado por el Equipo SCRUM.

El servicio del SCRUM Master al Product Owner:

- Asegurar que las metas, el alcance y el dominio del producto son entendidos por todos en el Equipo SCRUM tan bien como sea posible;
- Encontrar técnicas para una gestión efectiva del **Product Backlog**;
- Ayudar a que el Equipo SCRUM entienda la necesidad de que los ítems del **Product Backlog** sean claros y concisos;
- Entender la planeación de producto en un entorno empírico;
- Asegurar que el **Product Owner** conoce como ordenar el **Product Backlog** para maximizar el valor;
- Entender y practicar la agilidad; y,
- Facilitar los eventos de SCRUM cuando sean solicitados o necesarios.

El servicio del SCRUM Master al Equipo de Desarrollo:

- Entrenar al **Equipo de Desarrollo** en auto organización y multifuncionalidad;
- Ayudar al **Equipo de Desarrollo** a crear productos de alto valor;
- Remover impedimentos al progreso del **Equipo de Desarrollo**
- Facilitar los eventos de SCRUM cuando sean solicitados o necesarios; y,
- Entrenar al **Equipo de Desarrollo** en entornos organizacionales en los cuales SCRUM aún no ha sido adoptado y entendido completamente.

El servicio del SCRUM Master a la organización:

- Liderar y entrenar a la organización en la adopción de SCRUM;
- Planear implementaciones de SCRUM dentro de la organización;
- Ayudar a los colaboradores e interesados a entender y promover SCRUM y el desarrollo del producto empírico.
- Realizar cambios que incrementen la productividad del Equipo SCRUM; y,
- Trabajar con otros SCRUM Masters para incrementar la efectividad de la aplicación de SCRUM en la organización.”

Tabla 32. Homogeneización de los roles de SCRUM.

SD1. Categoría de Elemento	Eventos
ID	SCRUM Events
Nombre	Eventos de SCRUM
Propósito	“ Además del Sprint que es un contenedor para todos los otros eventos, cada evento de SCRUM es una oportunidad formal para inspeccionar y adaptar algo. Estos eventos están específicamente diseñados para permitir transparencia e inspección crítica. La exclusión de cualquiera de estos eventos resulta en una reducción de la transparencia y es una oportunidad perdida de inspeccionar y adaptar.”
Descripción	“ Los eventos prescritos son usados en SCRUM para crear regularidad y minimizar la necesidad e reuniones no definidas en SCRUM. Todos los eventos tienen una duración máxima. Una vez que un Sprint comienza, su duración es fija y no puede ser acortada o extendida. Los otros eventos pueden ser finalizados cuando el propósito los mismos ha sido alcanzado, asegurando que una cantidad apropiada de tiempo ha sido invertida.”
Objetivos	Definidos implícitamente
SD3. Actividades	

“ Actividad 1. Sprint:

El corazón de SCRUM es un Sprint, una duración máxima de un mes o menos en la cual un **incremento** del producto **“Hecho”** potencialmente entregable es creado. Los sprints tienen duraciones consistentes durante un esfuerzo de desarrollo. Un nuevo Sprint comienza inmediatamente después de la conclusión del Sprint previo.

Los sprints contienen y consisten de un **Sprint Planning**, **Daily SCRUM**, el trabajo de desarrollo, un **Sprint Review** y un **Sprint Retrospective**.

Durante el Sprint:

- No se hacen cambios que podrían poner en peligro el Sprint;
- Las metas de calidad no se reducen; y,
- El alcance puede ser clarificado y re negociado entre el **Product Owner** y el **Equipo de Desarrollo** a medida que se obtiene más conocimiento del proyecto.

Cada Sprint puede ser considerado un proyecto con no más de un mes como horizonte. Como los proyectos, los sprints son utilizados para conseguir algo. Cada Sprint tiene una meta acerca de lo que debe ser construido, un diseño y un plan flexible que guiará la construcción, el trabajo y el incremento del producto resultante.

Los sprints están limitados a un mes calendario. Cuando el horizonte de un Sprint es demasiado largo la definición de lo que está siendo construido puede cambiar, la complejidad y el riesgo pueden aumentar. Los sprints ofrecen predictibilidad al asegurar inspección y adaptación del progreso hacia la meta del Sprint al menos cada mes calendario. Además de que limitan el riesgo a un mes calendario respecto al costo.

Cancelar un Sprint: Un Sprint puede ser cancelado antes de que el tiempo sea completado. Solo el **Product Owner** tiene la autoridad para cancelar el Sprint, aunque él o ella pueden hacerlo bajo la influencia de los interesados, del **Equipo de Desarrollo** o del **SCRUM Master**.

Un Sprint sería cancelado si la meta se vuelve obsoleta. Esto puede ocurrir si la organización cambia la dirección o si el mercado o las condiciones tecnológicas cambian. En general, un Sprint debería ser cancelado si ya no tiene sentido continuar bajo las circunstancias dadas. Pero, debido a la duración corta de los sprints, rara vez son cancelados.

Cuando un Sprint es cancelado, cualquier ítem del **Product Backlog “Hecho”** es revisado. Si parte del trabajo es liberable, el **Product Owner** típicamente lo acepta. Todos los ítems del **Product Backlog** incompletos son re estimados y puestos de vuelta en la lista.

Cancelar un Sprint consume recursos, ya que todos se deben reunir en otro **Sprint Planning** para comenzar un nuevo Sprint. Cancelar un Sprint es traumático para el Equipo SCRUM y poco común.”

“ **Actividad 2. Sprint Planning:**

El trabajo que será llevado a cabo en un **Sprint** es planeado en el Sprint Planning. Este plan es creado por el trabajo colaborativo de todo el Equipo SCRUM.

Un Sprint Planning tiene una duración máxima de 8 horas para un **Sprint** de un mes. Para sprints más cortos el Sprint Planning es usualmente más corto. El **SCRUM Master** se asegura que el evento se lleve a cabo y que los asistentes entiendan su propósito. El **SCRUM Master** enseña al Equipo SCRUM a mantener el evento dentro del tiempo permitido.

Un Sprint Planning responde lo siguiente:

- Qué puede ser entregado en el **incremento** resultante del **Sprint** que viene?
- Cómo se logrará llevar a cabo el trabajo necesario para construir el **incremento**?

Qué puede ser hecho durante este Sprint? El **Equipo de Desarrollo** trabaja para visualizar la funcionalidad que será desarrollada durante el **Sprint**. El **Product Owner** discute el objetivo debería ser alcanzado en el **Sprint** y los ítems del **Product Backlog** que, si se completan en el **Sprint** alcanzarían ese objetivo. Todo el Equipo SCRUM colabora en el entendimiento del trabajo del **Sprint**.

La entrada para esta reunión es el **Product Backlog**, el último **incremento** del producto, la capacidad proyectada del **Equipo de Desarrollo** durante el **Sprint** y el desempeño previo del **Equipo de Desarrollo**. El número de ítems seleccionados del **Product Backlog** para el **Sprint** depende únicamente del **Equipo de Desarrollo**. Solo este puede evaluar lo que pueden lograr para el **Sprint** que viene.

Durante el Sprint Planning el Equipo SCRUM también define una meta del **Sprint**, que es un objetivo que será perseguido dentro del **Sprint** a través de la implementación del **Product Backlog**, y guía al **Equipo de Desarrollo** sobre por qué se está construyendo el incremento.

Cómo se llevará a cabo lo que será entregado al final del Sprint?: Habiendo establecido la meta del **Sprint** y seleccionado los ítems del **Product Backlog** para el Sprint, el Equipo de Desarrollo decide cómo convertirá la funcionalidad en un **incremento del producto “Hecho”** durante el **Sprint**. Los ítems del **Product Backlog** seleccionados para este **Sprint** más el plan para entregarlos es el **Sprint Backlog**. El **Equipo de Desarrollo** usualmente comienza diseñando el sistema y el trabajo necesario para convertir el **Product Backlog** en un **incremento** del producto funcional. EL trabajo puede variar en tamaño o en trabajo estimado. De cualquier forma, suficiente trabajo es planeado durante un Sprint Planning para que el **Equipo de Desarrollo** visualice lo que cree que puede hacer durante el **Sprint** que viene. El trabajo planeado para los primeros días del **Sprint** por el **Equipo de Desarrollo** es descompuesto al final de la reunión. A menudo en unidades de un día o menos el **Equipo de Desarrollo** se auto organiza para asignarse el trabajo del **Sprint Backlog**, tanto en el Sprint Planning como sea necesario durante el **Sprint**.

El **Product Owner** puede ayudar a aclarar los ítems seleccionados del **Product Backlog** y hacer intercambios. Si el **Equipo de Desarrollo** determina que tiene mucho o muy poco trabajo. Puede re negociar los ítems seleccionados del **Product Backlog** con el **Product Owner**. El **Equipo de Desarrollo** puede también invitar a otras personas para que provean información técnica o del dominio del problema. Al final del Sprint Planning, **El Equipo de Desarrollo** debería ser capaz de explicar al **Product Owner** y al **SCRUM Master** cómo planea auto organizarse como equipo para conseguir la meta del **Sprint** y construir el **incremento** del producto.”

“ Actividad 3. Daily SCRUM:

El Daily SCRUM es una actividad de 15 minutos para el **Equipo de Desarrollo**, es llevada a cabo todos los días del **Sprint**. El **Equipo de Desarrollo** planea el trabajo para las siguientes 24 horas. Esto optimiza la colaboración y el desempeño del equipo al inspeccionar el trabajo desde el último Daily SCRUM y visualizando el trabajo que viene en el **Sprint**. El Daily SCRUM es llevada a cabo siempre a la misma hora y en el mismo lugar todos los días para reducir la complejidad.

El **Equipo de Desarrollo** utiliza el Daily SCRUM para inspeccionar el progreso hacia la meta del **Sprint** y como el progreso lleva a completar el trabajo del **Sprint Backlog**. El Daily SCRUM optimiza la probabilidad de que el **Equipo de Desarrollo** alcance la meta del **Sprint**. Cada día, el **Equipo de Desarrollo** debería entender cómo pretende trabajar como un equipo auto organizado para lograr la meta del **Sprint** y construir el **incremento** del producto para el final del **Sprint**.

La estructura de la reunión es establecida por el **Equipo de Desarrollo** y puede ser conducida en diferentes maneras si se enfoca en el progreso hacia la meta del **Sprint**. Algunos Equipos de Desarrollo utilizarán preguntas, algunos lo harán más como una discusión. Una forma de realizar la reunión es hacer las siguientes preguntas:

- Qué hiciste ayer que ayudó al **Equipo de Desarrollo** en alcanzar la meta del **Sprint**?
- Qué harás hoy para ayudar al **Equipo de Desarrollo** a alcanzar la meta del **Sprint**?
- Tienes algún impedimento que prevenga al **Equipo de Desarrollo** en alcanzar la meta del **Sprint**?

El **Equipo de Desarrollo** o los miembros del equipo a menudo se reúnen inmediatamente después del **Daily SCRUM** para discusiones más detalladas, o para adaptar, o re planear, el resto del trabajo del **Sprint**.

El SCRUM Master se asegura que el **Equipo de Desarrollo** tenga la reunión, pero es el **Equipo de Desarrollo** el responsable de conducir el **Daily SCRUM**. El SCRUM Master enseña al **Equipo de Desarrollo** a mantener el **Daily SCRUM** dentro de los 15 minutos cada día.

El **Daily SCRUM** es una reunión interna para el **Equipo de Desarrollo**. Si otros están presentes, el **SCRUM Master** se asegura que no afecten la reunión.

Las Daily SCRUMs mejoran la comunicación, eliminan otras reuniones, identifican impedimentos a remover para el **Equipo de Desarrollo**, destacan y promueven la rápida toma de decisiones, y mejoran el nivel de conocimiento del **Equipo de Desarrollo**. Esta es una reunión clave para inspeccionar y adaptar.”

“ Actividad 4. Sprint Review:

Un **Sprint Review** es llevado a cabo al final del **Sprint** para inspeccionar el **incremento** y apdatar el **Product Backlog** si es necesario. Durante el **Sprint Review**, el Equipo SCRUM y los interesados colaboran sobre lo que fue hecho en el **Sprint**. Basándose en ello y en cualquier cambio al **Product Backlog** durante el **Sprint**, los asistentes colaboran en las siguientes cosas que pueden ser hechas para optimizar el valor. Esta es una reunión informal, no una reunión de estado del proyecto, y la presentación del **incremento** es para obtener retroalimentación y fomentar la colaboración.

Esta reunión dura máximo 4 horas para un **Sprint** de 1 mes. Para sprints más cortos, el evento es usualmente más corto. El **SCRUM Master** se asegura que el evento se lleva a cabo y que los asistentes entienden su propósito. El **SCRUM Master** enseña a todos los involucrados a mantener la reunión dentro del tiempo previsto.

El **Sprint Review** incluye los siguientes elementos:

- Los asistentes incluyen el Equipo SCRUM y los interesados claves invitados por el **Product Owner**;
- El Product Owner explica cuales ítems del **Product Backlog** han sido **“Hechos”** y cuales no;
- El **Equipo de Desarrollo** discute lo que se hizo bien en el **Sprint**, los impedimentos encontrados y la forma en que se solucionaron;
- El **Equipo de Desarrollo** demuestra el trabajo que fue **“Hecho”** y responde las preguntas sobre el **incremento**;
- el **ProductOwner** discute la condición actual del **Product Backlog** y proyecta el objetivo probable y las fechas de entrega basándose en el progreso realizado a la fecha si es necesario;
- Todos colaboran en lo que sigue por hacer, de tal manera que el **Sprint Review** provee información valiosa para el **Sprint** que sigue;
- Una revisión de cómo el mercado o el uso potencial del producto puede haber cambiado y qué es lo más valioso para hacer a continuación; y,
- Una revisión del cronograma, presupuesto, capacidades potenciales, y el mercado para los siguientes incrementos de funcionalidad o capacidad del producto.

El resultado del **Sprint Review** es un **Product Backlog** revisado que define los probables ítems del **Product Backlog** que serán realizados en el siguiente **Sprint**.”

“ Actividad 5. Sprint Retrospective:

El Sprint Retrospective es una oportunidad para el Equipo SCRUM para inspeccionarse a sí mismo y crear un plan para promulgar la mejora en el siguiente Sprint

El Sprint Retrospective ocurre después del Sprint Review y antes al siguiente Sprint Planning. Esta reunión dura máximo 3 horas para sprints de 1 mes. Para sprints más cortos, esta reunión es usualmente más corta. El SCRUM Master asegura que la reunión se lleva a cabo y que los asistentes entienden su propósito.

El propósito del Sprint Retrospective es:

- Inspeccionar cómo estuvo el último Sprint con respecto a las personas, relaciones, procesos y herramientas;
- Identificar y ordenar las cosas que se hicieron bien durante el Sprint que terminó y las que se pueden mejorar; y,
- Crear un plan para implementar las mejoras de tal forma que el Equipo SCRUM haga su trabajo

El SCRUM Master motiva al Equipo SCRUM a mejorar, dentro del marco de trabajo de SCRUM, su proceso de desarrollo y sus prácticas para hacer el proceso más efectivo y disfrutable para el siguiente Sprint. Durante cada Sprint Retrospective, el Equipo SCRUM planea formas para incrementar la calidad del producto al mejorar los procesos de trabajo o adaptar la definición de “Hecho”, si es apropiado y no entra en conflicto con los estándares del producto y de la organización.”

Tabla 33. Homogeneización de los Eventos de SCRUM.

SD1. Categoría de Elemento	Artefactos
ID	SCRUM Artifacts
Nombre	Artefactos de SCRUM
Propósito	“ Los artefactos de SCRUM están diseñados para maximizar la transparencia de información clave de tal forma que todos tengna el mismo entendimiento del artefacto.”
Descripción	“ Los artefactos de SCRUM representan trabajo o valor para proveer transparencia y oportunidades de inspección y adaptación.”
Objetivos	Definidos implícitamente
SC1. Artefactos	

“Artefacto 1. Product Backlog:

El Product Backlog es una lista ordenada de todas las necesidades conocidas en el producto. Es la única fuente de requerimientos y de cualquier cambio a realizar en el producto, El Product Owner es el responsable del Product Backlog, incluyendo su contenido, disponibilidad y ordenamiento.

Un Product Backlog nunca está completo. El desarrollo más temprano del mismo sienta las bases de lo que es inicialmente conocido y los requerimientos mejor entendidos. El Product Backlog evoluciona al igual que el producto y el entorno en el cual será utilizado. El Product Backlog es dinámico; cambia constantemente para identificar lo que el producto necesita para ser apropiado, competitivo y útil. Si existe un producto, existe un Product Backlog.

El Product Backlog lista todas las características, funciones, requerimientos, mejoras y correcciones que constituyen o cambios a ser realizados en el producto en las futuras entregas. Los ítems del Product Backlog tienen los atributos de descripción, orden, estimación y valor, a menudo incluyen descripción de los casos de prueba que proveerán el criterio de cuando están “Hechos”

A medida que un producto que es utilizado gana valor, y el mercado provee retroalimentación, el Product Backlog se vuelve más grande y exhaustivo. Los requerimientos siempre están cambiando y por lo tanto el Product Backlog es un artefacto vivo. Los cambios en la lógica del negocio, las condiciones del mercado o la tecnología también causan cambios en el Product Backlog.

Múltiples Equipos SCRUM a menudo trabajan en el mismo Product Backlog. Un Product Backlog es usado para describir el trabajo por hacer en el producto, los ítems del mismo pueden ser agrupados para que diferentes Equipos SCRUM trabajen en el mismo grupo de ítems.

El refinamiento del Product Backlog es el acto de agregar detalle, estimar y ordenar los ítems, es un proceso en caliente en el cual el Product Owner y el Equipo de Desarrollo colaboran en los detalles de los ítems. El Equipo SCRUM decide cómo y cuando se hace el refinamiento del Product Backlog. A menudo el refinamiento no consume más del 10% de la capacidad del Equipo de Desarrollo. De cualquier manera, los ítems del Product Backlog pueden ser actualizados en cualquier momento por el Product Owner o a discreción del Product Owner.

Los ítems de mayor prioridad en el Product Backlog son usualmente más claros y detallados que los de menor prioridad, estimaciones más precisas son hechas basadas en mayor claridad y detalle; los ítems de menor prioridad son menos detallados. Los ítems del Product Backlog que ocuparán al Equipo de Desarrollo en el siguiente Sprint son refinados de tal manera que puedan ser razonablemente “Hechos” dentro del Sprint. Los ítems del Product Backlog que tienen suficiente detalle y claridad para ser “Hechos” son denominados “Listos” para ser seleccionados en un Sprint Planning. Los ítems del Product Backlog usualmente adquieren este nivel de transparencia a través de las actividades de refinamiento descritas.

El equipo de desarrollo es responsable por todas las estimaciones. El Product Owner puede influenciar al equipo de desarrollo al ayudar a entender y seleccionar intercambios, pero las personas quienes llevarán a cabo el trabajo son quienes hacen la estimación final.”

“ Artefacto 2. Sprint Backlog:

El Sprint Backlog es el conjunto de los ítems seleccionados del Product Backlog para el Sprint, mas un plan para construir el incremento del producto y alcanzar la meta del Sprint. El Sprint Backlog es una visualización del Equipo de Desarrollo sobre la funcionalidad que será implementada en el siguiente incremento y el trabajo necesario para entregar funcionalidad del producto “Hecha”.

El Sprint Backlog hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar la meta del Sprint. Para asegurar la mejora continua, incluye al menos una mejora al proceso de alta prioridad identificada en el Sprint Retrospective previo.

El Sprint Backlog es un plan con suficiente detalle que las tareas en progreso pueden ser entendidos en el Daily SCRUM. El Equipo de Desarrollo modifica el Sprint Backlog a través del Sprint, y el Sprint Backlog emerge durante el Sprint, esto a medida que el Equipo de Desarrollo trabaja en el plan y aprende más sobre el trabajo necesario para alcanzar la meta del Sprint.

Cuando nuevo trabajo aparece, el Equipo de Desarrollo lo agrega al Sprint Backlog y a medida que se completa el trabajo restante es actualizado. Cuando los elementos del plan son declarados innecesarios, son eliminados. Solo el Equipo de Desarrollo puede cambiar su Sprint Backlog durante un Sprint. El Sprint Backlog es altamente visible, es la imagen en tiempo real del trabajo que el Equipo de Desarrollo planea completar durante el Sprint, y pertenece únicamente al Equipo de Desarrollo.”

“ Artefacto 3. Incremento:

El incremento es la suma de todos los ítems del Product Backlog completados durante un Sprint y el valor de los incrementos de los sprints previos. Al final del Sprint, el nuevo incremento debe estar “Hecho”, lo que significa que debe estar en condiciones utilizables y satisfacer la definición de “Hecho” del Equipo SCRUM. Un incremento es un cuerpo inspeccionable, es trabajo terminado que soporta el empirismo al final del Sprint. El incremento es un paso hacia la meta o visión. El incremento debe estar en condiciones utilizables independiente de si el Product Owner decide liberarlo.

Transparencia de los Artefactos SCRUM se basa en transparencia. Las decisiones para optimizar el valor y controlar el riesgo están basadas en el estado percibido de los artefactos. En la medida que la transparencia es completa, estas decisiones tienen fundamentos. En la medida que los artefactos no son transparentes, estas decisiones pueden tener falencias, el valor puede disminuir y el riesgo puede aumentar.

El SCRUM Master debe trabajar con el Product Owner, el Equipo de Desarrollo, y otros involucrados para entender si los artefactos son completamente transparentes. Existen prácticas para tratar la falta de transparencia; el SCRUM Master debe ayudar a todos a aplicar las prácticas en ausencia de total transparencia. Un SCRUM Master puede detectar falta de transparencia al inspeccionar los artefactos, sentir patrones, escuchar de cerca lo que se dice, y detectar diferencias entre los resultados esperados y los reales.

El trabajo del SCRUM Master es trabajar con el Equipo SCRUM ”

“Artefacto 4. Definición de “Hecho”:

El incremento es la suma de todos los ítems del Product Backlog completados durante un Sprint y el valor de los incrementos de los sprints previos. Al final del Sprint, el nuevo incremento debe estar “Hecho”, lo que significa que debe estar en condiciones utilizables y satisfacer la definición de “Hecho” del Equipo SCRUM. Un incremento es un cuerpo inspeccionable, es trabajo terminado que soporta el empirismo al final del Sprint. El incremento es un paso hacia la meta o visión. El incremento debe estar en condiciones utilizables independiente de si el Product Owner decide liberarlo.”

Tabla 34. Homogeneización de los artefactos de SCRUM.

Anexo B. Modelos de Procesos en BPMN

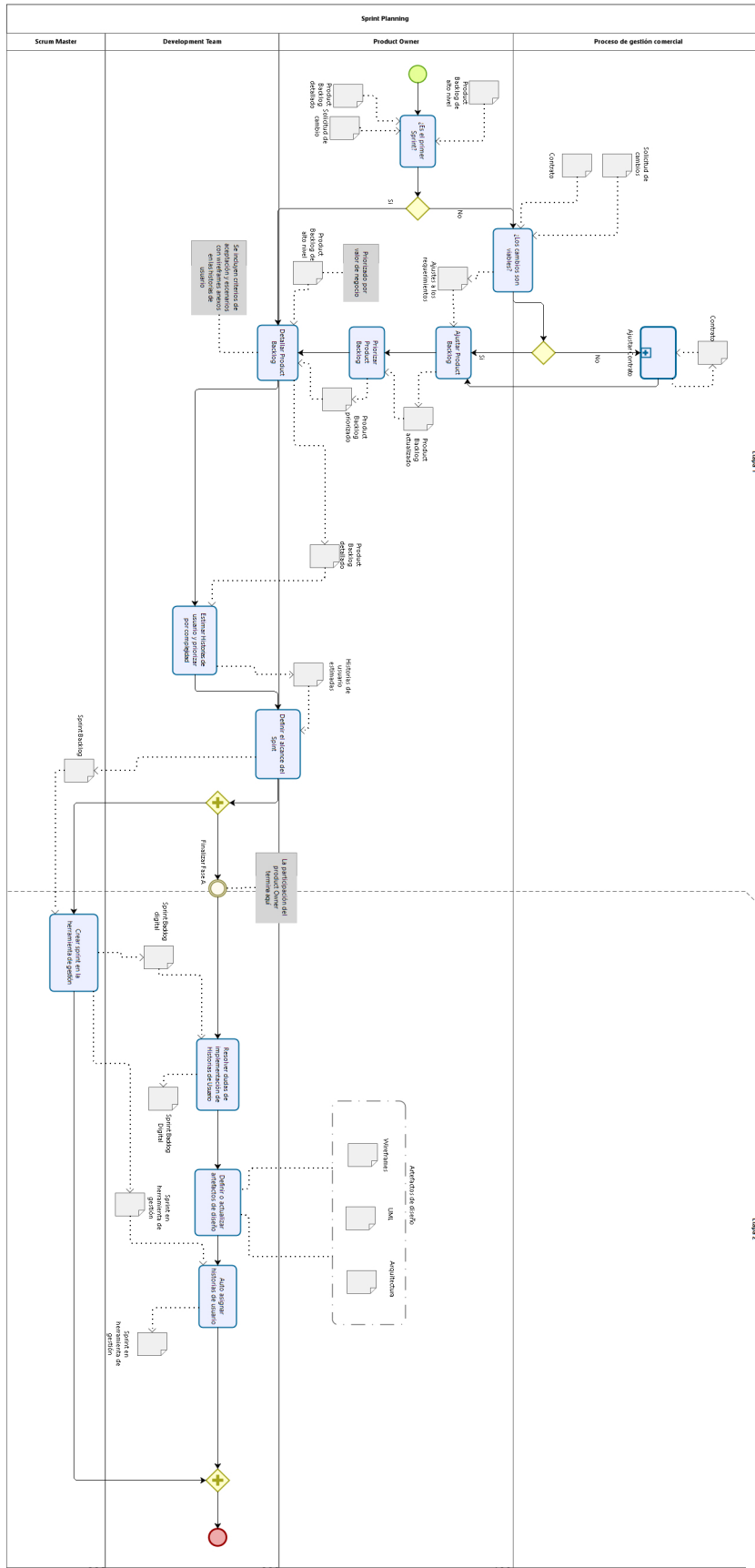


Figura 11. Modelo del proceso *Planeación del Sprint*.

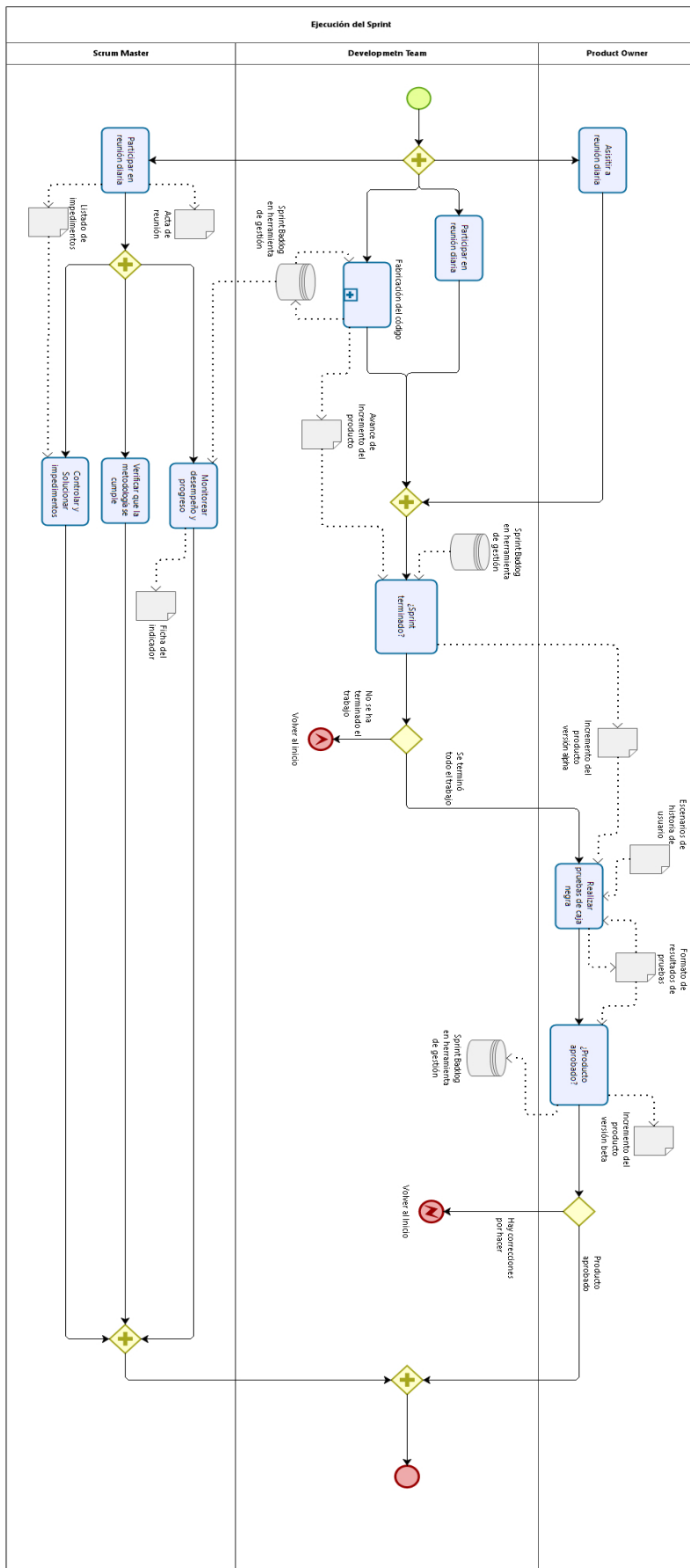


Figura 12. Modelo del proceso *Ejecución del Sprint*.

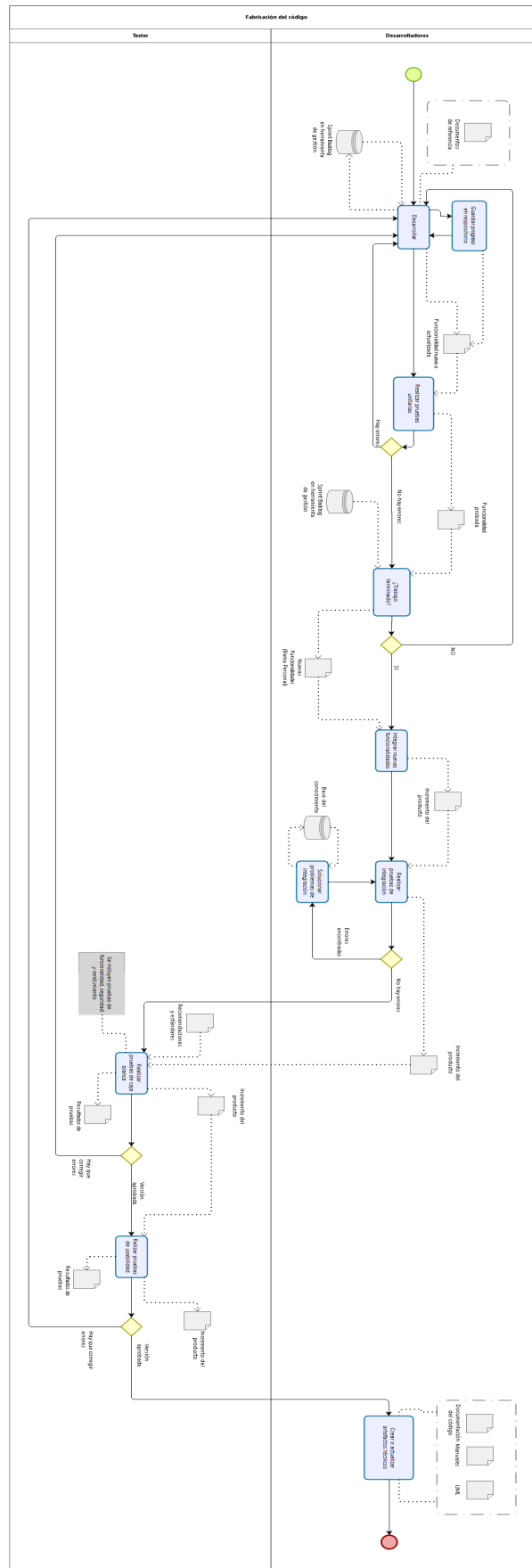


Figura 13. Modelo del proceso *Fabricación del código*.

Anexo C. Relaciones identificadas en la comparación de enfoques

Relación N°	Requisito ISO 9001: 2015	Evento, Rol y/o Artefacto de Scrum	Detalle	Observaciones	ISO 90003
			<p>El sprint planning como etapa de planeación en el ciclo de vida de vida del software hace parte de los procesos necesarios para cumplir con la provisión de los productos y servicios</p> <ul style="list-style-type: none"> * Cumple con el literal b) numeral 2) ya que durante el sprint planning se comunican al equipo de desarrollo los criterios de aceptación para los requisitos que serán implementados en el sprint * Cumple con el literal c) ya que durante el sprint planning como etapa del ciclo de vida del software, se identifican los recursos (humanos, técnicos, entre otros) para implementar los items del product backlog * Cumple con el literal e) ya que se genera como evidencia el sprint backlog 		
re/1	8.1	Sprint Planning	<p>La organización debe definir procesos para cumplir los requisitos para la provisión de productos y servicios, y para implementar las acciones determinadas en el capítulo 6.</p> <ul style="list-style-type: none"> * Cumple con el literal d) ya que la reunión diaria funciona como un control de los procesos puesto que en ella se detectan posibles impedimentos, y se evidencia si las cosas se están haciendo bien o mal * Cumple con el literal b) numeral 2) ya que en la revisión del sprint, el cliente y sus interesados son quienes validan o no que los criterios de aceptación han sido satisfechos * Cumple con el literal d) ya que la revisión del producto por parte del cliente y sus interesados es un control de calidad al proceso * Cumple con el literal e) dejando como evidencia una lista de asistencia a la reunión de revisión del sprint 		
re/2	8.1	Daily Scrum	<ul style="list-style-type: none"> * Cumple con el literal b) numeral 2) ya que en la revisión del sprint, el cliente y sus interesados son quienes validan o no que los criterios de aceptación han sido satisfechos * Cumple con el literal d) ya que la revisión del producto por parte del cliente y sus interesados es un control de calidad al proceso * Cumple con el literal e) dejando como evidencia una lista de asistencia a la reunión de revisión del sprint 		
re/3	8.1	Sprint Review	<ul style="list-style-type: none"> * Cumple con el literal b) numeral 2) ya que en la revisión del sprint, el cliente y sus interesados son quienes validan o no que los criterios de aceptación han sido satisfechos * Cumple con el literal d) ya que la revisión del producto por parte del cliente y sus interesados es un control de calidad al proceso * Cumple con el literal e) dejando como evidencia una lista de asistencia a la reunión de revisión del sprint 		
re/4	8.1	Sprint Retrospective	<ul style="list-style-type: none"> * Cumple con el literal a) ya que el Product Owner es el encargado de la determinación de los requisitos para los productos y servicios * Cumple con el literal b) numeral 2) ya que el Product Owner es quien representa al cliente para determinar los criterios de aceptación de los productos y servicios * Cumple con el literal c) ya que el Product Owner es quien determina los recursos necesarios para la implementación de los items del product backlog, en el contexto del desarrollo de software pueden ser recursos humanos, hardware, software, entre otros. 		
re/5	8.1	Product Owner	<p>El Scrum Master es una parte integral en la planificación y control de los procesos del ciclo de vida del software ya que es el experto en la metodología, el mediador, el facilitador, el coach y solucionador de problemas</p> <ul style="list-style-type: none"> * Cumple con el literal d) ya que el Scrum Master y su conocimiento funcionan como un control al proceso para determinar cuando se están haciendo las cosas bien y cuando hay que ajustar el rumbo * Cumple con el literal e) ya que el Scrum Master vela por la generación, actualización y conservación de todos los artefactos y evidencias adicionales de los procesos 		
re/6	8.1	Scrum Master	<ul style="list-style-type: none"> * Cumple con el literal c) ya que el Equipo de Desarrollo es quien determina los recursos necesarios para la implementación de los items del product backlog, en el contexto del desarrollo de software pueden ser recursos humanos, hardware, software, entre otros. * Cumple con el literal a) puesto que el Product Backlog es el artefacto que contiene todos los requisitos para el producto software * Cumple con el literal b) numeral 2) puesto que cada item del product backlog contiene los criterios de aceptación * Cumple con el literal c) puesto que los recursos necesarios que sean identificados para cumplir con cada item del product backlog deben quedar documentados en cada item * Cumple con el literal d) puesto que es como tal la evidencia y la representación del estado de los requisitos, y si estos han sido implementados, revisados y aprobados 		
re/7	8.1	Development Team	<ul style="list-style-type: none"> * Cumple con el literal d) puesto que la evidencia de los items del product backlog que son trabajados durante un sprint, el sprint backlog es actualizado durante el sprint dependiendo del estado de cada item. Además es la evidencia del trabajo asignado a cada miembro del equipo de desarrollo y contiene la trazabilidad del trabajo realizado * Cumple con el literal b) numeral 2) ya que la Definición de Hecho transveral al Equipo Scrum aclara el concepto de cuando uno de los items del product backlog ha sido terminado, incluyendo criterios de calidad como pruebas del software, aprobación del cliente entre otros. * Cumple con el literal d) ya que funciona como un control a los procesos puesto que si un item del product backlog que ha sido implementado no será declarado como terminado si no es conforme a la definición de "Hecho" 		
re/8	8.1	Product Backlog	<ul style="list-style-type: none"> * Cumple con el literal d) puesto que la evidencia de los items del product backlog que son trabajados durante un sprint, el sprint backlog es actualizado durante el sprint dependiendo del estado de cada item. Además es la evidencia del trabajo asignado a cada miembro del equipo de desarrollo y contiene la trazabilidad del trabajo realizado * Cumple con el literal b) numeral 2) ya que la Definición de Hecho transveral al Equipo Scrum aclara el concepto de cuando uno de los items del product backlog ha sido terminado, incluyendo criterios de calidad como pruebas del software, aprobación del cliente entre otros. * Cumple con el literal d) ya que funciona como un control a los procesos puesto que si un item del product backlog que ha sido implementado no será declarado como terminado si no es conforme a la definición de "Hecho" 		
re/9	8.1	Sprint Backlog	<ul style="list-style-type: none"> * Cumple con el literal d) puesto que la evidencia de los items del product backlog que son trabajados durante un sprint, el sprint backlog es actualizado durante el sprint dependiendo del estado de cada item. Además es la evidencia del trabajo asignado a cada miembro del equipo de desarrollo y contiene la trazabilidad del trabajo realizado * Cumple con el literal b) numeral 2) ya que la Definición de Hecho transveral al Equipo Scrum aclara el concepto de cuando uno de los items del product backlog ha sido terminado, incluyendo criterios de calidad como pruebas del software, aprobación del cliente entre otros. * Cumple con el literal d) ya que funciona como un control a los procesos puesto que si un item del product backlog que ha sido implementado no será declarado como terminado si no es conforme a la definición de "Hecho" 		
re/10	8.1	Definition of Done	<ul style="list-style-type: none"> * Cumple con el literal c) ya que durante la revisión del sprint, el cliente y sus interesados proporcionarían toda la retroalimentación posible con respecto al incremento o avance del producto software desarrollado 		
re/11	8.2.1	Sprint Review			

re/12	8.2.1	Product Owner	<ul style="list-style-type: none"> * Cumple con los literales a), b) y c) ya que el Product Owner es el encargado de entender, recopilar información de, y representar los deseos y necesidades del cliente y sus interesados ante el resto del Equipo Scrum * Cumple con el literal d), ya que la información que el cliente proporcione a la organización para la comprensión y entendimiento de los requisitos, deseos y necesidades es responsabilidad del Product Owner 		
re/13	8.2.1	Product Backlog	<ul style="list-style-type: none"> * Cumple con el literal a) ya que el Product Backlog contiene toda la información de los requisitos, deseos y necesidades del cliente con respecto al producto software * Cumple con el literal b) ya que el Product Backlog es actualizado cuando el cliente solicita cambios en los requisitos y/o funcionalidades del producto software * Cumple con el literal c) puesto que durante la revisión del sprint, la retroalimentación del cliente con respecto al incremento que se expone afecta directamente el estado de los items del Product Backlog * Cumple con el literal d) ya que en los items del Product Backlog que representan los requisitos, deseos y necesidades puede son propiedad del cliente 		
re/14	8.2.1	Increment	<ul style="list-style-type: none"> * Cumple los literales b) y c) Ya que el Incremento (avance de funcionalidad) es la parte del producto en desarrollo que se le expone al cliente para así obtener retroalimentación entre la cual pueden haber cambios o quejas 	Las conexiones de 15 a 19 quedan pendientes ya que el requisito 8.2.2 habla del momento en el que se determinan los requisitos para la provisión de productos y servicios y como tal Scrum no tiene un evento para determinar los requisitos para la provisión de productos y servicios, por tanto es necesario que en los procesos que se definan exista uno exclusivo para la determinación de los requisitos	
re/15	8.2.2	Sprint Planning	<ul style="list-style-type: none"> * Cumple con el literal a) ya que durante la planeación del Sprint, el Product Owner socializa a través del Product Backlog todos los requisitos que potencialmente se van a trabajar durante el sprint, incluyendo todas las observaciones del caso como requisitos legales y reglamentarios y los requisitos de calidad considerados por la organización. Además el Product Owner resuelve cualquier duda que el Equipo de Desarrollo exponga * Cumple con el literal b) ya que durante la socialización de los requisitos o items del Product Backlog, el Equipo de Desarrollo expone cualquier impedimento potencial respecto a la capacidad de implementación y consecuentemente de la provisión del producto o servicio software, es decir que durante la planeación del Sprint se analiza si es posible cumplir con las declaraciones acerca de la provisión del producto o servicio software 		
re/16	8.2.2	Product Owner	<ul style="list-style-type: none"> * Cumple con el literal a) ya que, el Product Owner al ser la interfaz entre el cliente y el Equipo Scrum, es encargado de entender y empoderarse de los deseos y necesidades del cliente, incluyendo los requisitos legales aplicables. Además, debe velar ante el Equipo de Desarrollo por el cumplimiento de los requisitos de calidad que la organización considere necesarios para el producto o servicio software 		
re/17	8.2.2	Development Team	<ul style="list-style-type: none"> * Cumple con el literal b) ya que el Equipo de Desarrollo es quien define si puede cumplir con las declaraciones acerca de la provisión de los productos o servicios software 		
re/18	8.2.2	Product Backlog	<ul style="list-style-type: none"> * Cumple con el literal a) ya que es el artefacto en el que se documentan todos los requisitos incluyendo los reglamentarios y legales aplicables, y los que la organización considere necesarios. 		
re/19	8.2.2	Definition of Done	<ul style="list-style-type: none"> * Cumple con el literal a) ya que en la Definición de Hecho se definen aquellos requisitos legales y reglamentarios aplicables y aquellos que la organización considere necesarios, y si una funcionalidad o item del Product Backlog implementado no cumple con esta definición, no será declarado como terminado 		
re/20	8.2.4	Sprint Planning	<ul style="list-style-type: none"> * Cumple con el requisito ya que los cambios que sean solicitados en las revisiones del sprint, actualizan la información documentada en el Product Backlog durante el Sprint Planning. También ya que a este evento asiste todo el equipo Scrum, todos las personas pertinentes son informadas de los cambios. 		
re/21	8.2.4	Sprint Review	<ul style="list-style-type: none"> * Cumple con el requisito ya que durante este evento es que se generan cambios en los requisitos para los productos y servicios, como consecuencia de las apreciaciones y decisiones del cliente y sus interesados. En el siguiente sprint posterior a la sprint review se ajustan el Product Backlog (Información documentada) de acuerdo a los cambios solicitados 		
re/22	8.2.4	Product Owner	<ul style="list-style-type: none"> * Cumple con el requisito ya que el Product Owner sería el encargado de entender los cambios y de gestionar los cambios al Product Backlog 		
re/23	8.2.4	Scrum Master	<ul style="list-style-type: none"> * Es una de las personas pertinentes a quien se le debe informar de los cambios a los requisitos 		
re/24	8.2.4	Development Team	<ul style="list-style-type: none"> * Es una de las personas pertinentes a quien se le debe informar de los cambios a los requisitos 		
re/25	8.2.4	Product Backlog	<ul style="list-style-type: none"> * Cumple con el requisito ya que es la información documentada 		
re/26	8.3.1	Sprint Planning	<ul style="list-style-type: none"> * Hace parte del proceso de diseño y desarrollo 		
re/27	8.3.1	Daily Scrum	<ul style="list-style-type: none"> * Hace parte del proceso de diseño y desarrollo 		
re/28	8.3.1	Sprint Review	<ul style="list-style-type: none"> * Hace parte del proceso de diseño y desarrollo 		
re/29	8.3.1	Sprint Retrospective	<ul style="list-style-type: none"> * Hace parte del proceso de diseño y desarrollo 		
re/30	8.3.1	Product Owner	<ul style="list-style-type: none"> * Rol del proceso de diseño y desarrollo 		
re/31	8.3.1	Scrum Master	<ul style="list-style-type: none"> * Rol del proceso de diseño y desarrollo 		
re/32	8.3.1	Development Team	<ul style="list-style-type: none"> * Rol del proceso de diseño y desarrollo 		

re/33	8.3.1	Product Backlog	* Artefacto del proceso de diseño y desarrollo	
re/34	8.3.1	Sprint Backlog	* Artefacto del proceso de diseño y desarrollo	
re/35	8.3.1	Increment	* Artefacto del proceso de diseño y desarrollo	
re/36	8.3.1	Definition of Done	* Artefacto del proceso de diseño y desarrollo	
			* Cumple con el literal a) ya que el sprint planning es un evento de naturaleza iterativa, con una duración fija basada en la duración del sprint, es decir a mayor duración del sprint, más complejidad de desarrollo, más planeación necesaria.	
			* Cumple con el literal f) ya que al sprint planning de las personas que participan activamente en el proceso de diseño y desarrollo solo deben los miembros del equipo Scrum.	
re/37	8.3.2	Sprint Planning	* Cumple con el literal h) ya que el sprint planning naturalmente se enfoca en los requisitos del Product Backlog	
			* Cumple con el literal a) ya que el daily scrum es una reunión muy corta y ágil para identificar posibles impedimentos.	
			* Cumple con el literal b) ya que funciona como una revisión del desarrollo	
			* Cumple con el literal c) al ser una actividad de verificación del desarrollo	
re/38	8.3.2	Daily Scrum	* Cumple con el literal f) ya que en esta reunión, aunque puede asistir el Product Owner solo participan el Scrum Master y el Development Team	
			* Cumple con los literales a), b), c) y g) ya que la naturaleza iterativa de la revisión del sprint, a la que asisten el cliente y sus interesados permite verificar y validar que los requisitos se están cumpliendo correctamente y en su totalidad, tiene una duración fija proporcional a la duración del sprint, es decir que a mayor duración del sprint, más complejidad de desarrollo, más extensa la revisión.	
re/39	8.3.2	Sprint Review	* Cumple con el literal a) ya que la naturaleza iterativa del sprint retrospective permite mejorar auto evaluar y mejorar el proceso de diseño y desarrollo, y tiene una duración fija	
re/40	8.3.2	Sprint Retrospective	* Cumple con el literal d) ya que es uno de los roles del proceso con responsabilidades específicas	
			* Cumple con el literal f) ya que es la interfaz entre el cliente y el Scrum Team, el Product Owner es el representante y vocero del cliente ante el Development Team y el Scrum Master	
re/41	8.3.2	Product Owner	* Cumple con el literal d) ya que es uno de los roles del proceso con responsabilidades específicas	
			* Cumple con el literal f) ya que es quien se encarga de que el Development Team pueda hacer su trabajo sin inconvenientes y sin intervención de factores externos.	
re/42	8.3.2	Scrum Master	* Cumple con el literal d) ya que es uno de los roles del proceso con responsabilidades específicas	
re/43	8.3.2	Development Team	* Cumple con el literal c) ya que es el artefacto que permite gestionar la verificación del diseño y desarrollo	
			* Cumple con el literal h) ya que es el artefacto que permite gestionar los requisitos para la provisión de los productos y servicios	
re/44	8.3.2	Product Backlog	* Cumple con el literal j) ya que es información documentada	
			* Cumple con el literal c) ya que es la información que permite gestionar la verificación y validación del diseño y desarrollo	
re/45	8.3.2	Sprint Backlog	* Cumple con el literal j) ya que es información documentada	
re/46	8.3.2	Increment	* Hace parte del proceso de desarrollo ya que es el producto que se debe revisar, verificar y validar	
re/47	8.3.2	Definition of Done	* Cumple con el literal d) ya que la 'Definition of Done' permite saber qué etapas y a cargo de que roles deben haberse superado para considerar un requisito del Product Backlog como 'Done'	
			* Cumple con el literal a) ya que el durante el Sprint Planning se analizan y detallan en profundidad los items del product backlog	
			* Cumple con el literal b) ya que con la información de los sprints anteriores se obtiene la velocidad del equipo, que permite determinar cuánta complejidad puede implementar el equipo de desarrollo en términos de funcionalidades durante un Sprint y de la misma manera se define el alcance del sprint.	
re/48	8.3.3	Sprint Planning	Durante el Sprint Planning se asegura que los items del Product Backlog que serán implementados en el sprint han sido claramente entendidos y no tienen ambigüedades	
			* Cumple con el literal a) ya que el Product Owner es el encargado de la determinación de los requisitos para los productos y servicios	
			* Cumple con el literal c) ya que el Product Owner al ser el vocero y representante del cliente ante el equipo Scrum, conoce los requisitos legales y reglamentarios para el producto que se está construyendo	
			* Cumple con el literal d) ya que el Product Owner conoce y es responsable de que se implementen las normas o códigos de prácticas para el producto que se está construyendo	
re/49	8.3.3	Product Owner		

re/60	8.3.3	Product Backlog	<ul style="list-style-type: none"> * Cumple con el literal a), c) y d) ya que el Product Backlog contiene toda la información de los requisitos, deseos y necesidades del cliente con respecto al producto software 		
re/61	8.3.3	Sprint Backlog	<ul style="list-style-type: none"> * Cumple con el literal a), c) y d) ya que el Sprint Backlog es el conjunto de items del Product Backlog que será implementado en el Sprint. * Los items que están en el Sprint Backlog están listos para ser implementados y libres de ambigüedades 		
re/62	8.3.3	Increment	<ul style="list-style-type: none"> * Es información documentada para verificar y validar que los requisitos o entradas para el diseño y desarrollo han sido abordados en su totalidad * Cumple con el literal a), c) y d) ya que la Definition of Done considera todos los criterios para determinar que un item del Sprint Backlog está "Terminado" al incluir requisitos de desempeño, legales y reglamentarios aplicables y las normas o códigos de prácticas que la organización se haya comprometido a implementar 		
re/63	8.3.3	Definition of Done	<ul style="list-style-type: none"> * Cumple con el literal a) ya que en el Sprint Planning se definen los resultados a lograr durante el Sprint a través del alcance del Sprint o bien los items seleccionados del Product Backlog que conforman el Sprint Backlog * Cumple con el literal e) ya que cualquier cambio o corrección que sea obtenido como retroalimentación en la reunión Sprint Review es abordado y analizado en el Sprint Planning 		
re/64	8.3.4	Sprint Planning	<ul style="list-style-type: none"> * Cumple con los literales b) y c) ya que durante esta reunión es cuando se verifica que todos los items que conformaban el alcance del sprint han sido abordados y además que el producto resultante satisface los requisitos. En esta reunión participa todo el equipo Scrum, el cliente y sus interesados; siendo los últimos quienes verifican y validan el producto resultante (increment) * Cumple con el literal e), ya que cualquier acción necesaria sobre los inconvenientes o inconformidades es obtenida en esta revisión 		
re/55	8.3.4	Sprint Review	<ul style="list-style-type: none"> * Participa activamente del Sprint Review al documentar todos los comentarios del cliente y sus interesados 		
re/66	8.3.4	Product Owner	<ul style="list-style-type: none"> * Participa activamente del Sprint Review, verificando que se cumple el tiempo estimado para la reunión, que todos asistan, que la reunión se haga de acuerdo a lo que Scrum establece, y que cualquier retroalimentación por parte del cliente y/o sus interesados se documente para posterior análisis 		
re/67	8.3.4	Scrum Master	<ul style="list-style-type: none"> * Participa activamente del Sprint Review, presentando el producto resultante del Sprint o Increment 		
re/58	8.3.4	Development Team	<ul style="list-style-type: none"> * Cumple con el literal f) ya que es información documentada que es actualizada de acuerdo a los resultados del Sprint Review 		
re/59	8.3.4	Product Backlog	<ul style="list-style-type: none"> * Cumple con el literal f) ya que es información documentada que es actualizada de acuerdo a los resultados del Sprint Review 		
re/60	8.3.4	Sprint Backlog	<ul style="list-style-type: none"> * Es el producto resultante del sprint, es el objeto de la verificación y validación contra las entradas para el diseño y desarrollo. 		
re/61	8.3.4	Increment	<ul style="list-style-type: none"> * En la Definition of Done, se puede establecer que un item del Product Backlog no está terminado hasta que no ha sido verificado y validado por el cliente y sus interesados 		
re/62	8.3.4	Definition of Done	<ul style="list-style-type: none"> * Cumple con el literal a), b), c) y d) ya que en esta revisión el cliente y sus interesados verifican y validan el cumplimiento de los requisitos de las entradas del diseño y desarrollo, además se verifica que el incremento presentado es adecuado para la posterior construcción del producto de manera iterativa e incremental, se verifican que se cumplen los criterios de aceptación. Y además se especifican las características que son esenciales para el propósito previsto del producto y su provisión segura y correcta (en el contexto del software, estas características pueden ser por ejemplo: tiempos de respuesta, concurrencia, seguridad de la información entre otros) 		
re/63	8.3.5	Sprint Review	<ul style="list-style-type: none"> * Cumple con el literal a), c) y d) ya que el Product Owner es el representante del cliente ante el equipo Scrum, por tanto es el responsable de la satisfacción del cliente entendiendo el contexto del negocio y sus necesidades. Así mismo el Product Owner debe asegurarse que el producto cumple con todos los criterios de aceptación incluyendo las características específicas esenciales para el propósito previsto y la provisión segura y correcta. 		
re/64	8.3.5	Product Owner	<ul style="list-style-type: none"> * Cumple con el literal b) ya que el Scrum Master puede guiar al equipo de desarrollo con respecto a los incrementos resultantes y la forma en que se sigue construyendo el producto. Además el proceso de desarrollo debe incluir fases de pruebas durante el sprint y el Scrum Master debe asegurarse que los procesos definidos se siguen y generan las salidas correspondientes como evidencias de resultados de pruebas entre otros. 		
re/65	8.3.5	Scrum Master			

re/66	8.3.5	Development Team	<ul style="list-style-type: none"> * Cumple con el literal a) ya que el Product Backlog es una entrada para el diseño y desarrollo * Cumple con el literal c) ya que cada ítem del Product Backlog incluye los requisitos de seguimiento y medición (pruebas) Y los criterios de aceptación * Cumple con el literal d) ya que cada ítem del Product Backlog tiene especificadas las características esenciales para su propósito previsto Y su privisión segura y correcta 	<ul style="list-style-type: none"> * Cumple con el literal a) ya que el Sprint Backlog es un artefacto del diseño y desarrollo * Cumple con el literal c) ya que cada ítem del Sprint Backlog incluye los requisitos de seguimiento y medición (pruebas) Y los criterios de aceptación * Cumple con el literal d) ya que cada ítem del Sprint Backlog tiene especificadas las características esenciales para su propósito previsto Y su privisión segura y correcta 		
re/67	8.3.5	Product Backlog	<ul style="list-style-type: none"> * Cumple con el literal a) ya que el Sprint Backlog es un artefacto del diseño y desarrollo * Cumple con el literal c) ya que cada ítem del Sprint Backlog incluye los requisitos de seguimiento y medición (pruebas) Y los criterios de aceptación * Cumple con el literal d) ya que cada ítem del Sprint Backlog tiene especificadas las características esenciales para su propósito previsto Y su privisión segura y correcta 	<ul style="list-style-type: none"> * Cumple con el literal a) ya que el Sprint Backlog es un artefacto del diseño y desarrollo * Cumple con el literal c) ya que cada ítem del Sprint Backlog incluye los requisitos de seguimiento y medición (pruebas) Y los criterios de aceptación * Cumple con el literal d) ya que cada ítem del Sprint Backlog tiene especificadas las características esenciales para su propósito previsto Y su privisión segura y correcta 		
re/68	8.3.5	Sprint Backlog	<ul style="list-style-type: none"> * Funciona como un control a las salidas del diseño y desarrollo ya que no serán aceptadas hasta que cumplan la Definition of Done de la organización 	<ul style="list-style-type: none"> * Funciona como un control a las salidas del diseño y desarrollo ya que no serán aceptadas hasta que cumplan la Definition of Done de la organización 		
re/69	8.3.5	Increment	<ul style="list-style-type: none"> * Cumple con el literal b) c) y d) ya que en el Sprint Planning se abordan los cambios que hayan sido identificados en la última revisión de sprint, en este evento se reúne el Scrum Team para revisar los cambios, analizarlos, autorizarlos y minimizar los impactos adversos 	<ul style="list-style-type: none"> * Cumple con el literal b) c) y d) ya que en el Sprint Planning se abordan los cambios que hayan sido identificados en la última revisión de sprint, en este evento se reúne el Scrum Team para revisar los cambios, analizarlos y minimizar los impactos adversos 		
re/70	8.3.5	Definition of Done	<ul style="list-style-type: none"> * Funciona como un control a las salidas del diseño y desarrollo ya que no serán aceptadas hasta que cumplan la Definition of Done de la organización 	<ul style="list-style-type: none"> * Funciona como un control a las salidas del diseño y desarrollo ya que no serán aceptadas hasta que cumplan la Definition of Done de la organización 		
re/71	8.3.6	Sprint Planning	<ul style="list-style-type: none"> * Durante el Sprint Review es cuando el Development Team presenta la salida 'Increment' del Sprint recién terminado, Y es en este evento que el cliente Y sus interesados deciden si el incremento presentado puede salir a producción 	<ul style="list-style-type: none"> * Durante el Sprint Review es cuando el Development Team presenta la salida 'Increment' del Sprint recién terminado, Y es en este evento que el cliente Y sus interesados deciden si el incremento presentado puede salir a producción 		
re/72	8.3.6	Sprint Review	<ul style="list-style-type: none"> * El Product Owner es el principal rol con respecto a comunicar los cambios que el cliente desea, Y tiene responsabilidad en la autorización de los cambios Y las acciones tomadas para prevenir impactos adversos 	<ul style="list-style-type: none"> * El Product Owner es el principal rol con respecto a comunicar los cambios que el cliente desea, Y tiene responsabilidad en la autorización de los cambios Y las acciones tomadas para prevenir impactos adversos 		
re/73	8.3.6	Product Owner	<ul style="list-style-type: none"> * Es la información documentada ya que cada ítem del Product Backlog que presente cambios deberá tener su historial de modificaciones, revisiones y aprobaciones 	<ul style="list-style-type: none"> * Es la información documentada ya que cada ítem del Product Backlog que presente cambios deberá tener su historial de modificaciones, revisiones y aprobaciones 		
re/74	8.3.6	Product Backlog	<ul style="list-style-type: none"> * El Product Owner debe asegurarse de que se genere información documentada con respecto al incremento que el cliente ha aprobado para sacar a producción, cada incremento debe tener identificación única con respecto al Sprint en el que fue producido Y el conjunto de ítems del Product Backlog que tiene implementados Y terminados de acuerdo a la 'Definition of Done' 	<ul style="list-style-type: none"> * El Product Owner debe asegurarse de que se genere información documentada con respecto al incremento que el cliente ha aprobado para sacar a producción, cada incremento debe tener identificación única con respecto al Sprint en el que fue producido Y el conjunto de ítems del Product Backlog que tiene implementados Y terminados de acuerdo a la 'Definition of Done' 		
re/75	8.5.2	Sprint Review	<ul style="list-style-type: none"> * El Scrum Master es responsable de que el proceso de desarrollo definido se siga correctamente Y que todos los recursos para el mismo estén disponibles Y sean utilizados correctamente, entre ellos el Scrum Master debe asegurarse que los medios apropiados (herramienta de gestión de proyectos de software) sean utilizados para identificar cada uno de los ítems del Product Backlog, su estado, su implementación durante un Sprint, su aprobación durante un Sprint Review entre otros. 	<ul style="list-style-type: none"> * El Scrum Master es responsable de que el proceso de desarrollo definido se siga correctamente Y que todos los recursos para el mismo estén disponibles Y sean utilizados correctamente, entre ellos el Scrum Master debe asegurarse que los medios apropiados (herramienta de gestión de proyectos de software) sean utilizados para identificar cada uno de los ítems del Product Backlog, su estado, su implementación durante un Sprint, su aprobación durante un Sprint Review entre otros. 		
re/76	8.5.2	Product Owner	<ul style="list-style-type: none"> * Si bien el Scrum Master es responsable de asegurarse que las salidas o incrementos sean correctamente identificados, el Development Team es quien debe llevar a cabo la gestión de los ítems del Product Backlog en la herramienta de gestión de proyectos de software de tal forma que en los momentos apropiados los estados de los ítems del Product Backlog sean actualizados, incluyendo todos los criterios de la Definition of Done 	<ul style="list-style-type: none"> * Si bien el Scrum Master es responsable de asegurarse que las salidas o incrementos sean correctamente identificados, el Development Team es quien debe llevar a cabo la gestión de los ítems del Product Backlog en la herramienta de gestión de proyectos de software de tal forma que en los momentos apropiados los estados de los ítems del Product Backlog sean actualizados, incluyendo todos los criterios de la Definition of Done 		
re/77	8.5.2	Scrum Master	<ul style="list-style-type: none"> * El Product Backlog es el conjunto de requisitos para el producto software, es la información documentada con sus respectivos identificadores de ítems, de estados, de aprobaciones, historial Y control de cambios, cumplimiento de los criterios de la Definition of Done entre otros 	<ul style="list-style-type: none"> * El Product Backlog es el conjunto de requisitos para el producto software, es la información documentada con sus respectivos identificadores de ítems, de estados, de aprobaciones, historial Y control de cambios, cumplimiento de los criterios de la Definition of Done entre otros 		
re/78	8.5.2	Development Team	<ul style="list-style-type: none"> * El Sprint Backlog es el conjunto de requisitos para el producto software que se implementan durante un Sprint, contiene el estado de cada ítem implementado incluyendo los estados de las pruebas de software necesarias realizadas Y el cumplimiento de los criterios de la Definition of Done 	<ul style="list-style-type: none"> * El Sprint Backlog es el conjunto de requisitos para el producto software que se implementan durante un Sprint, contiene el estado de cada ítem implementado incluyendo los estados de las pruebas de software necesarias realizadas Y el cumplimiento de los criterios de la Definition of Done 		
re/79	8.5.2	Product Backlog	<ul style="list-style-type: none"> * Es el objeto de la identificación ya que es la salida que se puede o no sacar a producción al final de un Sprint 	<ul style="list-style-type: none"> * Es el objeto de la identificación ya que es la salida que se puede o no sacar a producción al final de un Sprint 		
re/80	8.5.2	Sprint Backlog				
re/81	8.5.2	Increment				

				<ul style="list-style-type: none"> * Durante el Sprint Review es cuando el Development Team presenta la salida 'Increment' del Sprint recién terminado, y es en este evento que el cliente y sus interesados deciden si el incremento presentado puede salir a producción. El incremento que sale a Producción es una salida que se preserva a través de gestión de la configuración incluyendo el almacenamiento del código, la identificación (versionado) entre otros.
re/82	8.5.4	Sprint Review		
re/83	8.5.4	Product Owner		
re/84	8.5.4	Scrum Master		<ul style="list-style-type: none"> * El Scrum Master debe asegurarse que el Development Team hace una correcta gestión de la configuración de tal forma que los incrementos sean preservados
re/85	8.5.4	Development Team		<ul style="list-style-type: none"> * El Development Team debe hacer una correcta gestión de la configuración para que los incrementos sean preservados
re/86	8.5.4	Product Backlog		<ul style="list-style-type: none"> * El Product Backlog contiene los items que representan los requisitos del producto software, cada item debe ser identificado y durante el desarrollo la gestión de la configuración puede ayudar a preservar el código fuente de cada item implementado a través de sistemas de control de versiones
re/87	8.5.4	Sprint Backlog		<ul style="list-style-type: none"> * El Sprint Backlog contiene los items del Product Backlog que han sido implementados en un Sprint y por tanto los items que conforman el nuevo incremento, estos items deben ser identificados y preservados una vez que cumplan con la Definition of Done
re/88	8.5.4	Increment		<ul style="list-style-type: none"> * Es el objeto de la preservación
re/89	8.6	Sprint Review		<ul style="list-style-type: none"> * Cumple con el literal a) ya que durante el Sprint Review, el cliente y sus interesados verifican y validan la conformidad con los criterios de aceptación.
re/90	8.6	Product Owner		<ul style="list-style-type: none"> * Cumple con el literal b) ya que es el directo responsable de autorizar la liberación, como representante del cliente ante el equipo Scrum. Además el Product Owner debe asegurarse que cada item del Product Backlog que se ha implementado en el incremento ha sido declarado como 'Done'
re/91	8.6	Scrum Master		<ul style="list-style-type: none"> * El Scrum Master también es responsable que cada item del Product Backlog que ha sido implementado en un incremento haya sido declarado como 'Done'
re/92	8.6	Product Backlog		<ul style="list-style-type: none"> * Es la información documentada ya que el Product Backlog contiene todos los items que representan los requisitos del cliente, y además los atributos relacionados con la aprobación, verificación y validación de que se cumplen los criterios de aceptación
re/93	8.6	Sprint Backlog		<ul style="list-style-type: none"> * Es la información documentada ya que el Sprint Backlog contiene los items implementados durante el Sprint, y cada item tiene los atributos relacionados con la aprobación, verificación y validación de que se cumplen los criterios de aceptación
re/94	8.6	Increment		<ul style="list-style-type: none"> * Es el producto que puede ser liberado si así lo deciden el cliente y sus interesados
re/95	8.6	Definition of Done		<ul style="list-style-type: none"> * Es el concepto que cada item del Product Backlog y Sprint Backlog debe cumplir para ser declarado como 'Done' o 'Terminado' incluyendo todas las posibles aprobaciones, validaciones, pruebas de software, seguridad de la información entre otros.
re/96	8.7.1	Sprint Planning		<ul style="list-style-type: none"> * Durante el Sprint Planning se pueden llevar a cabo análisis y planeación de correcciones a salidas no conformes identificadas en el Sprint Review inmediatamente anterior
re/97	8.7.1	Sprint Review		<ul style="list-style-type: none"> * Durante el Sprint Review, el cliente y sus interesados identifican salidas no conformes en el incremento del Sprint terminado, si existen no conformidades, estas serán corregidas durante el Sprint posterior
re/98	8.7.1	Product Owner		<ul style="list-style-type: none"> * El Product Owner al ser el representante del Cliente debe llevar control y trazabilidad de las salidas no conformes identificadas por el cliente y sus interesados y a su vez debe llevar control de que las salidas no conformes son corregidas y verificadas y validadas nuevamente
re/99	8.7.1	Scrum Master		<ul style="list-style-type: none"> * El Scrum Master es responsable de que el proceso de desarrollo definido se siga correctamente, entre las actividades del mismo deben existir las respectivas pruebas de software, por ejemplo pruebas de caja blanca y caja negra, pruebas de desempeño, pruebas de usabilidad, pruebas de seguridad entre otros; de tal manera que todos los items del Product Backlog o funcionalidades que no pasen estas pruebas sean declaradas como salidas no conformes y sean tratadas como tal
re/100	8.7.1	Development Team		<ul style="list-style-type: none"> * El Development Team debe incluir Testers que lleven a cabo pruebas de software a los items del Sprint Backlog implementados por los desarrolladores, y deben dejar información documentada de los resultados de las pruebas para que aquellos items que no pasan las pruebas sean tratados como salidas no conformes y se les hagan las respectivas correcciones
re/101	8.7.1	Product Backlog		<ul style="list-style-type: none"> * El Product Backlog contiene los items que representan los requisitos del producto software, cada item debe ser identificado y debe ser sometido a las pruebas de software correspondientes cuando es implementado (en el Sprint Backlog) y debe guardar información documentada de que la funcionalidad aprobó las pruebas de software, o dado el caso de no aprobó las pruebas de software debe tener información documentada de la corrección y la posterior ejecución de pruebas nuevamente.
re/102	8.7.1	Sprint Backlog		<ul style="list-style-type: none"> * El Sprint Backlog contiene los items del Product Backlog que son implementados durante el Sprint, cada item debe ser identificado y conservar información documentada de las pruebas de software realizadas, su aprobación o reprobación y cuando sea necesario su corrección y re ejecución de las pruebas de software

re/103	8.7.1	Definition of Done	<ul style="list-style-type: none"> * La Definition of Done es esencial para identificar o diferenciar las salidas que son conformes o no conformes, ya que un todos los items del Product Backlog que son implementados en cada sprint son salidas no conformes hasta que sean declarados como Done. Por ejemplo la Definition of Done puede tener como restricciones que cada item está "terminado" si ha aprobado las pruebas de software, si el cliente lo ha validado, si se cumplen todos los criterios de aceptación entre otros 		
re/104	8.7.2	Daily Scrum	<ul style="list-style-type: none"> * Durante el Daily Scrum, el Scrum Master y el Development Team se ponen al día con respecto a las funcionalidades o items del Sprint Backlog que no pasan las pruebas de software (salidas no conformes), el Scrum Master debe asegurarse que las pruebas son documentadas y que las acciones necesarias son tomadas para corregir los errores encontrados. 	Esta conexión fue eliminada porque ya se explica en el Scrum Master como rol, y no necesariamente está relacionada con Daily Scrum.	
re/105	8.7.2	Sprint Review	<ul style="list-style-type: none"> * Durante el Sprint Review el cliente y sus interesados son quienes identifican salidas no conformes una vez que el Sprint ha sido terminado, el Product Owner debe conservar información documentada de la retroalimentación del cliente y debe asegurarse que las correcciones necesarias sean realizadas posteriormente 		
re/106	8.7.2	Product Owner	<ul style="list-style-type: none"> * El Product Owner como encargado del éxito del producto y de la gestión del Product Backlog debe asegurarse que la información documentada respecto a las pruebas de software, y salidas no conformes de cada item se conserva de manera correcta e íntegra. Además de que las correcciones necesarias sean realizadas * El Scrum Master es responsable de que el proceso de desarrollo definido se siga correctamente, entre las actividades del mismo deben existir las respectivas pruebas de software, por ejemplo pruebas de caja blanca y caja negra, pruebas de desempeño, pruebas de usabilidad, pruebas de seguridad entre otros; de tal manera que todos los items del Product Backlog o funcionalidades que no pasen estas pruebas sean declaradas como salidas no conformes y sean tratadas como tal 		
re/107	8.7.2	Scrum Master	<ul style="list-style-type: none"> * El Development Team debe incluir Testers que lleven a cabo pruebas de software a los items del Sprint Backlog implementados por los desarrolladores, y deben dejar información documentada de los resultados de las pruebas para que aquellos items que no pasan las pruebas sean tratados como salidas no conformes y se les hagan las respectivas correcciones 		
re/108	8.7.2	Development Team	<ul style="list-style-type: none"> * Cada item del Product Backlog debe tener como anexo los resultados satisfactorios de las pruebas de software realizadas, en caso de que haya fallado las pruebas debe tener evidencia de las correcciones y las segundas pruebas satisfactorias, o las que sean necesarias hasta que el item se declara como "Terminado". 		
re/109	8.7.2	Product Backlog	<ul style="list-style-type: none"> * Cada item del Sprint Backlog debe tener como anexo los resultados satisfactorios de las pruebas de software realizadas, en caso de que haya fallado las pruebas debe tener evidencia de las correcciones y las segundas pruebas satisfactorias, o las que sean necesarias hasta que el item se declara como "Terminado". 		
re/110	8.7.2	Sprint Backlog	<ul style="list-style-type: none"> * El Sprint Retrospective es un momento adecuado para llevar a cabo actividades de evaluación del desempeño, seguimiento y medición ya que durante este evento el equipo Scrum evalúa cómo fue el desempeño durante el Sprint que acaba de terminar y las oportunidades de mejora que se pueden abordar 	Aunque es cierto lo mencionado, para el requisito 9.1.1 no aplica la retrospectiva ya que en este momento no se está haciendo seguimiento y medición, si no analizando y evaluando	
re/111	9.1.1	Sprint Retrospective	<ul style="list-style-type: none"> * El Scrum Master es un rol apropiado para encargarse de gestionar el seguimiento y medición del proceso de diseño y desarrollo. ISO 90003 provee diferentes indicadores que el desarrollo de software ofrece como evidencia para el cumplimiento de los requisitos * Hay gran variedad de métodos de seguimiento, medición y análisis para Scrum con indicadores como la velocidad del equipo y burndown chart. La organización debe encargarse de seleccionar los indicadores y métricas apropiadas y el Scrum Master puede gestionar la documentación * El seguimiento y la medición deberían llevarse a cabo durante la ejecución del Sprint 	<ul style="list-style-type: none"> * La duración planeada y real de las actividades del proceso * El costo planeado y real de las actividades del proceso * Los niveles de calidad planeados y medidas progresivas de las características de calidad seleccionadas 	El Scrum Master puede encargarse de la medición a utilizando métricas propuestas por ISO 90003.
re/112	9.1.1	Scrum Master	<ul style="list-style-type: none"> * El Development Team hace parte de lo que se debe seguir y medir junto al producto software que constituye 		
re/113	9.1.1	Development Team	<ul style="list-style-type: none"> * Cada item del Product Backlog debe tener la respectiva estimación de complejidad realizada por el Development Team, se debe hacer seguimiento de la estimación inicial y la estimación real de cada item del Product Backlog. 		
re/114	9.1.1	Product Backlog	<ul style="list-style-type: none"> * Cada item del Sprint Backlog debe tener la respectiva estimación de complejidad realizada por el Development Team, se debe hacer seguimiento a la estimación inicial y la estimación real de cada item del Sprint Backlog; además de su duración en horas hombre estimada y su duración en horas hombre real 		
re/115	9.1.1	Sprint Backlog	<ul style="list-style-type: none"> * Cumple con el requisito, ya que durante el Sprint Review se garantiza que el Producto Software que se está construyendo, satisface los deseos y necesidades del cliente. Si el cliente o sus interesados tiene alguna duda, inquietud o incomodidad se toma nota para resolverla o bien para ajustar el Producto de tal manera que se maximize la satisfacción del cliente 		
re/116	9.1.2	Sprint Review	<ul style="list-style-type: none"> * El Product Owner es el director responsable de la satisfacción del cliente ya que es el representante del cliente y sus interesados ante el Scrum Team 		
re/117	9.1.2	Product Owner			

re/118	9.1.3	Sprint Review	<ul style="list-style-type: none"> * Durante el Sprint Review, es posible evaluar la conformidad de los productos y servicios como resultado del análisis de la retroalimentación del cliente y sus interesados, al igual que la satisfacción de los mismos. * Durante el Sprint Retrospective se lleva a cabo una autoevaluación del desempeño del equipo Scrum en el Sprint que acaba de terminar. La autoevaluación permite identificar oportunidades de mejora y consecuentemente la mejora del sistema de gestión de la calidad. 	
re/119	9.1.3	Sprint Retrospective	<ul style="list-style-type: none"> * El Product Owner es el responsable de la conformidad de los productos y servicios y de la satisfacción del cliente 	
re/120	9.1.3	Product Owner	<ul style="list-style-type: none"> * El Scrum Master es el responsable de que lo planeado se implemente de forma eficaz, y responsable de que las reuniones de retrospectiva 'Sprint Retrospective' se lleven a cabo de forma adecuada, identificando y abordando las oportunidades de mejora 	
re/121	9.1.3	Scrum Master	<ul style="list-style-type: none"> * El equipo de desarrollo, además del Product Owner y Scrum Master también es responsable de la conformidad de los productos y servicios al garantizar que los incrementos implementados cumplan los requisitos de las entradas, y de implementar lo planeado de manera eficaz 	
re/122	9.1.3	Development Team	<ul style="list-style-type: none"> * El Product Backlog contiene información valiosa sobre lo que se ha planeado y si se ha implementado de manera eficaz, además de información de la conformidad de cada ítem con los requisitos de las entradas 	
re/123	9.1.3	Product Backlog	<ul style="list-style-type: none"> * El Sprint Backlog contiene información valiosa sobre lo que se ha planeado y si se ha implementado de manera eficaz, además de información de la conformidad de cada ítem con los requisitos de las entradas 	
re/124	9.1.3	Sprint Backlog	<ul style="list-style-type: none"> * Las reuniones Sprint Review proporcionan la información más importante sobre la satisfacción del cliente y la retroalimentación de las partes interesadas, además de la conformidad de los productos y servicios 	
re/125	9.3.2	Sprint Review	<ul style="list-style-type: none"> * El Product Owner como directo responsable de la satisfacción del cliente debe estar presente en la revisión por la dirección 	
re/126	9.3.2	Product Owner	<ul style="list-style-type: none"> * El Product Owner como directo responsable de la eficacia del proceso de diseño y desarrollo debe estar presente en la revisión por la dirección, incluyendo su responsabilidad con respecto al desempeño de los procesos y las oportunidades de mejora 	
re/127	9.3.2	Scrum Master	<ul style="list-style-type: none"> * Proporcionan evidencia de la satisfacción del cliente, conformidad con los requisitos y eficacia de implementación de lo planeado a lo largo de todo el proyecto 	
re/128	9.3.2	Product Backlog	<ul style="list-style-type: none"> * Proporcionan evidencia de la satisfacción del cliente, conformidad con los requisitos y eficacia de implementación de lo planeado en cada sprint 	
re/129	9.3.2	Sprint Backlog	<ul style="list-style-type: none"> * Cuando ha ocurrido una no conformidad que fue detectada por el cliente y sus interesados en un Sprint Review, El Sprint Planning es el evento donde se reacciona a la no conformidad, ya que ahí se toman acciones para controlar y corregir la no conformidad, además de evaluar la necesidad de acciones para eliminar las causas de la no conformidad 	
re/130	10.2.1	Sprint Planning	<ul style="list-style-type: none"> * El Product Owner es el responsable de que las no conformidades detectadas por el cliente y sus interesados sean gestionadas correctamente 	
re/131	10.2.1	Product Owner	<ul style="list-style-type: none"> * El Scrum Master es el responsable de que las no conformidades sean detectadas durante el Sprint en las actividades de pruebas de software, y que sean gestionadas correctamente 	
re/132	10.2.1	Scrum Master	<ul style="list-style-type: none"> * El Development Team debe tener miembros competentes en pruebas de software para garantizar que el producto está libre de no conformidades, y que las no conformidades identificadas sean gestionadas correctamente antes de la liberación de un incremento 	
re/133	10.2.1	Development Team	<ul style="list-style-type: none"> * El Product Backlog contiene información sobre las no conformidades encontradas durante el sprint o en la revisión del mismo, al igual que información de la gestión de la no conformidad 	
re/134	10.2.1	Product Backlog	<ul style="list-style-type: none"> * El Sprint Backlog contiene información sobre las no conformidades encontradas durante el sprint en las actividades de pruebas de software, al igual que información de la gestión de la no conformidad 	
re/135	10.2.1	Sprint Backlog	<ul style="list-style-type: none"> * El incremento es la evidencia de la eficacia de las acciones correctivas sobre no conformidades encontradas en incrementos previos 	
re/136	10.2.1	Increment	<ul style="list-style-type: none"> * El Product Owner es el responsable de conservar la información documentada de las no conformidades detectadas en las revisiones de sprint y la gestión correcta de las mismas 	
re/137	10.2.2	Product Owner	<ul style="list-style-type: none"> * El Scrum Master es el responsable de conservar la información documentada de las no conformidades detectadas durante el sprint en las actividades de pruebas de software y la gestión correcta de las mismas 	
re/138	10.2.2	Scrum Master	<ul style="list-style-type: none"> * Contiene la evidencia de la no conformidad detectada y su gestión a lo largo del proyecto 	
re/139	10.2.2	Product Backlog		
re/140	10.2.2	Sprint Backlog		

Anexo D. Revisión sistemática acerca de la integración del estándar ISO 9001 con enfoques ágiles de desarrollo de software



Universidad del Cauca

**Revisión Sistemática acerca de la integración del estándar ISO 9001 con
enfoques ágiles de desarrollo de software**

*Facultad de Ingeniería Electrónica y de Telecomunicaciones
Ingeniería de Sistemas, Diciembre de 2018*

Estudiante: Miguel Andrés Morcillo
Tutor: PhD. César Jesús Pardo Calvache

Tabla de contenido.

1. Introducción.	4
2. Planeación de la revisión.	4
2.1. Formulación de la pregunta.	4
2.1.1. Foco de la pregunta.	4
2.1.2. Amplitud y calidad de la pregunta.	4
2.1.2.1. Problema.	4
2.1.2.2. Pregunta de investigación.	5
2.1.2.3. Palabras claves y sinónimos.	5
2.1.2.4. Intervención.	5
2.1.2.5. Control.	5
2.1.2.6. Resultado.	6
2.1.2.7. Población.	6
2.1.2.8. Aplicación.	6
2.1.2.9. Diseño experimental.	6
2.2. Selección de las fuentes.	7
2.2.1. Definición del criterio de selección de fuentes.	7
2.2.2. Lenguaje de estudio.	7
2.2.3. Identificación de fuentes.	7
2.2.3.1. Método de selección de fuentes.	7
2.2.3.2. Cadenas de búsqueda.	8
2.2.3.3. Lista de fuentes.	9
2.3. Selección de los estudios.	9
2.3.1. Definición de los estudios.	10
2.3.1.1. Definición de los criterios de inclusión y de exclusión de estudios.	10
2.3.1.1. Procedimiento para la selección de estudios.	10
3. Ejecución de la selección.	11
3.1. Ejecución de la búsqueda en las fuentes de datos seleccionadas.	12

3.1.1. Ejecución de la búsqueda en la biblioteca ScienceDirect@ComputerScience.....	12
3.1.2. Ejecución de la búsqueda en la biblioteca Springer Link.	13
3.1.3. Ejecución de la búsqueda en la biblioteca IEEE Xplore.....	15
3.1.4. Ejecución de la búsqueda en la biblioteca ACM Digital Library	17
3.2. Selección de estudios primarios.....	18
3.3. Evaluación de la calidad de los estudios.	18
3.4. Extracción de información.....	19
3.4.1 Definición de los criterios de inclusión y extracción de información.....	19
3.4.2. Formulario para la extracción de información.	20
4. Análisis de los resultados.....	21
4.1. Estudios analizados.....	21
4.2. Presentación de resultados.	33
4.2.1. Análisis estadístico de los estudios primarios.	33
4.2.2. Análisis estadístico de los estudios de caso realizados.	37
4.2.2.1. Factores de éxito y fracaso	39
5. Conclusiones y trabajos futuros.	42
Bibliografía.	44
Anexo A.	47
Anexo B.	61

Índice de tablas.

Tabla 1. Palabras clave.....	5
Tabla 2. Cadenas de búsqueda.....	9
Tabla 3. Resultados de búsqueda en ScienceDirect@ComputerScience.....	13
Tabla 4. Resultados de búsqueda en Springer Link.	15
Tabla 5. Resultados de búsqueda en IEEE Xplore.....	17
Tabla 6. Resultados de búsqueda en ACM Digital Library.....	18
Tabla 7. Formulario para la extracción de información.....	20
Tabla 8. Estudios primarios analizados.....	24
Tabla 9. Estudios con aplicación en un estudio de caso.	32
Tabla 10. Estudios encontrados, relevantes y primarios clasificados por fuente de datos.....	33
Tabla 11. Tipos de empresa en los estudios de caso.	39
Tabla 12. Métodos ágiles en los estudios de caso.....	39
Tabla 13. Factores de éxito o fracaso	42

Índice de ilustraciones.

Ilustración 1. Procedimiento para la selección de estudios.	11
Ilustración 2. Publicaciones por año.....	34
Ilustración 3. Porcentaje de estudios por país.	35
Ilustración 4. Modelos, estándares y metodologías ágiles analizadas en los estudios primarios.....	36
Ilustración 5. Técnicas de armonización utilizadas	37
Ilustración 6. Éxito de los estudios de caso de ISO 9001 con métodos ágiles y otros estándares con métodos ágiles.....	38

1. Introducción.

2. Planeación de la revisión.

2.1. Formulación de la pregunta.

Se definen los objetivos de la revisión sistemática acerca de la integración de la norma ISO 9001 con metodologías de desarrollo de software ágil.

2.1.1. Foco de la pregunta.

Con la revisión sistemática se busca identificar iniciativas y estudios relacionados en los cuales se definan o propongan métodos, marcos, estrategias y/o soluciones para llevar a cabo la unión, integración, y/o armonización de ISO 9001 y metodologías ágiles de desarrollo de software.

2.1.2 Amplitud y calidad de la pregunta.

2.1.2.1. Problema.

Las organizaciones que proveen productos y servicios software buscan implementar sistemas de gestión de la calidad basados en la norma técnica internacional ISO 9001, a manera de garantizar la calidad de sus procesos y por lo tanto de los productos y/o servicios que proveen, en el otro lado los clientes también se ven influenciados a preferir organizaciones certificadas en ISO 9001 con la premisa de que obtendrán un producto y/o servicio con calidad garantizada. La adopción de un sistema de gestión de la calidad es una decisión estratégica para una organización que le puede ayudar a mejorar su desempeño global y proporcionar una base sólida para las iniciativas de desarrollo sostenible [1].

Scrum, siendo un marco de trabajo dentro del cual personas pueden solucionar problemas complejos y adaptados a clientes específicos, a la vez que productiva y creativamente entregar productos del mayor valor posible [2].

2.1.2.2. Pregunta de investigación.

¿Qué trabajos e iniciativas relacionadas con la implementación de ISO 9001 y metodologías ágiles se han llevado a cabo?

2.1.2.3. Palabras claves y sinónimos.

Ítem.	Palabras clave	Sinónimo
1	ISO	ISO/IEC
2	Agile	SCRUM, XP
3	Standard	Model, framework, technology, model, method, methodology
4	Quality	QA
5	Assurance	Guarantee, insurance
6	Software	Product, service, system
7	Development	Construction, execution
8	Implementation	Application, utilization, usage, integration, union, harmonization

Tabla 1. Palabras clave.

2.1.2.4. Intervención.

Se observarán las propuestas y trabajos existentes para la implementación de ISO 9001 y metodologías ágiles.

2.1.2.5. Control.

Para la revisión se han observado antes algunos trabajos relacionados con el objetivo de obtener las palabras claves, algunos fueron encontrados en las fuentes de búsqueda seleccionadas y otros aunque no fueron encontrados, se consideró importante tener algunos en cuenta dentro del conjunto de estudios primarios como literatura gris por su relación con el tema en cuestión.

2.1.2.6. Resultado.

El resultado esperado en esta revisión sistemática es la identificación de aquellos estudios entre propuestas, iniciativas y/o trabajos que están relacionados con la implementación de ISO 9001 y metodologías ágiles. Posteriormente los estudios identificados serán analizados con el objetivo de determinar similitudes, diferencias y necesidades de investigación.

2.1.2.7. Población.

La población que será analizada, se encuentra en las publicaciones de las fuentes de datos seleccionadas y que han pasado por el criterio de búsqueda aplicado para esta revisión sistemática.

2.1.2.8. Aplicación.

Los beneficiarios de los resultados obtenidos en esta revisión son todos aquellos investigadores, académicos, universidades, profesionales y organizaciones de la industria relacionados con la norma ISO 9001 y su implementación usando metodologías ágiles.

2.1.2.9. Diseño experimental.

El meta análisis de esta revisión sistemática tiene como objetivo determinar, entre otros:

- Publicaciones por año.
- Publicaciones por país.
- Porcentaje de estándares/modelos/metodologías analizados.
- Tipos de empresas.

- Factores de éxito y fracaso en la implementación de ISO 9001 y metodologías ágiles.
- Técnicas de armonización utilizadas.

2.2. Selección de las fuentes.

El objetivo principal de esta sección es seleccionar las fuentes de datos donde se llevará a cabo la revisión.

2.2.1. Definición del criterio de selección de fuentes.

El criterio de selección de fuentes de búsqueda está basado en la relación con el tema de la revisión sistemática, la disponibilidad de acuerdo al perfil académico del investigador y la accesibilidad de las mismas vía Web en Internet. De acuerdo a los criterios anteriores, el listado de fuentes también ha sido validado por investigadores expertos en los temas de calidad de software y metodologías ágiles.

2.2.2. Lenguaje de estudio.

El lenguaje en el cual se llevará a cabo la búsqueda en las diferentes fuentes de datos para la obtención de los estudios primarios es inglés. El lenguaje en el que se realizará el informe de esta revisión es español.

2.2.3. Identificación de fuentes.

El objetivo de este apartado es seleccionar las fuentes de búsqueda para realizar la revisión sistemática.

2.2.3.1. Método de selección de fuentes.

Basándose en la opinión del: PhD. César Pardo y PhD. Francisco Pino, investigadores expertos en los temas de esta revisión sistemática, se han seleccionado las fuentes de datos que a su juicio, son aquellas que al cumplir con las condiciones y criterios de selección (ver 2.2.1.) ofrecen artículos de calidad.

2.2.3.2. Cadenas de búsqueda.

Combinando los conectores lógicos “AND” y “OR” con las palabras claves identificadas, se realizó una primera búsqueda con el prototipo inicial de cadena de búsqueda sobre una fuente. Obteniendo así las siguientes cadenas de búsqueda básicas.

No.	Cadena de búsqueda básica
1	<p>(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology) AND (quality OR guarantee OR insurance OR assurance OR software OR product OR service OR system OR development OR execution OR construction OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization)</p>
2	<p>(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction) AND (quality OR guarantee OR insurance OR assurance OR software OR product OR service OR system OR development)</p>
3	<p>(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND</p>

	(standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction)
--	--

Tabla 2. Cadenas de búsqueda.

Debido al estado del arte del problema identificado para esta revisión, existen pocos estudios relevantes respecto a la implementación de ISO 9001 y metodologías ágiles, las tendencias muestran que hay variedad de estudios que tratan estándares ISO como 12207, 15504 y algunas metodologías ágiles, o referente a la calidad hay variedad de estudios que tratan modelos como CMM, CMMI y metodologías ágiles. Por esa razón se ha decidido utilizar la cadena de búsqueda número 3, que es más simple respecto a 1 y 2, y utiliza menos operadores lógicos AND, permitiendo así la inclusión de estudios que podrían haber sido dejados por fuera con las cadenas 1 y 2.

La cadena de búsqueda seleccionada sin embargo, fue adaptada a cada motor de búsqueda de las fuentes de datos seleccionadas, las adaptaciones realizadas se presentarán más adelante.

2.2.3.3. Lista de fuentes.

La lista de fuentes obtenida para llevar a cabo la revisión sistemática es la siguiente:

- ScienceDirect@ComputerScience,
- Springer Link,
- IEEE Xplore,
- ACM Digital Library,
- Un conjunto de artículos y trabajos proporcionados por los expertos fueron considerados como literatura gris.

2.3. Selección de los estudios.

Con las fuentes de datos definidas para llevar a cabo la búsqueda de esta revisión sistemática, se describirá el proceso y los criterios de selección de los estudios primarios.

2.3.1. Definición de los estudios.

2.3.1.1. Definición de los criterios de inclusión y de exclusión de estudios.

El criterio de inclusión aplicado a los estudios encontrados como resultado de la búsqueda en cada fuente de datos se enfocó en el análisis del título, abstract y palabras clave. Esto permitió determinar qué estudios tenían relación con la implementación de ISO 9001 y metodologías ágiles, obteniendo así los estudios relevantes.

El criterio de exclusión aplicado a los estudios relevantes se enfocó en la lectura y análisis detallada del abstract y la introducción de cada estudio, para aquellos estudios en los que no era clara la relación con total o parcial con el tema de esta revisión fue necesario leer parte del cuerpo del estudio, de esta forma se excluyeron estudios que aunque estaban relacionados parcialmente con el criterio de búsqueda, el desarrollo de la investigación no aportaba para esta revisión.

2.3.1.1. Procedimiento para la selección de estudios

En la **Ilustración 1** se puede observar el procedimiento para la selección de estudios.

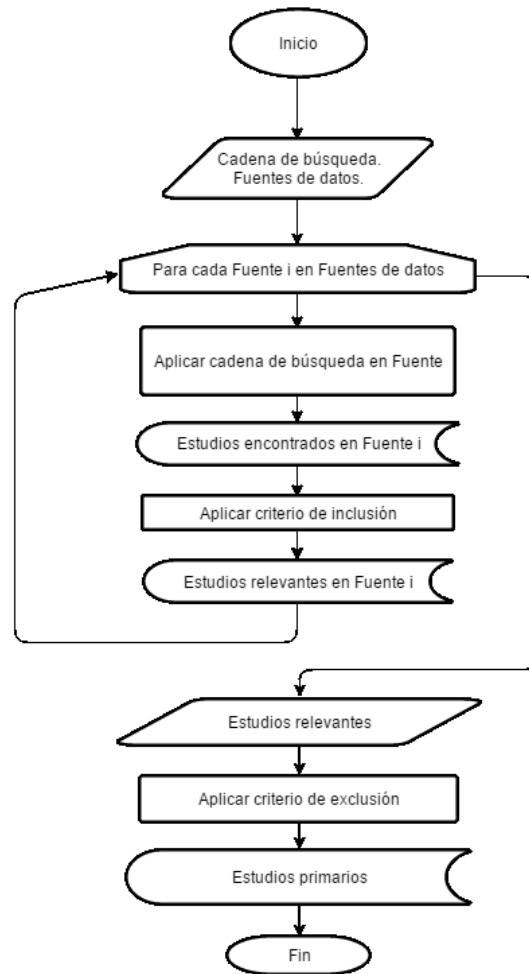


Ilustración 1. Procedimiento para la selección de estudios.

3. Ejecución de la selección.

Una vez llevada a cabo la planeación de la revisión, se procede a realizar la ejecución de la misma en cada una de las fuentes seleccionadas aplicando los criterios y el procedimiento establecido.

A continuación se presenta información sobre la ejecución de la revisión en cada una de las fuentes, incluyendo el número de búsqueda, el motor de búsqueda, la cadena original, el alcance de la búsqueda, las secciones de ayuda que cada

biblioteca proveía, captura de pantalla de la configuración de la cadena de búsqueda adaptada a cada buscador y los resultados obtenidos en cada fuente.

3.1. Ejecución de la búsqueda en las fuentes de datos seleccionadas.

3.1.1. Ejecución de la búsqueda en la biblioteca ScienceDirect@ComputerScience.

Al aplicar la cadena de búsqueda a la biblioteca ScienceDirect@ComputerScience para diciembre de 2018, 8 resultados fueron arrojados por el buscador de los cuales 1 fue seleccionado como relevante. (Ver anexo A).

Resultados de búsqueda	
Búsqueda N°:	1
Motor de búsqueda:	ScienceDirect@ComputerScience
Cadena de búsqueda:	(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction)
Alcance de la búsqueda:	Título, Abstract, Keywords
Ayudas para la búsqueda:	Looking for phrases instead of words: http://help.sciencedirect.com/acceso.unicauca.edu.co/flare/sdhelp_Left.htm#CSHID=stadv_main_all.htm StartTopic=Content%2Fstadv_main_all.htm SkinName=svs_SD

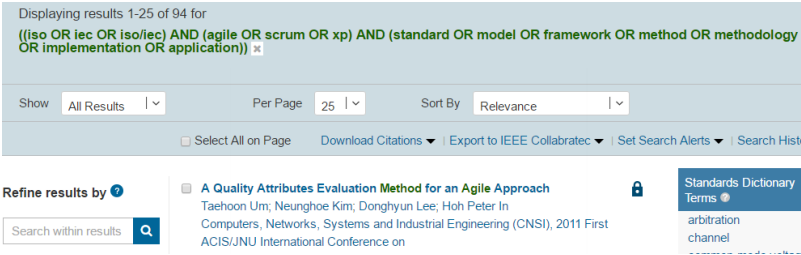
	<p style="text-align: center;">Conectores: http://help.sciencedirect.com.acceso.unicauca.edu.co/flare/Content/stadv_connectors.htm</p> <p style="text-align: center;">Campos: http://help.sciencedirect.com.acceso.unicauca.edu.co/flare/Content/stexpt_field_all.htm</p>
<p>Screenshot de configuración de búsqueda:</p>	
<p>Resultados:</p>	<p style="text-align: center;">8 resultados:</p> <p style="text-align: center;"> http://www.sciencedirect.com.acceso.unicauca.edu.co/science?_ob=ArticleListURL&_method=list&_ArticleListID=-1051602275&_sort=r&_st=5&md5=5cb04c40aa156b07411785ae19d2e2d7&searchtype=a </p>

Tabla 3. Resultados de búsqueda en ScienceDirect@ComputerScience

3.1.2. Ejecución de la búsqueda en la biblioteca Springer Link.

Al aplicar la cadena de búsqueda a la biblioteca Springer Link para diciembre de 2018, 224 resultados fueron arrojados por el buscador de los cuales 11 fueron seleccionados como relevantes. **(Ver anexo A).**

Resultados de búsqueda	
Búsqueda N°:	2
Motor de búsqueda:	Springer Link
Cadena de búsqueda:	<p style="text-align: center;">(9001) AND (ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction)</p> <p style="text-align: center;">Cadena adaptada al buscador:</p> <p style="text-align: center;">(9001)AND((iso)OR(iec)OR("iso/iec"))AND((agile) OR(scrum)OR(xp))AND((standard)OR(model)OR(fr amework)OR(technology)OR(method)OR(methodol ogy)OR(implementation)OR(application)OR(utilizati on)OR(usage)OR(integration)OR(union)OR(harmoni zation)OR(execution)OR(construction))</p>
Alcance de la búsqueda:	Todo
Ayudas para la búsqueda:	Springer search help: http://link.springer.com/searchhelp

	<p>(agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction)</p> <p>Cadena adaptada al buscador:</p> <p>((iso OR iec OR iso/iec) AND (agile OR scrum OR xp) AND (standard OR model OR framework OR method OR methodology OR implementation OR application))</p>
Alcance de la búsqueda:	Full Metadata
Ayudas para la búsqueda:	<p>Command search examples:</p> <p>http://ieeexplore.ieee.org/Xplorehelp/#/searching-ieee-xplore/search-examples</p>
Screenshot de configuración de búsqueda:	 <p>The screenshot shows a search results page for the query: ((iso OR iec OR iso/iec) AND (agile OR scrum OR xp) AND (standard OR model OR framework OR method OR methodology OR implementation OR application)). The page displays 1-25 of 94 results. The search interface includes options for 'Show All Results', 'Per Page 25', and 'Sort By Relevance'. A result snippet is visible: 'A Quality Attributes Evaluation Method for an Agile Approach' by Tae-hoon Um, Neunghoe Kim, Donghyun Lee, and Hoh Peter In, published in 'Computers, Networks, Systems and Industrial Engineering (CNSI), 2011 First ACIS/JNU International Conference on...'. There is also a 'Standards Dictionary Terms' button for 'arbitration channel'.</p>
Resultados:	<p>112 resultados</p> <p>http://ieeexplore.ieee.org/search/searchresult.jsp?action=search&sortType=&rowsPerPage=&matchBoolean=true&searchField=Search_All&queryText=((iso%20OR%20iec%20OR%20iso%20Fiec)%20AND%20%0A(agile%20OR%20scrum%20OR%20xp)%20AND%0A(standard%20OR%20model%20OR%20framework%20OR%20method%20OR%20methodology%20OR%20implementation%20OR%20application))&newsearch=true</p>

--	--

Tabla 5. Resultados de búsqueda en IEEE Xplore.

3.1.4. Ejecución de la búsqueda en la biblioteca ACM Digital Library

Al aplicar la cadena de búsqueda a la biblioteca ACM Digital Library para diciembre de 2018, 20 resultados fueron arrojados por el buscador de los cuales 1 fue seleccionado como relevante. **(Ver anexo A).**

Resultados de búsqueda	
Búsqueda N°:	4
Motor de búsqueda:	ACM Digital Library
Cadena de búsqueda:	<p style="text-align: center;">(ISO OR IEC OR ISO/IEC) AND (agile OR SCRUM OR XP) AND (standard OR model OR framework OR technology OR method OR methodology OR implementation OR application OR utilization OR usage OR integration OR union OR harmonization OR execution OR construction)</p> <p style="text-align: center;">Cadena adaptada al buscador:</p> <p style="text-align: center;">iso OR iec OR "iso/iec" AND agile OR scrum OR xp</p>
Alcance de la búsqueda:	Any Field
Ayudas para la búsqueda:	No fue encontrado enlace de ayuda para la búsqueda, empíricamente se encontró más fácil manejar los operadores lógicos desde la barra de búsqueda

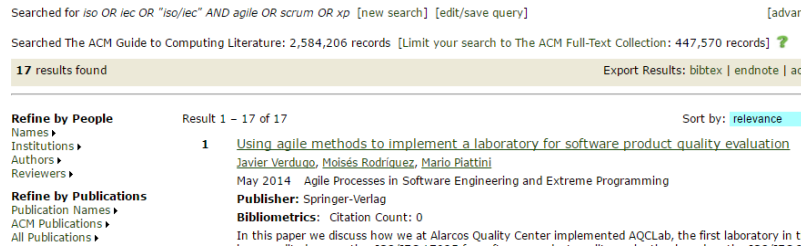
	<p>sencilla. Las palabras escritas en minúscula y los operadores en mayúscula (AND y OR) las frases deben llevar comillas dobles</p>
<p>Screenshot de configuración de búsqueda:</p>	
<p>Resultados:</p>	<p>20 resultados</p> <p>http://dl.acm.org/results.cfm?query=iso%20OR%20iec%20OR%20%22iso%2Fiec%22%20AND%20agile%20OR%20scrum%20OR%20xp&filtered=&within=owners.owner=GUIDE&dte=&bfr=&srt=_score</p>

Tabla 6. Resultados de búsqueda en ACM Digital Library.

3.2. Selección de estudios primarios.

21 estudios fueron seleccionados como relevantes al aplicar el criterio de inclusión a los resultados arrojados por los buscadores de las fuentes de datos, 4 estudios más fueron añadidos al conjunto de estudios relevantes considerados como literatura gris, dando un total de 25 estudios relevantes.

Posteriormente el conjunto de los estudios relevantes fue sometido a criterio de exclusión, arrojando así 20 estudios primarios, sin embargo fueron retirados 3 estudios debido a que 1 de ellos era un estándar internacional de ISO/IEEE sobre cómo generar documentación en proyectos ágiles, otros 2 eran demasiado extensos para el alcance de esta revisión.

3.3. Evaluación de la calidad de los estudios.

Se presume que todos los estudios seleccionados de ScienceDirect, Springer Link, IEEE Xplore y ACM Digital Library son de calidad, ya que para haber sido publicados en esta fuente han debido superar una cadena de evaluaciones y filtros mínimos de calidad.

Respecto a los estudios de la literatura gris que no fueron encontrados en las fuentes, y fueron incluidos como estudios primarios. Fueron revisados y sugeridos por los expertos, por tanto se presume que son de calidad.

3.4. Extracción de información.

Una vez seleccionados los estudios primarios es posible comenzar con la extracción de la información más relevante de cada uno de ellos. Dicha extracción se realiza a partir de criterios de extracción de información definidos a continuación.

3.4.1 Definición de los criterios de inclusión y extracción de información.

De acuerdo al objetivo de la revisión sistemática, el criterio de inclusión de información se enfoca en los aportes de interés más importantes con relación a la implementación de ISO 9001 y metodologías ágiles, para identificar los aportes se consideran los siguientes factores:

- Conocer los nombres y la cantidad de estándares, modelos y/o metodologías analizadas.
- Identificar las soluciones propuestas y/o usadas para llevar a cabo la implementación de ISO 9001 y metodologías ágiles.
- Identificar el tipo de investigación y/o análisis utilizado (mapeo, comparación, compatibilización, combinación, etc.).
- Declaraciones tenidas en cuenta y demás ideas y/o factores que se consideren importantes.

3.4.2. Formulario para la extracción de información.

La información de los estudios primarios fue analizada y almacenada en una tabla validada por los expertos cuyo formato tiene la siguiente estructura:

Título	
Publicación	
Año	
Autor(es)	
Abstract	
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	Identifica los modelos de calidad utilizados y/o integrados a ISO 9001, así mismo, trata de identificar el enfoque o metodología ágil utilizada.
Métodos de armonización utilizados	Identifica los métodos o técnicas de homogeneización, comparación e integración utilizados para la definición de modelos integrados.
Nivel de detalle de la propuesta	Referente al nivel de detalle en la integración de los modelos utilizados, por ejemplo: elementos de proceso definidos, descripciones, entre otros.
Tipo de evaluación	Clasifica el tipo de evaluación utilizada para validar la propuesta, estudios de caso, encuestas, validación de expertos, experimentos, etc.
Resumen y aspectos a destacar	

Tabla 7. Formulario para la extracción de información.

Las tablas con los resúmenes de todos los estudios primarios pueden ser encontradas en el anexo B. **(ver anexo B).**

4. Análisis de los resultados.

En esta sección se presentan los datos reunidos de los estudios primarios seleccionados.

4.1. Estudios analizados.

En la siguiente tabla se presentan los estudios analizados con su nombre, los modelos que son analizados y la aplicación de la investigación, además a cada estudio se les asignado un número.

Nº estudio	Nombre del artículo	Modelos analizados	Aplicación
1	Achieving ISO 9001 Certification for an XP Company	ISO 9001:2000, XP	Estudio de caso
2	A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process	ISO 29110, Scrum, XP, Upedu	NO SE EVIDENCIA
3	Applying ISO/IEC 12207:2008 with Scrum and Agile Methods	ISO/IEC 12207:2008, Scrum	Análisis de 25 empresas
4	Assessing XP at an European	XP	Estudio de caso

	Company		
5	Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods	NO SE EVIDENCIA	NO SE EVIDENCIA
6	Certifying for CMM Level 2 and ISO 9001 with XP@Scrum	CMM Level 2, ISO 9001:2000, XP@Scrum	Estudio de caso
7	Combining Extreme Programming with ISO 9000	ISO 9001:2000, XP	Estudio de caso
8	Getting ISO 9001 Certified for Software Development Using Scrum and Open Source Tools	ISO 9001:2008, Scrum	Estudio de caso
9	How can Agile and Documentation-Driven Methods be Meshed in Practice	GAMP, FDA, Scrum	Estudios de caso
10	Investigation of The Capability of XP to Support The Requirements of ISO 9001 Software Process Certification	ISO 9001:2008, ISO 90003, ISO 12207, XP	NO SE EVIDENCIA

11	ISO 9001 And Agile Development	ISO 9001:2000, SCRUM/XP	Estudio de caso
12	Reconciling The Irreconcilable A Software Development Approach That Combines Agile with Formal	ISO 9001/ISO 12207/SPICE/ACSS/NASA-SEL, Scrum/XP/ASD	Estudio de caso
13	Requirements of An ISO Compliant XP Tool	ISO 9001:2000, XP	NO SE EVIDENCIA
14	Standards And Agile Software Development	ISO 12207:1995, XP	NO SE EVIDENCIA
15	Structure of A Multi-Model Catalog For a Software Projects Management Including Agile And Traditional Practices	PMI/CMMI/ISO 9001/TSP, Scrum/Kanban/XP .	NO SE EVIDENCIA
16	The Agile and The Disciplined Software Approaches: Combinable or just Compatible	CMMI/ISO 9001, Scrum/XP	NO SE EVIDENCIA
17	The Application of ISO 9001 to Agile Software Development	ISO 9001, Scrum, XP	NO SE EVIDENCIA

18	The Effect of Moving From A Plan Driven To An Incremental Software Development Approach with Agile Practices	ISO 9001:2000, Scrum, XP	Estudio de caso
----	--	--------------------------	-----------------

Tabla 8. Estudios primarios analizados.

A continuación se presenta una tabla referente a los estudios cuya aplicación es un estudio de caso, la propuesta de implementación, el tipo de empresa en la que se hizo el estudio de caso, los modelos aplicados, las lecciones aprendidas, los desafíos enfrentados y una conclusión del estudio referente al éxito o fracaso de implementación de los diferentes modelos.

Nº de estudio	Propuesta	Tipo de empresa	Modelos aplicados	Lecciones aprendidas	Desafíos enfrentados	Conclusión
1	TickIT, Mapping	Gran empresa	ISO 9001:2000, XP	* La calidad y mejora debe recaer en las salidas de los procesos de XP, y no en artefactos creados puramente para cumplir con el estándar. * Es necesario usar una herramienta de gestión (virtual white board)	* Elaborar los manuales y políticas de calidad con el nivel de detalle necesario para cumplir con el estándar	Éxito

				* Pruebas en todas las etapas del desarrollo de software		
4	Tailoring	Mediana empresa	ISO 9001:2000, XP	<ul style="list-style-type: none"> * El Coach de XP debe fomentar la participación activa de todos en el Planning Game * No se debe puntuar a los desarrolladores por su productividad o efectividad * XP minimiza la necesidad de Diagramas de Gantt * XP mejora la sensación de control del proyecto para los desarrolladores 	<ul style="list-style-type: none"> * En ninguno de los proyectos se pudo aplicar "metáfora del sistema" * XP no tiene una receta clara para trabajar la arquitectura * los clientes no asimilan fácilmente la impredecibilidad de costos y tiempos fijos * El problema más grande es gran necesidad del cliente en el sitio 	Éxito
6	Tailoring	Pequeña empresa	CMM Level2, ISO 9001:2000, XP@Scrum	<ul style="list-style-type: none"> * Es necesaria mínima burocracia para crear tanto como sea posible * No reinventar la rueda y usar prácticas ya 	<ul style="list-style-type: none"> * Difícil de escribir buenas metáforas * Dificultan al decidir cuándo usar pair programming * Carencia de fase de mejora 	Éxito

				probadas * Utilizar buenas herramientas de gestión y soporte al proceso de desarrollo * Se incrementó la satisfacción de clientes y desarrolladores * Se requiere mucha disciplina * Se percibió mejor especialización de trabajo y roles asignados * Es necesario un consultor de ISO * Es necesario usar métricas para control del proceso * Es necesario contratar a los mejores * Permitir personalización de acuerdo al proyecto * El encargado de calidad debe ser coach y no un policía	y optimización * Difícil gestión de clientes al pedirles requerimientos claros y precisos en forma de tests de aceptación	
--	--	--	--	---	--	--

7	XPMM	Mediana empresa	ISO 9001:2000, XP	<p>* El equipo de desarrollo no debe lidiar con documentación de requerimientos o de la dirección, es necesario dar esa responsabilidad a otra persona como el tester</p> <p>* Para implementar correctamente pair programming es necesario un espacio de trabajo abierto</p>	<p>* Los problemas más importantes son el mantenimiento del software por falta de documentación</p>	Éxito
8	Mapping	Mediana empresa	ISO 9001:2008, Scrum	<p>* Las presentaciones, diapositivas e informes de progreso no son considerados de valor, el software funcionando si</p> <p>* Un malentendido común es que agilidad no necesita documentación</p>	<p>* La reducción de la documentación fue difícil</p> <p>* El mayor problema fue la documentación para la forma en que acciones correctivas y preventivas son gestionadas</p>	Éxito

				<p>, aunque la prioridad es el software si se necesita algo de documentación</p> <ul style="list-style-type: none"> * Es necesario buen entrenamiento en ISO o un consultor externo * Es necesario que el diagrama organizacional del QMS sea simple, no hay que asignar varios roles a una sola persona * Es necesario usar herramientas de gestión y soporte para el proceso de desarrollo y trazabilidad * Es necesario un software para gestionar la trazabilidad de impedimentos, PNCs y las respectivas acciones 		
--	--	--	--	--	--	--

				correctivas y preventivas		
9	Meshing	Pequeña empresa	GAMP, Scrum	<ul style="list-style-type: none"> * La empresa era ágil y posteriormente trataron de implementar el estándar GAMP4 * Es necesaria mucha disciplina * La agilidad había generado una enorme confianza entre el cliente y los desarrolladores, lo que los influenció para no documentar lo necesario para el estándar 	<ul style="list-style-type: none"> * Excesiva documentación requerida * La gran confianza entre cliente y desarrolladores evitó la generación de documentación necesaria 	Fracaso
9	Meshing	Gran empresa	FDA, Scrum	<ul style="list-style-type: none"> * La empresa ya estaba certificada y posteriormente trataron de usar prácticas ágiles * Es necesaria mucha disciplina * La comunicación y transferencia de conocimiento 	<ul style="list-style-type: none"> * Difícil contactar con el cliente * Difícil implementación de Product Owner * Excesiva documentación FDA * Difícil transición entre requerimiento 	Éxito

				mejoró, reduciendo la necesidad de extensiva documentación pero no eliminando la necesidad	s recolectados al inicio, y recolectados iterativamente	
11	Tailoring	Gran empresa	ISO 9001:2000, XP/Scrum	<p>* Es necesario contratar un consultor externo para que se encargue del proceso de implementación de ISO 9001 sin afectar al equipo de desarrollo</p> <p>* Es necesario que la dirección esté comprometida con la implementación y mejora del proceso</p> <p>* ISO 9001 trajo a las prácticas ágiles mejoras en prácticas de la dirección, recolección de métricas, prácticas de testing</p>	* NO SE EVIDENCIA	Éxito

				automatizado y planeación del proceso		
12	Tailoring, Action-research	Gran empresa	ISO 9001/ISO 12207/S PICE/ECSS/NA SA-SEL, Scrum/XP/ASD	<ul style="list-style-type: none"> * Son necesarias herramientas para la gestión de los requerimientos, proceso de desarrollo y tests automatizados * Es necesaria mucha disciplina * Todos los miembros deben estar comprometidos, desde la dirección, el equipo, hasta el cliente y sus interesados 	<ul style="list-style-type: none"> * Sprint backlog difícil de implementar debido a la falta de disciplina * Algunos proyectos tuvieron retrasos únicamente debido a las decisiones tardías de los clientes 	Éxito
18	Tailoring, Mapping	Gran empresa	ISO 9001, XP/Scrum	<ul style="list-style-type: none"> * Las prácticas ágiles permiten tener requerimientos más estables, menos re trabajo y cambios * Las prácticas ágiles permitieron finalizar todo lo que se 	<ul style="list-style-type: none"> * Retrasos entre iteraciones debido a tiempos de espera de testing de los incrementos, debido a la estructura iterativa de dependencias que la 	Éxito

				comenzó, estimaciones más precisas y detección temprana de fallos * Las prácticas ágiles redujeron la documentación al mover al equipo de desarrollo a un lugar juntos * Las prácticas ágiles redujeron drásticamente los costos de mantenimiento	empresa definió	
--	--	--	--	--	--------------------	--

Tabla 9. Estudios con aplicación en un estudio de caso.

A continuación en la Tabla 10 se presenta el número de estudios encontrados con la cadena de búsqueda seleccionada en la Tabla 2. **Cadenas de búsqueda.** Tabla 2 clasificados por la fuente de datos (330 estudios), se presenta también el número de estudios seleccionados como relevantes (21 estudios) y primarios (16 estudios). Como se mencionó anteriormente, otros 2 estudios fueron seleccionados como primarios por los expertos, dando así un total de 18 estudios primarios.

Fuente de datos	Estudios encontrados	Estudios relevantes	Estudios primarios seleccionados
ScienceDirect@ComputerScience	8	1	1
Springer Link	223	11	8
IEEE Xplore	112	8	6
ACM Digital Library	20	1	1

Total	363	21	16
-------	-----	----	----

Tabla 10. Estudios encontrados, relevantes y primarios clasificados por fuente de datos

4.2. Presentación de resultados.

Extraída la información de los estudios seleccionados, se realizó un análisis estadístico para presentar tendencias y hallazgos de la revisión sistemática.

4.2.1. Análisis estadístico de los estudios primarios.

Como se puede observar en la ¡Error! No se encuentra el origen de la referencia. el número de estudios por año relacionados con la implementación de la norma ISO 9001:2000 y metodologías ágiles tuvo un incremento en el año 2003 con el 22.2% de publicaciones encontradas (4 artículos), esto 3 años después de que fuera publicada la versión 2000. Asimismo, se puede observar interés en los años 2002, 2006, 2007, 2008, 2011, 2013 y 2014, donde se encontró el 5.5% de publicaciones (1 artículo) en cada uno. Para el 2015 se evidencia un incremento en el interés por la implementación e integración de la norma con metodologías ágiles, encontrando el 16.6% de las publicaciones (3 artículos). Por otra parte, en los años 2005, 2009 y 2012 no se encontraron estudios publicados en las fuentes de búsqueda consultadas. Para los años 2016, 2017 y 2018 no se encontraron estudios relacionados. La norma ISO 9001 ha evolucionado 4 veces desde la primera versión publicada en el año 1987, pasando por actualizaciones en los años 1994, 2000, 2008, y su más reciente actualización del 2015. Dado que el estándar ISO 9001 es genérico, existen varias interpretaciones estandarizadas de directrices para la interpretación y aplicación de ISO 9001 en sectores específicos, para la industria del desarrollo de software se destacan las directrices presentadas en TickIT [21] e ISO 90003:2014.

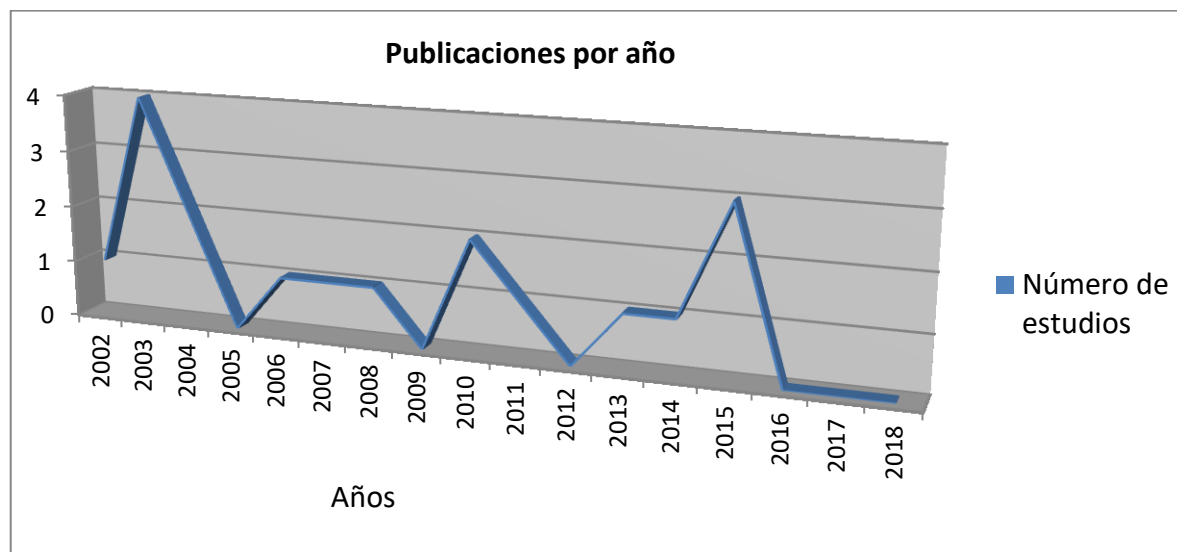


Ilustración 2. Publicaciones por año.

Del análisis de los trabajos presentados en la figura anterior, se puede observar que hay un interés en la homogeneización, comparación y/o integración de la norma ISO 9001 con relación a enfoques ágiles, sin embargo, es importante destacar que los aportes por año han sido y siguen siendo muy pocos, la realidad demanda mucha más investigación en el tema especialmente para las pequeñas y medianas empresas que en busca de mejora de la calidad divagan respecto a qué modelo se adapta mejor a su contexto y necesidades.

En la **Ilustración 3** se presenta el porcentaje de estudios por país, se han tenido en cuenta los países donde se han realizado los estudios de caso, donde se encuentran las universidades en las que se han llevado a cabo las investigaciones o las nacionalidades de los autores principales para un total de 15 países clasificados. Del análisis realizado se puede notar una gran diversidad de países interesados en la implementación de ISO 9001 y metodologías ágiles, sin embargo, los países con mayor número de estudios encontrados son Estados Unidos, Italia y Dinamarca, cada uno con un 11.1% (2 estudios), mientras que los países Reino Unido, México, España, Holanda, Polonia, Montenegro, Canadá, Portugal, Sudáfrica, Colombia, Noruega y Suecia tienen cada uno un 5.6% (1 estudio) del total de aportes.

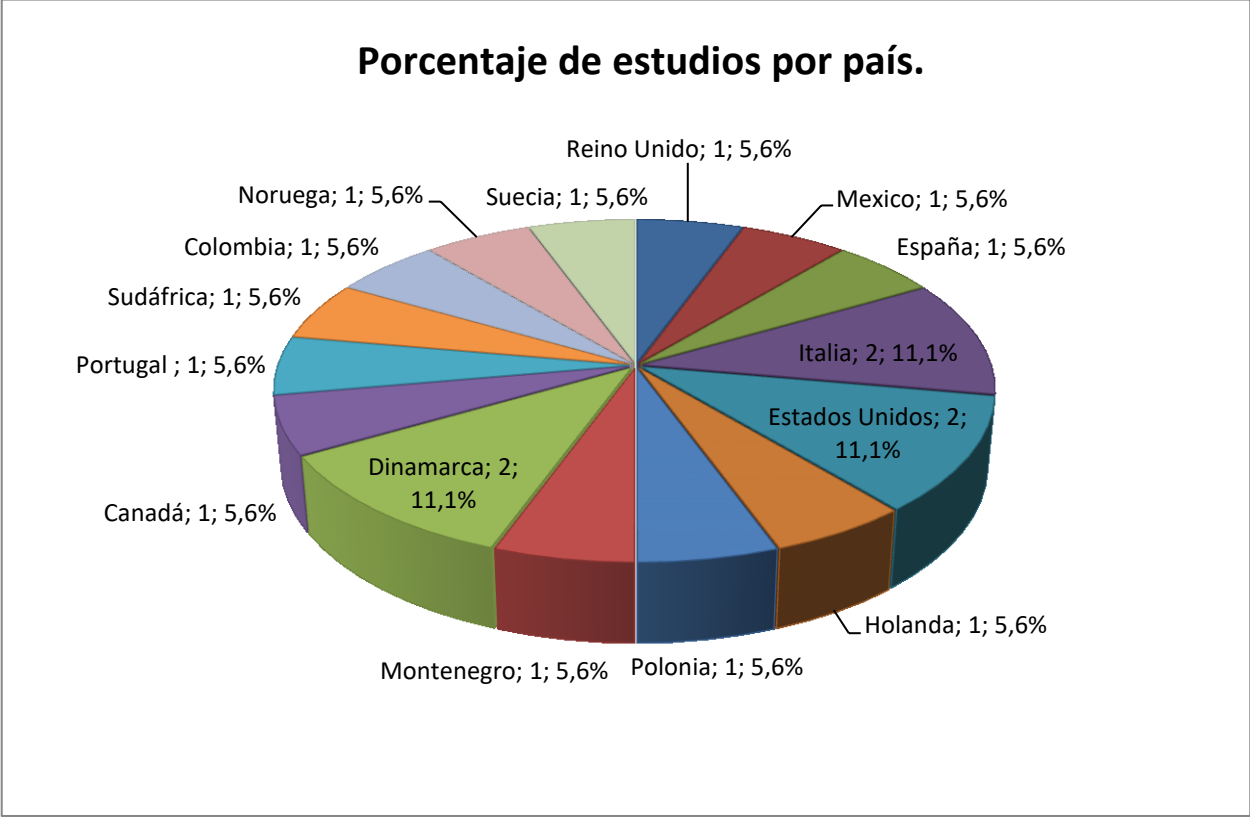


Ilustración 3. Porcentaje de estudios por país.

A continuación se puede ver en la **Ilustración 4** la distribución individual por modelo, estándar o metodología ágil analizada (en adelante enfoques para facilitar el análisis de los resultados). Los enfoques más analizados son: la metodología ágil XP con un 25.5% (14 estudios), el estándar ISO 9001 con un 21.8% (12 estudios), la metodología ágil Scrum con 20.0% (11 estudios), otros enfoques tradicionales como SPICE, ACSS, NASA-SEL, CMM, CMMI, PMI, TSP, ISO 29110, ISO 12207, GAMP y FDA abarcan un 1.8% cada uno (1 estudio), y finalmente otros métodos ágiles entre los que están UPEDU, ASD y Kanban son analizados en un 1.8% cada uno (1 estudio) del total de aportes encontrados.

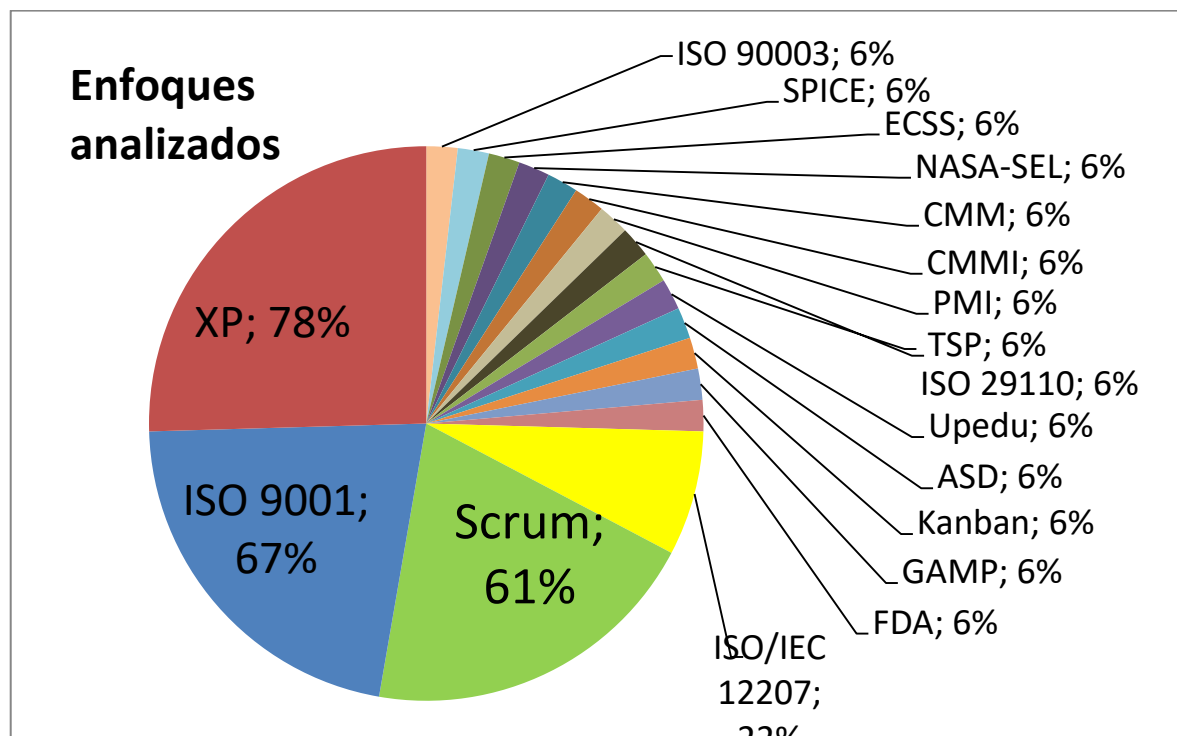


Ilustración 4. Modelos, estándares y metodologías ágiles analizadas en los estudios primarios

De la **Ilustración 4** también podemos observar que el 34.5% de los enfoques analizados fueron estándares de ISO (ISO 90003, SPICE, ISO 29110, ISO 12207 e ISO 9001), asimismo el 7.2% de los enfoques analizados fueron modelos que son aceptados y ampliamente utilizados en la industria (CMM, CMMI, PMI y TSP), otro 7.2% de los enfoques fueron estándares referentes a sectores específicos como el de la industria aeroespacial (ECSS, NASA-SEL), la manufactura (GAMP) y el estándar de la organización americana FDA (Food & Drug Administration), finalmente el 50.9% de los enfoques analizados fueron los métodos ágiles enfoques ágiles (Upedu, ASD, Kanban, Scrum y XP).

A continuación, en la **Ilustración 5** se muestran las técnicas de armonización utilizadas en los estudios primarios. La técnica más utilizada por los autores es el Mapping con un 38.9% (7 estudios), En un 33% los autores no especifican una técnica de armonización (6 estudios) debido a que se limitan a mencionar ventajas y desventajas de los diferentes enfoques o simplemente no hacen explícita la técnica de armonización utilizada, finalmente las técnicas de Risk based Spiral

Model Anchor Points [8], XPMM [10], Meshing [12], Tailoring [15], y Multi-model Catalog for Software Projects [18] tienen un 5.6% cada una (1 estudio).

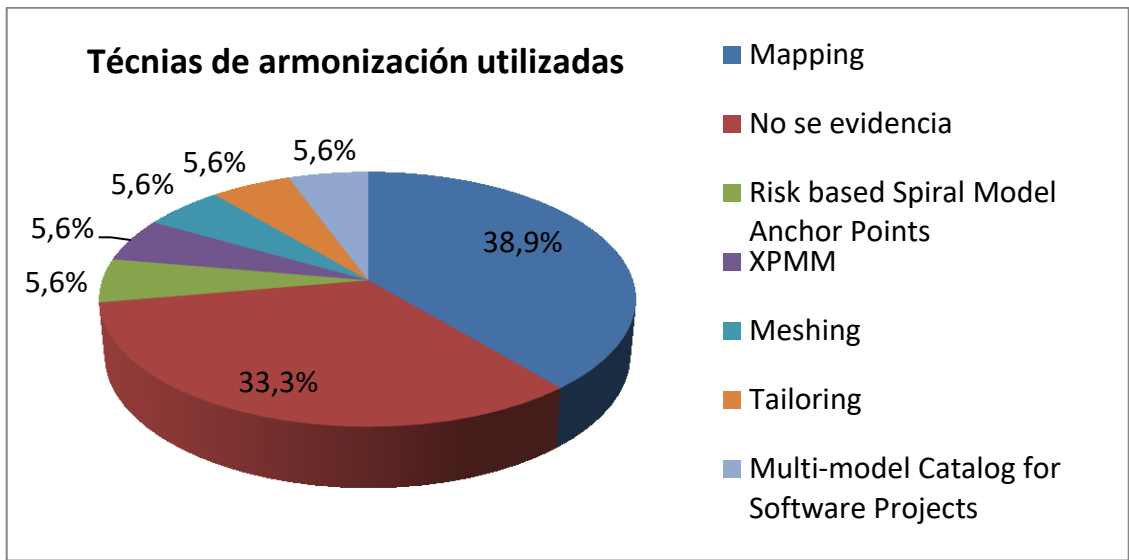


Ilustración 5. Técnicas de armonización utilizadas

4.2.2. Análisis de los estudios de caso realizados.

Con relación a los estudios primarios analizados, el 55.5% tienen aplicación como estudio de caso (10 estudios), el 80% de los estudios de caso realizados involucra estándares ISO y metodologías ágiles (8 estudios), mientras que el 20% restante analiza otros estándares similares a ISO 9001 y metodologías ágiles. Como se puede observar en la **Ilustración 6** el 100% de los estudios de caso que involucran estándares ISO y metodologías ágiles tuvo éxito (8 estudios), mientras que en los estudios de caso que involucran otros estándares y metodologías ágiles, sólo el 50% tuvo éxito (1 estudio), y el 50% restante fracasó en la implementación de los dos modelos (1 estudio).

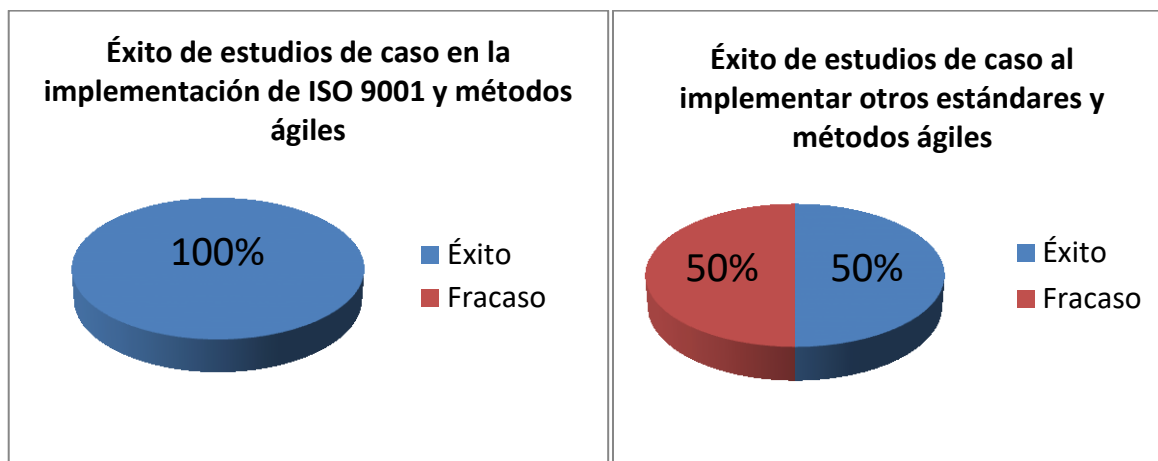


Ilustración 6. Éxito de los estudios de caso de ISO 9001 con métodos ágiles y otros estándares con métodos ágiles.

En la **Tabla 11** se presenta la información relacionada con el tipo de empresas en las cuales fueron realizados los estudios de caso, el tipo de empresa ha sido definido de acuerdo a cómo los autores se refieren a las empresas estudios de caso. Como se puede observar, fue posible organizar los estudios de caso en tres grupos de acuerdo a su tamaño:

- i) **Grandes empresas:** el 50% de los estudios de caso fueron realizados en grandes empresas (5 estudios), es importante mencionar que los estudios de caso clasificados en este grupo de empresas se caracterizaron por tener una proyección global y cuyos clientes también fueron grandes empresas.
- ii) **Medianas empresas:** el 30% fue realizado en medianas empresas (3 estudios), estas empresas se caracterizaron por tener una proyección internacional en menor medida que las grandes empresas, asimismo, fueron empresas con una cantidad menor de empleados.
- iii) **Pequeñas empresas:** el 20% de los estudios de caso fue realizado en pequeñas empresas (2 estudios), por lo general, las pequeñas empresas se desempeñan en mercados nacionales de cada región y satisfacen necesidades específicas para algún sector industrial, a diferencia de las medianas y grandes empresas que se desempeñan en diversos sectores como aeroespacial, industrial o proyectos con niveles críticos de seguridad. Cabe mencionar que la única empresa que fracasó al implementar un estándar de disciplina dirigido por el plan con una metodología ágil fue una pequeña empresa ágil, que buscaba cumplir con el estándar GAMP (Good Automated Manufacturing Practice), un conjunto de directrices para fabricantes y

usuarios de sistemas automatizados en la industria farmacéutica que se caracteriza porque requiere una gran cantidad de documentación. Los factores de éxito y fracaso identificados de los estudios de caso analizados serán tratados más adelante.

%	Nº de estudios	Tipo de empresa
50%	5	Gran empresa
30%	3	Mediana empresa
20%	2	Pequeña empresa

Tabla 11. Tipos de empresa en los estudios de caso.

En la Tabla 12, se presentan los métodos/metodologías ágiles utilizadas en los estudios de caso. El 40% corresponde a una metodología híbrida que utiliza partes de XP y Scrum (4 estudios), algunos la llaman XP@Scrum, otros autores la denominan XP/Scrum. En el resto de estudios de caso se utilizan por igual, un 30% la metodología Scrum (3 estudios) y otro 30% XP (3 estudios). El 80% de los estudios de caso utilizan ISO 9001 junto con alguna de las siguientes metodologías ágiles (8 estudios), el 20% (2 estudios) restante utiliza otros estándares de calidad similares a ISO 9001 junto con alguna de las siguientes metodologías ágiles.

%	Nº de estudios	Método/Metodología ágil
40%	4	XP y Scrum
30%	3	Scrum
30%	3	XP

Tabla 12. Métodos ágiles en los estudios de caso.

4.2.2.1. Factores de éxito o fracaso

En la Tabla 13 se muestran los factores de éxito o fracaso para la implementación de la norma ISO 9001 y metodologías ágiles, los autores expresaron algunas situaciones y características a tener en cuenta. Es importante resaltar que los factores identificados no están ordenados de acuerdo a un nivel de importancia, ni de identificación, ni de prioridad. Los factores de éxito se muestran siguiendo un patrón compuesto de: número de factor, enunciado (donde se titula el factor

identificado), descripción donde se extiende el concepto para dar mayor claridad en el entendimiento del factor y la referencia donde fue encontrado dicho factor.

FACTOR 1	
Enunciado	Los procesos, sus actividades y artefactos definidos para la organización deben existir porque contribuyen a la mejora de la calidad y no únicamente para cumplir con el estándar
Descripción	Hace referencia a que los procesos definidos realmente tengan un valor para la organización, esto debido a que las actividades y artefactos que son definidos únicamente para cumplir con el estándar son nocivos a mediano y largo plazo, el personal tiende a tomar una postura de desagrado y aburrimiento al cumplir con actividades que realmente no aportan a los objetivos de la organización. Siempre se debe dejar espacio para la personalización del proceso y encontrar una forma adecuada de cumplir con el estándar sin agregar actividades sin razón.
Referencia	[4], [9]
FACTOR 2	
Enunciado	El uso de herramientas informáticas, tanto web, desktop o móviles permiten gestionar mejor, tanto a los procesos como a las personas, además que ayudan a mantener la trazabilidad y los registros de una forma más fácil sin olvidar que ahorran papel.
Descripción	Las herramientas software pueden ayudar a gestionar el proyecto, los procesos, las actividades, el personal, base de conocimiento de soporte o errores conocidos, y pueden reemplazar a la evidencia en papel escrita reduciendo tiempos.
Referencia	[4], [15], [11]
FACTOR 3	
Enunciado	Un consultor externo es necesario, tener una opinión tercera de un experto permite saber si las cosas se están haciendo bien desde la perspectiva de ISO 9001.
Descripción	No sólo ayuda a tener una opinión imparcial sobre cómo se lleva a cabo el proceso sino también que encargar a un experto de la implementación del proceso permite que el equipo de desarrollo no se vea afectado en labores de modelado y documentación.
Referencia	[4], [9]
FACTOR 4	
Enunciado	Involucrar las pruebas en todas las partes del ciclo de vida del desarrollo de software donde sea posible, el objetivo es la calidad y

	asegurarse que el producto correcto es aprobado y el incorrecto es rechazado es fundamental para el estándar ISO 9001.
Descripción	Realizar actividades de prueba de software no sólo mejora la calidad del producto sino que reduce los costos de soporte y re trabajo, además que ayuda a satisfacer muchos de los requisitos de ISO 9001
Referencia	[4], [11], [21]
FACTOR 5	
Enunciado	El trabajo en equipo es esencial, mantener un ambiente de trabajo positivo con personal motivado conduce a una mayor sinergia en la organización.
Descripción	En las prácticas ágiles la transferencia de conocimiento se hace in situ, cara a cara y con comunicación directa, permitiendo así una reducción en los tiempos de aprendizaje y mejorando el entorno de trabajo, esto permite que la sinergia crezca y el equipo funciona de manera más eficiente.
Referencia	[21]
FACTOR 6	
Enunciado	Utilizar métricas en los diferentes procesos es necesario para identificar oportunidades de mejora y tomar las acciones respectivas.
Descripción	Lo que no se mide no se controla, para entender el funcionamiento y mejorar los procesos es necesario medirlos
Referencia	[9], [10]
FACTOR 7	
Enunciado	La documentación debe ser reducida en orden de mantener la agilidad, pero no puede ser suprimida.
Descripción	Existen varias alternativas para mejorar la gestión de la documentación y conservar las evidencias necesarias para una certificación, por ejemplo el uso de herramientas informáticas o asignar personal específico para esa tarea.
Referencia	[9], [15], [19], [21]
FACTOR 8	
Enunciado	El talento humano, el éxito en la implementación de un estándar como ISO 9001 y una metodología ágil se logra con una fuerte disciplina y el mejor equipo.
Descripción	
Referencia	[9]
FACTOR 9	
Enunciado	La falta de disciplina es el principal factor que lleva al fracaso, la agilidad no significa anarquía o indisciplina en ninguna manera.

Descripción	
Referencia	[12], [21]
FACTOR 10	
Enunciado	La falta de compromiso de todos los involucrados, incluyendo la dirección, el personal de la organización, los clientes y los interesados es una potencial causa de fracaso.
Descripción	Todos los involucrados deben estar comprometidos para tener éxito en la implementación de una metodología ágil, más con ISO 9001. Desde clientes, proveedores e interesados, hasta la alta dirección, todos deben ser conscientes de su labor y llevarla a cabo de la mejor manera
Referencia	[9], [15], [21]
FACTOR 11	
Enunciado	La falta de documentación, es una causa de fracaso para la implementación de un sistema de gestión de la calidad conforme a ISO 9001 o estándares similares.
Descripción	Si bien se busca reducir la documentación para mantener la agilidad, la documentación es necesaria y debe ser generada y conservada en la medida necesaria.
Referencia	[12], [21]

Tabla 13. Factores de éxito o fracaso

5. Conclusiones y trabajos futuros.

Revisados los trabajos relacionados con la implementación de ISO 9001 y metodologías ágiles desde el año 2000 hasta 2018, se puede decir que los estudios encontrados en las fuentes seleccionadas tienden a analizar en mayor medida la metodología ágil XP, respecto al estándar la norma ISO 9001:2000 predomina, en menor medida se analiza la norma ISO 9001:2008. Diferentes autores expresaron como un impedimento el hecho de que XP no tiene una actividad de mejora continua o reflexión sobre el proceso, aspecto que es muy importante para ISO 9001, Scrum por otra parte tiene una naturaleza de mejora continua con la actividad “Retrospectiva”, sin embargo es necesario que se lleven a cabo investigaciones al respecto para tener una perspectiva más amplia de las ventajas y desventajas de tal implementación.

También se puede evidenciar que la implementación de ISO 9001 y metodologías ágiles es posible y conlleva beneficios para las medianas y grandes empresas, como reportan los estudios de caso revisados y los estudios teóricos. Sin embargo, son necesarios más estudios de caso en micro y pequeñas empresas, puesto que medianas y grandes a menudo ya tienen experiencia en estándares de calidad al momento de implementar prácticas ágiles, las micro y pequeñas empresas presentan inconvenientes en la selección de modelos y estándares que se adapten a sus necesidades y la forma en cómo implementarlos.

Es importante entender que los métodos ágiles potencialmente dan soporte a las cláusulas de la norma ISO 9001 referente al Diseño y Desarrollo del producto o servicio, y no a las cláusulas que hacen referencia a la organización como un todo, es decir sus prácticas a nivel de gestión, como los aspectos organizacionales y legales entre otros.

Contrario a lo que se puede percibir, la norma ISO 9001 y las metodologías ágiles no son opuestas, los estudios de caso revisados reportan un 100% de éxito, y numerosos beneficios al comparar el antes y el después de la adopción de prácticas ágiles a un sistema de gestión de la calidad conforme a ISO 9001, sin embargo, es necesario que la literatura reporte más estudios de caso en los que una organización que utilice metodologías ágiles adopte un sistema de gestión de la calidad conforme a ISO 9001.

El estándar ISO 90003:2014 define las directrices para la aplicación de ISO 9001:2008 en la Industria del desarrollo de software, sin embargo, poco o nada se ha encontrado de esta norma en la literatura de las fuentes seleccionadas, dada la naturaleza genérica de la familia de normas ISO 9000, a menudo hay criterios subjetivos por parte de diferentes consultores o incluso auditores cuando se trata de desarrollo de software, precisamente lo que busca el estándar ISO 90003 es establecer un único punto de vista para entender y aplicar ISO 9001 al desarrollo de productos y servicios de software, por esto es necesario que se lleven a cabo más investigaciones y estudios de caso que aporten y enriquezcan la literatura

sobre la utilización de ISO 9003 como una guía para aplicar ISO 9001 a la Industria del desarrollo de software.

Bibliografía.

- [1] "TickITPlus," [Online]. Available: <http://www.tickitplus.org>.
- [2] ICONTEC, *Norma Técnica Colombiana NTC-ISO 9001: Sistema de gestión de la calidad, Requisitos.*, Quito, 2015.
- [3] [Online]. Available: <http://scrumguides.org/scrum-guide.html>. [Accessed 31 Agosto 2016].
- [4] G. Wright, "Achieving ISO 9001 Certification for an XP Company," *Extreme Programming and Agile Methods - XP/Agile Universe 2003*, 2003.
- [5] E. Irrazabal, F. Vásquez, R. Díaz and J. Garzás, "Applying ISO/IEC 12207:2008 with SCRUM and Agile Methods," *Software Process Improvement and Capability Determination* , vol. 155, 2011.
- [6] S. Galván, M. Mora, R. V. O'Connor, F. Acosta and F. Álvarez, "A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process," *Procedia Computer Science*, vol. 64, 2015.
- [7] O. Murri, R. Deias and G. Mughedu, "Assesing XP at a European Internet Company," *IEEE Software*, Vols. 20, Issue 3, 2003.
- [8] B. Bohem and R. Turner, "Balancing Agility And Discipline: Evaluating and Integrating Agile Methods and Plan-Driven Methods," *ICSE'04 Proceedings of the 26th International Conference on Software Engineering*, 2004.
- [9] C. Vriens, "Certifying for CMM Level 2 and ISO 9001 with XP@Scrum," *ADC'03 Proceedings of the Conference on Agile Development*, 2003.
- [10] J. R. Nawrocki, M. Jasiński, B. Walter and A. Wojciechowski, "Combining Extreme Programming with ISO 9000," *EurAsia-ICT 2002: Information and Communication Technology*, vol. 2510, no. Lecture Notes in Computer Science, pp. 786-794, 2002.

- [11] T. Popovic, "Getting ISO 9001 Certified for Software Development Using Scrum and Open Source: A Case Study," *Technicki Vjesnik*, vol. 22, no. 6, pp. 1633-1640, 2015.
- [12] L. T. Heeager, "How can Agile and Documentation-Driven Methods be Meshed in Practice," *Agile Processes in Software Engineering and Extreme Programming*, vol. 179, no. Lecture Notes in Business Information Processing, pp. 62-77, 2014.
- [13] M. Qasaimeh and A. Abran, "Investigation of the Capability of XP to Support the Requirements of ISO 9001 Software Process Certification," *Software Engineering Research, Management and Applications, ACIS International Conference*, 2010.
- [14] B. McMichael and M. Lombardi, "ISO 9001 and Agile Development," *AGILE'07 Proceedings of the AGILE 2007*, pp. 262-265, 2007.
- [15] J. G. A., O. B. d. Silva and P. R. d. Cunha, "Reconciling the irreconcilable: A Software development approach that combines Agile with Formal.," *Proceedings of the 39th Annual Hawaii International Conference on System Sciences*, 2006.
- [16] M. Melis, W. Ambu, S. Pinna and K. Mannaro, "Requirements of an ISO Compliant XP Tool," *Extreme Programming and Agile Processes in Software Engineering*, vol. 3092, no. Lecture Notes in Computer Science, pp. 266-269, 2004.
- [17] W. M. Theunissen, D. G. Kourie and B. W. Watson, "Standards and Agile Software Development," *SAICSIT'03 Proceedings of the 2003 annual research conference of the Suth African institute of computer scientists and information technologists on Enablement through technology*, pp. 178-188, 2003.
- [18] A. F. Bustamante, J. A. Hincapié and G. P. G. Hurtado, "Structure of a Multi-model Catalog for Software Projects Management Including Agile and Traditional Practices," *Trends and Applications in Software Engineering*, vol. 405, no. Advances in Intelligent Systems and Computing, pp. 87-97, 2015.
- [19] L. T. Heeager, "The Agile and the Disciplined Software Approaches: Combinable or just Compatible," *Information Systems Development*,

pp. 35-49, 2013.

- [20] T. Stalhane and G. H. Hanssen, "The Application of ISO 9001 to Agile Software Development," *PROFES'08 Proceedings of the 9th International Conference on Product-Focused Software Process Improvement*, pp. 371-385, 2008.
- [21] K. Petersen and C. Wohlin, "The effect of moving from a plan-driven to an incremental software development approach with agile practices," *Empirical Software Engineering*, vol. 15, no. 6, pp. 654-693, 2010.

Anexo A.

Estudios encontrados en la biblioteca ScienceDirect@ComputerScience. El prefijo SI/NO corresponde a la elección del estudio como relevante.

1. SI | A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process
2. NO | A work product pool approach to methodology specification and enactment
3. NO | Processes versus people: How should agile software development maturity be defined?
4. NO | Quality evaluation for Model-Driven Web Engineering methodologies
5. NO | Approximation of COSMIC functional size to support early effort estimation in Agile
6. NO | Investigating software process in practice: A grounded theory perspective
7. NO | Knowledge Discovery Metamodel-ISO/IEC 19506: A standard to modernize legacy systems
8. NO | Identification and analysis of the elements and functions integrable in integrated management systems

Estudios encontrados en la biblioteca SpringerLink. El prefijo SI/NO corresponde a la elección del estudio como relevante.

1. NO | Using Agile Methods to Implement a Laboratory for Software Product Quality Evaluation
2. NO | The Route to Software Process Improvement in Small- and Medium-Sized Enterprises
3. SI | The Application of ISO 9001 to Agile Software Development
4. NO | The Role of Extreme Programming in a Plan-Driven Organization
5. NO | Prescriptive Process Models
6. NO | Using ISO/IEC 29110 to Harness Process Improvement in Very Small Entities
7. SI | Achieving ISO 9001 Certification for an XP Company
8. NO | An Integrated Framework to Guide Software Process Improvement in Small Organizations
9. NO | Improving Processes and Products
10. NO | A Method Assessment Framework
11. NO | The Tutelkan SPI Framework for Small Settings: A Methodology Transfer Vehicle
12. NO | Assessing Quality

13. SI | A Review of Prominent Work on Agile Processes Software Process Improvement and Process Tailoring Practices
14. NO | HProcessTOOL: A Support Tool in the Harmonization of Multiple Reference Models
15. NO | Bridging the Gap Between SPI and SMEs in Educational Settings: A Learning Tool Supporting ISO/IEC 29110
16. NO | A Comparative Analysis of Agile Maturity Models
17. NO | Software process improvement in small and medium software enterprises: a systematic review
18. NO | Framework of Agile Patterns
19. NO | Investigating the Capability of Agile Processes to Support Life-Science Regulations: The Case of XP and FDA Regulations with a Focus on Human Factor Requirements
20. NO | Software Process Improvement in Small Companies as a Path to Enterprise Architecture
21. SI | How Can Agile and Documentation-Driven Methods be Meshed in Practice?
22. SI | The Agile and the Disciplined Software Approaches: Combinable or Just Compatible?
23. NO | Software Testing
24. NO | A research framework for building SPI proposals in small organizations: the COMPETISOFT experience
25. NO | Lecture 12 Methodology for Information Systems: System Design, Usability, and Evaluation
26. NO | A Process for Driving Process Improvement in VSEs
27. NO | A Security Assurance Model to Holistically Assess the Information Security Posture
28. NO | A Detailed Software Process Improvement Methodology: BG-SPI
29. SI | The effect of moving from a plan-driven to an incremental software development approach with agile practices
30. NO | Overview of Software Processes and Software Evolution
31. NO | Strategy, Business Model and Architecture in Today's Automotive Industry
32. NO | ISO2: A New Breath for the Joint Development of IS and ISO 9001 Management Systems
33. SI | Structure of a Multi-model Catalog for Software Projects Management Including Agile and Traditional Practices
34. NO | Leveraging Reuse-Related Maturity Issues for Achieving Higher Maturity and Capability Levels
35. SI | Requirements of an ISO Compliant XP Tool

36. NO | ESMRMB 2012, 29th Annual Scientific Meeting, Lisbon, Portugal, 4-6 October: EPOSTM Poster / Paper Poster / Clinical Review Poster / Software Exhibits
37. NO | Software Process Improvement Initiatives Based on Quality Assurance Strategies: A QATAM Pilot Application
38. NO | Problems Encountered When Implementing Agile Methods in a Very Small Company
39. NO | 07:00 Planning & Scheduling: P&S
40. NO | In search for a widely applicable and accepted software quality model for software quality engineering
41. NO | Software or Service? That's the Question!
42. NO | Information Architecture for Quality Management Support in Hospitals
43. NO | The Diffusion of Agile Software Development: Insights from a Regional Survey
44. NO | Lean and Agile Project Management: For Large Programs and Projects
45. NO | Safety Critical Software Development – Extending Quality Management System Practices to Achieve Compliance with Regulatory Requirements
46. NO | Medical Software – Issues and Best Practices
47. NO | Supporting Agile Development of Authorization Rules for SME Applications
48. NO | Defining Agile Patterns
49. NO | Software Process Improvement and Certification of a Small Company Using the NTP 291 100 (MoProSoft)
50. NO | Improving Estimates by Hybridizing CMMI and Requirement Engineering Maturity Models – A LEGO Application
51. NO | Experiences with the Certification of a Generic Functional Safety Management Structure According to IEC 61508
52. NO | Safety Critical Software Process Assessment: How MDevSPICE® Addresses the Challenge of Integrating Compliance and Capability
53. NO | Towards optimal software engineering: learning from agile practice
54. NO | An Impact Study of Business Process Models for Requirements Elicitation in XP
55. NO | Harmonizing Software Development Processes with Software Development Settings – A Systematic Approach
56. NO | Software Process Management: A Model-Based Approach
57. SI | Combining Extreme Programming with ISO 9000
58. NO | QUALITUS: An Integrated Information Architecture for the Quality Management System of Hospitals
59. NO | Views from an Organization on How Agile Development Affects Its Collaboration with a Software Development Team

60. NO | Letting Organizations to Find the Correct Way to Start in the Implementation of Software Process Improvements
61. NO | Improvement of Task Management with Process Models in Small and Medium Software Companies
62. NO | Risk orientation in software testing processes of small and medium enterprises: an exploratory and comparative study
63. NO | The Agile Safety Case
64. NO | FACIT-SME: A Semantic Recommendation System for Enterprise Knowledge Interoperability
65. NO | Changing Situational Contexts Present a Constant Challenge to Software Developers
66. NO | An elicitation instrument for operationalising GQM+Strategies (GQM+S-EI)
67. NO | An Integrated Modelling Method for Assessment of Manufacturing Quality Systems Applied to Aerospace Manufacturing Supply Chains⁶⁸.
69. NO | Improving Quality and Cost-Effectiveness in Enterprise Software Application Development: An Open, Holistic Approach for Project Monitoring and Control
70. NO | Agile product development through CAD and rapid prototyping technologies: an examination in a traditional pump-manufacturing company
71. NO | Requirements Engineering and Process Modelling in Software Quality Management— Towards a Generic Process Metamodel
72. NO | Why a CMMI Level 5 Company Fails to Meet the Deadlines?
73. NO | Quality Certification in the Virtual Enterprise
74. NO | Problems and Solutions in Distributed Software Development: A Systematic Review
75. NO | The Many Forms of Process Improvement – Results of an International Survey
76. NO | Trustworthy Software Development
77. NO | Extreme Programming in a Research Environment
78. NO | Software Process Improvement Models Implementation in Malaysia
79. NO | History of Software Engineering
80. NO | Influence of Human Factors in Software Quality and Productivity
81. NO | Panel: Why Agile, Why Lean?
82. NO | Software Quality Perception
83. NO | Software Engineering
84. NO | System Integration
85. NO | Expanding the Horizons of Software Development Processes: A 3-D Integrated Methodology
86. NO | Global Navigation Satellite System based tolling: state-of-the-art

87. NO | Software Engineering
88. NO | Introduction
89. NO | Software engineering technology for the 21st century
90. NO | V-Modell XT und Standards
91. NO | Quality Certification in the Virtual Enterprise: An Objective Tool for Supply Chain Management and Customer Satisfaction
92. NO | Patterns of Trust: Role of Certification for SME Cloud Adoption
93. SI | Practical Aspects of XP Practices
94. NO | Computer-aided design of experiments: an enabler of agile manufacturing
95. NO | Requirements Engineering: Solutions and Trends
96. NO | Quality and Test Management
97. NO | Strategy selection for sustainable manufacturing with integrated AHP-VIKOR method under interval-valued fuzzy environment
98. NO | Derivation of Green Metrics for Software
99. NO | Capability Maturity Model Integrated for Ship Design and Construction
100. NO | Dimensional Metrology for Manufacturing Quality Control
101. NO | Automated bug assignment: Ensemble-based machine learning in large scale industrial contexts
102. NO | A Hybrid VFT-GQM Method for Developing Performance Criteria and Measures
103. NO | Organisational Knowledge in IS: A Tool to Support is Intervention Methodologies
104. NO | Methods of Assessing the Level of the Technology Innovation and Polish Innovativeness in Years 2010–2014
105. NO | Ensuring Success and Quality through the Use of Standards in Team Projects: Students' Perceptions
106. NO | Automating Information and Technology Services
107. NO | A Palette of Lean Indicators to Detect Waste in Software Maintenance: A Case Study
108. NO | Improving by Involving: A Case Study in a Small Software Company
109. NO | Software quality improvement: a model based on managing factors impacting software quality
110. NO | Software Process Management: Practices in China
111. NO | Software Process in Practice: A Grounded Theory of the Irish Software Industry
112. NO | Decision-aiding tools in innovative product development contexts
113. NO | An empirical study on lean awareness and potential for lean implementations in Qatar industries
114. NO | Enabling Competitive Design of Next Generation Reconfigurable Manufacturing Enterprises

115. NO | Factors that Influence the Productivity of Software Developers in a Developer View
116. NO | Method for Quality Appraisal in Supply Networks
117. NO | Global Software Development and Quality Management: A Systematic Review
118. NO | Keiran J. Dunne and Elena S. Dunne (eds.): Translation and localization project management: the art of the possible
119. NO | An Empirical Study of Lead-Times in Incremental and Agile Software Development
120. NO | Experience Reports
121. NO | The Four “P”s of Enterprise ICT
122. SI | Comparison of CMM Level 2 and eXtreme Programming
123. NO | Involving Users to Improve the Level of Their Satisfaction from a Software Product Designed for Public Organization
124. NO | A Method to Support the Adoption of Reuse Technology in Large Software Organizations
125. NO | Research and Development: Business into Transfer Information and Communication Technology
126. NO | Managing Evolving Business Workflows through the Capture of Descriptive Information
127. NO | IT impact on talent management and operational environmental sustainability
128. NO | Engineering change: an overview and perspective on the literature
129. NO | A Web-based interactive advisor for assembly line balancing
130. NO | Approach to SAP Testing and QM Best Practices
131. NO | Planning Business Process Management Implementation by a Human Resources Development Support Initiative
132. NO | From offshore outsourcing to insourcing and partnerships: four failed outsourcing attempts
133. NO | Systementwicklung und Lifecycle Management
134. NO | Value-Based Knowledge Management: the Contribution of Group Processes
135. NO | Management System for Manufacturing Components Aligned with the Organisation IT Systems
136. NO | Managing Intellectual Capital — via E-Learning — at Cisco
137. NO | Rule-Based Detection of Process Conformance Violations in Application Lifecycle Management
138. NO | Motivation and Introduction
139. NO | Quality and the Service Level Agreement
140. NO | Quality and the Service Level Agreement

- 141. NO | Acceptance Testing HTML
- 142. NO | Information technology, the organizational capability of proactive corporate environmental strategy and firm performance: a resource-based analysis
- 143. NO | Does Use of Development Model Affect Estimation Accuracy and Bias?
- 144. NO | Do We Need New Management Perspectives for Software Research Projects?
- 145. NO | Reducing hardware risks in the development of Telematic Rescue Assistance Systems: A methodology
- 146. NO | Résumés
- 147. NO | Information Sources and Their Importance to Prioritize Test Cases in the Heterogeneous Systems Context
- 148. NO | Developments of Manufacturing Systems with a Focus on Product and Process Quality
- 149. NO | An empirically validated simulation for understanding the relationship between process conformance and technology skills
- 150. NO | Die Aufgaben des Informationsmanagements
- 151. NO | Multiple Objective Decision Making
- 152. NO | A Framework for Coping with Process Evolution
- 153. NO | Software Process Improvement from a Human Perspective
- 154. NO | Enabling Traceability in the Wine Supply Chain
- 155. NO | Distributed Project Management
- 156. NO | An empirically based terminology and taxonomy for global software engineering
- 157. NO | Supporting Enterprise Changes Using Actor Performance Assessment
- 158. NO | Concept for Quality Control Management Services in Distributed Design Networks – Conceptual Paper
- 159. NO | Infosys Technologies: improving organizational knowledge flows
- 160. NO | Efficient analysis, handling and use of customer complaints
- 161. NO | Web-Based Support for Early Supplier Involvement in New Product Development
- 162. NO | Pitfalls in Remote Team Coordination: Lessons Learned from a Case Study
- 163. NO | Seibold, H. IT-Risikomanagment
- 164. NO | The Pre-application System of Real Estate Registration
- 165. NO | Management der Informationssysteme
- 166. NO | Nuclear Crisis Use-Case Management in an Event-Driven Architecture
- 167. NO | Emergent Case Management for Ad-hoc Processes: A Solution Based on Microblogging and Activity Streams
- 168. NO | Sicherheitsarchitektur
- 169. NO | Collaborative Systems in Crisis Management: A Proposal for a

Conceptual Framework

- 170. NO | Event-cloud platform to support decision-making in emergency management
- 171. NO | Exploring the Role of Usability in the Software Process: A Study of Irish Software SMEs
- 172. NO | An integrated analytic network process with mixed-integer non-linear programming to supplier selection and order allocation
- 173. NO | Schlüsselfaktor IT-Sicherheit
- 174. NO | Sicherheitsarchitektur
- 175. NO | Business Rules, Constraints and Simulation for Enterprise Governance
- 176. NO | On Representational Issues About Combinations of Classical Theories with Nonmonotonic Rules
- 177. NO | Cost Management Practices in Collaborative Product Development Processes
- 178. NO | Managing Service-Based EAI Architectures Evolution Using a Formal Architecture-Centric Approach
- 179. NO | Software for the Changing E-Business
- 180. NO | Grundlagen des Software Engineering
- 181. NO | Multivariate Statistical and Computational Intelligence Techniques for Quality Monitoring of Production Systems
- 182. NO | Grundlagen des Software Engineering
- 183. NO | Zielverwirklichungsmanagement ZVM
- 184. NO | Refining code ownership with synchronous changes
- 185. NO | “Leave the Programmers Alone”- A Case Study
- 186. NO | Engineering change
- 187. NO | Grundlagen des Software Engineering
- 188. NO | The Effect of New Standards on the Global Movement Toward Usable Medical Devices
- 189. NO | Heisig, P.; Mertins K. (Hrsg.) Integration von Wissensmanagement in Geschäftsprozesse
- 190. NO | Overview of Software Process Models and Descriptive Criteria for their Analysis
- 191. NO | Specification Model for the Development and Operation of a Virtual Company in the Aerospace Industry
- 192. NO | Supply Chain Coordination and IT: The Role of Third Party Logistics Providers
- 193. NO | Sicherheit
- 194. NO | Le quattro “P” dell’azienda ICT
- 195. NO | Software-bezogene Vertragsleistungen
- 196. NO | Standards zur IT-Sicherheit und Regulatory Compliance

- 197. NO | Kneuper, R. CMMI
- 198. NO | Globalization
- 199. NO | Motivazioni e Introduzione
- 200. NO | Industrie 4.0 bestimmt die Arbeitswelt der Zukunft
- 201. NO | Reifegradmodelle und Prozessverbesserung
- 202. NO | Systementwicklung
- 203. NO | Die V-Modell XT Grundlagen
- 204. NO | Managing the ITIL
- 205. NO | Software-Hardware-Schnittstellen
- 206. NO | Grundlagen
- 207. NO | Fallstudie numetris AG
- 208. NO | Vorgehensmodelle und Standards der Entwicklung
- 209. NO | User Interface Engineering: Einleitung
- 210. NO | Erstellung von Software
- 211. NO | Risikolexikon
- 212. NO | Ergebnistypen der Pressearbeit
- 213. NO | Software Process Assessment and Improvement
- 214. NO | Foundations
- 215. NO | A systematic review of software usability studies
- 216. NO | Safety Management Report (SMR)
- 217. NO | Governance and Management of Software Processes
- 218. NO | Considerations about quality in model-driven engineering
- 219. NO | The Agile Safety Plan for Signalling Systems
- 220. NO | Service-Centered Operation Methodology (MOCA) Application
Supported by Computer Science to Improve Continuously Care Quality in Public
Services
- 221. NO | The Safety Case: Introduction and Definition of the System
The Safety Case: Introduction and Definition of the System
- 222. NO | IT Processes
- 223. NO | Ganzheitliche Prozessgestaltungs- und -optimierungskonzepte

Estudios encontrados en la biblioteca IEEE Xplore. El prefijo SI/NO corresponde a la elección del estudio como relevante.

- 1. NO | A Quality Attributes Evaluation Method for an Agile Approach
- 2. NO | Requirements Engineering Quality Revealed through Functional Size Measurement: An Empirical Study in an Agile Context
- 3. NO | An agile software quality framework lacking
- 4. NO | Automatic code generation from a UML model to IEC 61131-3 and system configuration tools

5. NO | An IEC61499 Execution Environment for an aJile-based Field Device
6. NO | Agile Development in a Medical Device Company
7. NO | rasdaman: Array Databases Boost Spatio-Temporal Analytics
8. NO | Hybrid user centered development methodology: The practical case of courseware sere
9. NO | 9.1 A self-calibrating NFC SoC with a triple-mode reconfigurable PLL and a single-path PICC-PCD receiver in 0.11um CMOS
10. SI | Reconciling the Irreconcilable? A Software Development Approach that Combines Agile with Formal
11. SI | Balancing agility and discipline: evaluating and integrating agile and plan-driven methods
12. NO | Software Metrics for Agile Software Development
13. NO | OpenUP/MMU-ISO 9241-210. Process for the Human Centered Development of Software Solutions
14. NO | Digital Audio/Video Channel Recorder
15. NO | IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements. Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications
16. NO | An agile development process for petrochemical safety conformant software
17. NO | Using XP in Telecommunication Software Development
18. NO | Toward improving Agile Mantema: Measurement, control and evaluation of maintenance projects in SMEs
19. NO | Communication types for manufacturing systems. A proposal to distributed control system based on IEC 61499
20. NO | Formal modeling and verification in the software engineering framework of IEC 61499: a way to self-verifying systems
21. NO | State of the art and future of research in software process improvement
22. NO | Development of a software process framework to assist organizations developing mobile medical apps
23. NO | Development of manufacturing automation systems through object oriented concepts and international standards
24. NO | Using Personas to Support the Goals in User Stories
25. NO | AGILEUX Model: Towards a Reference Model on Integrating UX in Developing Software Using Agile Methodologies
26. NO | Toward assessing the implementation and use of agile methodologies in SMEs: An analysis of the actual agile methodologies assessment tools
27. NO | Bridge Methods: Complementary Steps Integrating Agile Development Tools and Methods with Formal Process Methodologies

28. NO | An information security policy in converged network environment
29. NO | A design pattern for holonic manufacturing system in the IEC61499-based model-view-controller framework
30. SI | Investigation of the Capability of XP to Support the Requirements of ISO 9001 Software Process Certification
31. NO | Direct verbal communication as a catalyst of agile knowledge sharing
32. NO | Slicing the V-Model -- Reduced Effort, Higher Flexibility
33. NO | ISO standards for software user documentation
34. SI | Systems and software engineering -- Developing user documentation in an agile environment
35. SI | An integrative approach to project management in a small team developing a complex product
36. NO | Agile standardization by means of PCE Requests
37. NO | The design of an Object-Oriented Embedded Platform for Substation Data Integration
38. NO | Agile process tailoring and problem analysis (APPLY)
39. NO | Modeling requirements: The customer communication
40. NO | Land Voted 2008 Computer Society President-Elect
41. NO | IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications
42. NO | A reference method for airborne software requirements
43. NO | What Do We Know about Agile Software Development?
44. NO | IEEE Draft Standard for Software and Systems Engineering--Software Testing--Part 2: Test Process
45. NO | Enabling agile manufacturing through reconfigurable control solutions
46. NO | How the FBI Learned to Catch Bad Guys One Iteration at a Time
47. NO | PDM data classification from step - An object oriented string matching approach
48. NO | Agile Software Assurance: An Empirical Study
49. NO | A Process Based Unification of Process-Oriented Software Quality Approaches
50. NO | Measuring the Structural Quality of Business Applications
51. NO | COSME: A distributed control platform for communicating machine tools in Agile Manufacturing Systems
52. NO | Ahaa --agile, hybrid assessment method for automotive, safety critical smes
53. SI | ISO 9001 and Agile Development
54. NO | Agile modeling of an evolving ballistic missile defense system with

Object-Process Methodology

55. NO | Implementation of MOPROSOFT level I and II in software development companies in the colombian caribbean, a commitment to the software product quality region
56. NO | Design Alternatives in the IEC 61499 Function Block Model
57. NO | An Agile methodology for Manufacturing Control Systems development
58. NO | Part 3: Carrier Sense Multiple Access With Collision Detect on (CSMA/CD) Access Method and Physical Layer Specifications
59. NO | PRP and HSR for High Availability Networks in Power Utility Automation: A Method for Redundant Frames Discarding
60. NO | Growing a Build Management System from Seed
61. NO | Systems engineering processes for agile software development
62. NO | ISO/IEC/IEEE International Standard for Ethernet
63. NO | Efficient Energy Delivery Management for PHEVs
64. NO | PCA based cost estimation model for agile software development projects
65. NO | Adept: A Unified Assessment Method for Small Software Companies
66. NO | III-Phase Verification and Validation of IEC Standard Programmable Logic Controller
67. NO | Integrating Legacy Systems within The Service-oriented Architecture
68. NO | Standards, Agility, and Engineering
69. NO | Historical Data Repositories in Software Engineering: Status and Possible Improvements
70. NO | International Standards for Information Development and Content Management
71. NO | Design and implementation of a prototype control system according to IEC 61499
72. NO | MAnGve maturity model (M3): A proposal for a doctoral thesis
73. NO | Certification and regulatory challenges in medical device software development
74. NO | Experiences with model-centred design methods and tools in safe robotics
75. NO | Qualitative comparison of agile and iterative software development methodologies
76. NO | Dynamic reconfiguration of distributed control applications with reconfiguration services based on IEC 61499
77. NO | IEEE Standard for Information technology--Telecommunications and information exchange between systems--Local and metropolitan area networks--Specific requirements Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications
78. NO | IEC 61499 execution model based on life cycle of function blocks
79. NO | Compliance of CMMI Process Area with Specification Based

Development

80. NO | Control System Software Design Methodology for Automotive Industry
81. NO | Improving the User Story Agile Technique Using the INVEST Criteria
82. NO | Architectural Naming, a Secret of Agile Thinking
83. NO | Position Statement: Advances and Challenges of Software Engineering
84. SI | Assessing XP at a European Internet company
85. NO | Notice of Violation of IEEE Publication Principles Spatial data based on SOA
86. NO | Defining Agile Software Quality Assurance
87. SI | Certifying for CMM Level 2 and ISO9001 with XP@Scrum
88. NO | Agent-based control system for next generation manufacturing
89. NO | A Web-based Tool for Automatizing the Software Process Improvement Initiatives in Small Software Enterprises
90. NO | Agile testing concepts based on keyword-driven testing for industrial automation systems
91. NO | Evaluation of software engineering management best practices in the Western Cape
92. NO | Empirical Studies on Quality in Agile Practices: A Systematic Literature Review
93. NO | Experience teaching software project management in both industrial and academic settings
94. NO | Definition of a Hybrid Measurement Process for the Models ISO/IEC 15504-ISO/IEC 12207:2008 and CMMI Dev 1.3 in SMEs
95. NO | Implementation of the ISO/IEC 29110 standard in agile environments: A systematic literature review
96. NO | P26515 FDIS, May 2018 - ISO/IEC/IEEE Approved Draft International Standard - Systems and Software Engineering -- Developing Information for Users in an Agile Environment
97. NO | IEEE/ISO/IEC Draft Standard for Software and systems engineering -- Developing user documentation in an agile environment
98. NO | ISO/IEC/IEEE Draft International Standard - Systems and Software Engineering -- Developing Information for Users in an Agile Environment
99. NO | Framework of Multiuser Satisfaction for Assessing Interaction Models Within Collaborative Virtual Environments
100. NO | Scrum+: An agile guide for the global software development (GSD) multi-model project management
101. NO | From ISA 88/95 meta-models to an OPC UA-based development tool for CPPS under IEC 61499
102. NO | IT governance program and improvements in Brazilian small business: Viability and case study

- 103. NO | Designing Automation Distributed Systems Based on IEC-61499 and UML
- 104. NO | Innovation Welcome: An Agile Approach to Model-Based Development of Safety-Critical Embedded Systems
- 105. NO | Enabling an automation architecture of CPPs based on UML combined with IEC-61499
- 106. NO | Towards Continuous Security Compliance in Agile Software Development at Scale
- 107. NO | DevOps in Regulated Software Development: Case Medical Devices
- 108. NO | Online reconfiguration of automatic production line using IEC 61499 FBs combined with MAS and ontology
- 109. NO | Developing the stakeholder requirements definition process — A journey of customization
- 110. NO | Software process improvement assesment for multimodel environment tool to diagnose an organization
- 111. NO | Improve software quality through practicing DevOps
- 112. NO | The Metrologist's place is by the machines!

Estudios encontrados en la biblioteca ACM Digital Library. El prefijo SI/NO corresponde a la elección del estudio como relevante.

- 1. SI | Standards and agile software development
- 2. NO | Using agile methods to implement a laboratory for software product quality evaluation
- 3. NO | Procedural Assessment Process of Software Quality Models Using Agility
- 4. NO | Agile maturity model: analysing agile maturity characteristics from the SPICE perspective
- 5. NO | Comparison of Plan-driven and Agile Project Management Approaches: Theoretical Bases for a Case Study in Estonian Software Industry
- 6. NO | Agile, CMMI®, RUP®, ISO/IEC 12207...: is there a method in this madness?
- 7. NO | Empirical Studies on Quality in Agile Practices: A Systematic Literature Review
- 8. NO | A work product pool approach to methodology specification and enactment
- 9. NO | Knowledge Discovery Metamodel-ISO/IEC 19506: A standard to modernize legacy systems
- 10. NO | Towards approximating COSMIC functional size from user requirements in agile development processes using text mining
- 11. NO | Processes versus people
- 12. NO | Approximation of COSMIC functional size to support early effort

estimation in Agile

13. NO | Cohesion and coupling metrics for ontology modules
14. NO | Quality evaluation for Model-Driven Web Engineering methodologies
15. NO | Process Assessment and Improvement: A Practical Guide for Managers, Quality Professionals and Assessors, Second Edition
16. NO | Video coding standards: AVS China, H.264/MPEG-4 PART 10, HEVC, VP6, DIRAC and VC-1
17. NO | Managing and Leading Software Projects
18. NO | Scaling for agility: A reference model for hybrid traditional-agile software development methodologies
19. NO | An industry experience report on managing product quality requirements in a large organization
20. NO | Using weekly open defect reports as an indicator for software process efficiency: theoretical framework and a longitudinal automotive industrial case study

Anexo B.

Título	Achieving ISO 9001 Certification for an XP Company
Publicación	Extreme Programming and Agile Methods – XP/Agile Universe 2003
Año	2003
Autor(es)	Graham Wright
Resumen	Generalmente se asume que la certificación ISO 9001 es incompatible con métodos de desarrollo ágil, particularmente eXtreme Programming. De cualquier forma es posible obtener la certificación en una manera compatible con XP y que no reduce la agilidad. La clave para esto es hacer de la documentación, monitoreo de procesos y trazabilidad requerida para la certificación una salida natural del proceso de desarrollo más que un producto artificial creado puramente para satisfacer aquellos requerimientos. Este estudio describe el éxito en la certificación de una

	compañía que utiliza XP.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001:2000 eXtreme Programming
Métodos de armonización utilizados	TickIT, para la implementación de ISO 9001:2000 al desarrollo software. Mapping entre XP e ISO 9001 (Tabla comparativa).
Nivel de detalle de la propuesta	El autor describe con alto nivel de detalle el proceso de desarrollo de software, haciendo referencia a las prácticas de XP, las herramientas utilizadas para la gestión del proyecto y las ideas equivocadas respecto a ISO 9001 y su certificación
Tipo de evaluación	Estudio de Caso. Organización: Workshare
Aspectos a destacar	
<p>El autor hace énfasis en repetidas ocasiones que XP no tiene ningún inconveniente en cumplir con el estándar ISO 9001:2000.</p> <p>En orden de mantener la agilidad, es esencial que el hecho de demostrar la calidad y mejora en el proceso recaiga en las salidas de los procesos de XP, y no en artefactos creado puramente para cumplir con el estándar.</p> <p>El autor afirma que para ellos fue esencial utilizar una “Herramienta de gestión” a lo que el denomina “Virtual White Board” En la que se pudiera hacer seguimiento a las historias de usuario desde que se creaban, se documentaban, se modificaban, se asignaban, se implementaban, se testeaban y se establecían como “DONE”.</p> <p>El autor afirma que para ellos fue clave el hecho de que antes de comenzar a prepararse para la certificación, la empresa ya estaba acostumbrada a la documentación en los procesos debido a que su equipo era “grande” (30 desarrolladores, 10 product managers entre otros roles adicionales distribuidos en 3</p>	

ubicaciones), y para mantener un orden y control era necesario documentación semi-formal.

El autor afirma que es clave que en el desarrollo de software se hagan pruebas en todas las etapas, a los auditores les gustan las pruebas y es importante que el código sea desarrollado bajo algunos estándares para legibilidad y entendimiento.

Es necesario que la organización designe a un “process compliance manager” experto en el estándar para que continuamente verifique el cumplimiento de las cláusulas.

El autor concluye que para ellos la principal tarea fue elaborar los manuales y políticas de calidad con el nivel de detalle necesario para cumplir con el estándar, además de elaborar sub-manuales para cada departamento de la organización.

De cualquier manera es muy importante entender que mapear ISO 9001 a XP o a cualquier otra metodología ágil no es el principal objetivo en la búsqueda de la certificación, más que eso es necesario probar que el proceso es monitoreado, medido y continuamente mejorado.

Título	A Compliance Analysis of Agile Methodologies with the ISO/IEC 29110 Project Management Process
Publicación	Procedia Computer Science Volume 64, 2015
Año	2015
Autor(es)	Sergio Galván, Manuel Mora, Rory V. O’Connor, Francisco Acosta, Francisco Álvarez
Resumen	Estándares de procesos de software (por ejemplo ISO/IEC 12207, ISO/IEC 15504) y modelos (por ejemplo CMMI) proveen un conjunto de buenas prácticas y directrices para mejorar la calidad del proceso de software y el producto resultante de ese proceso. De cualquier forma, no prescriben

	<p>una metodología de desarrollo en particular (por ejemplo RUP, MSF), y así los equipos de desarrollo de software enfrentan un problema de conformidad entre la metodología de desarrollo seleccionada y un estándar en particular o modelo que buscan adoptar. En esta investigación, el asunto en particular de conformidad de las metodologías de desarrollo ágil (SCRUM, XP y UPEDU) y el nuevo estándar ISO/IEC 29110 es estudiado. Porque el nuevo estándar está enfocado en el proceso de software en compañías de desarrollo de software muy pequeñas o equipos de proyectos de software muy pequeños en el rango de 1 a 25 personas, y el las metodologías de desarrollo de software ágil (ASDMs) son primordialmente para estos objetivos, este estudio es importante. El estándar ISO/IEC 29119 contiene dos procesos: Gestión de proyecto e Implementación del Software. Este estudio está enfocado en el primer proceso. Los principales hallazgos indican que las metodologías UPEDU y SCRUM presentan un alto nivel de conformidad con el proceso de gestión de proyecto de ISO/IEC 29110, mientras que XP tiene un nivel moderado. Así, los equipos de desarrollo de software interesados en conseguir conformidad con el proceso de Gestión de Proyecto de ISO/IEC 29110 pueden contar con dos ASDMs. De cualquier forma, un estudio completo de conformidad (Con ambos Gestión de proyecto e Implementación de Software) aún hace falta.</p>
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 29110 SCRUM XP UPEDU
Métodos de armonización utilizados	Mapping entre roles, artefactos y actividades utilizando IDEF0 para la notación representación gráfica de los procesos
Nivel de detalle de la propuesta	Alto nivel, correspondiente únicamente al proceso de “Project Management” del estándar ISO/IEC 29110, el

	proceso de “Software Implementation” No es analizado
Tipo de evaluación	Validación de expertos respecto a los diagramas de IDEF0 representando el proceso de Project Management de ISO/IEC 29110 y las comparaciones respecto a roles, actividades y artefactos con las metodologías ágiles SCRUM, XP y UPEDU
Aspectos a destacar	
<p>Respecto a ISO/IEC 29110 y sus dos procesos principales.</p> <p>Project Management: Busca establecer y llevar a cabo las tareas de la implementación del software, las cuales cumplirán los objetivos del proyecto de acuerdo a la calidad, costo y tiempo esperado. Incluye cuatro actividades: planeación, control, ejecución y cierre.</p> <p>Software Implementation: Busca analizar sistemáticamente, diseño, construcción, integración y pruebas del producto software procesado de acuerdo a los requerimientos especificados. Incluye seis actividades: iniciación, análisis, diseño, construcción, pruebas y entrega.</p> <p>El autor afirma: “En cuánto a los roles podemos observar que SCRUM tiene una alta conformidad en comparación con ISO/IEC 29110. Mientras que, UPEDU y XP tienen una conformidad moderada. Para esta investigación consideramos tres roles porque son los descritos en los documentos oficiales de ISO/IEC 29110. Podemos concluir que SCRUM tiene una buena organización en el enfoque de roles y satisface muchas cláusulas en ISO/IEC 29110 en comparación con UPEDU y XP”.</p> <p>El autor, respecto a las actividades afirma: “En conclusión, SCRUM y XP son metodologías ágiles más famosas que UPEDU, pero hemos encontrado que UPEDU muestra un comportamiento más relevante en aspectos del proceso de Project Management puesto por ISO/IEC 29110 que las otras dos metodologías ágiles”.</p> <p>El autor, respecto a los artefactos afirma: “En conclusión para esta categoría de conformidad de los artefactos, UPEDU se encontró con un alto nivel de conformidad. Los 8 artefactos oficiales reportados en ISO/IEC 29110 son muy similares en esencia, con los artefactos reportados en UPEDU. En segundo lugar podemos ubicar a SCRUM con algunos aspectos que deben ser mejorados como el</p>	

“Repositorio de proyecto”, “Configuración del Software” y “Registro de Aceptación”. Finalmente, XP tiene algunas falencias (Evaluaciones de bajo nivel) en las mismas categorías que SCRUM. XP necesita explicaciones más claras y organización para sus artefactos.”

Finalmente el autor concluye: “Hemos encontrado que las metodologías ágiles UPEDU y SCRUM pueden ser considerados con un alto nivel de conformidad mientras que XP presenta un nivel moderado.” “En la práctica se puede contar con UPEDU o SCRUM como las metodologías ágiles casi ya disponibles conforme al proceso de Project Management de ISO/IEC 29110”

Título	Applying ISO/IEC 12207:2008 with SCRUM and Agile Methods
Publicación	Software Process Improvement and Capability Determination Volume 155 2011
Año	2011
Autor(es)	Emmanuel Irrazabal, Felipe Vásquez, Rafael Díaz, Javier Garzás
Resumen	Actualmente y en los años recientes, varias iniciativas internacionales orientadas específicamente a poner juntos pequeñas y medianas empresas, procesos y metodologías ágiles han sido identificadas, de esta manera diferentes estudios han identificado el mapeo entre metodologías ágiles y el desarrollo de software, modelos de proceso como CMMI-DEV e ISO/IEC 12207, pero los estudios relacionados a ISO/IEC 12207 son basados en la versión de 1995. Por lo tanto este trabajo se enfoca en la relación entre prácticas ágiles, específicamente SCRUM, y un subconjunto de procesos desde la versión 2008 del estándar ISO/IEC 12207. SCRUM es una de las metodologías ágiles más populares y es un proceso iterativo incremental. Estas dos

	<p>características significan dividir el proyecto en fases o iteraciones y entrega incremental del proyecto. Las relaciones indicadas en el trabajo son obtenidas del análisis de trabajos previos y la experiencia de consultoría en 25 empresas que cumplen con el estándar implementando metodologías ágiles. El principal propósito del estudio es conocer la extensión a la que las prácticas ágiles ayudan en la implementación de prácticas indicadas en este modelo de proceso.</p>
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO/IEC 12207:2008 SCRUM
Métodos de armonización utilizados	Mapping utilizando tablas de comparación entre SCRUM y un subconjunto de procesos específicos de ISO/IEC 12207:2008 (Muy específico)
Nivel de detalle de la propuesta	Muy alto nivel, no hay información relacionada a estudio de caso ni encuestas, sólo menciona que hay un consultor que visitó 25 empresas pero sólo es un dato
Tipo de evaluación	“Estudios de caso”
Aspectos a destacar	
<p>El autor concluye según sus tablas de mapping utilizadas que SCRUM sirve para satisfacer un 83% del Project Planning Process de la norma ISO/IEC 12207 y un 75% del Project Assessment and Control Process de ISO/IEC 12207. También asegura que otros 7 procesos de ISO/IEC 12207 pueden ser parcialmente satisfechos utilizando SCRUM.</p> <p>El autor hace referencia a que ISO/IEC 12207:2008 no pone restricciones en el método utilizado para implementar los procesos, y que las metodologías ágiles pueden ser complementarias a la norma.</p> <p>Desde mi punto de vista es un artículo muy mal escrito, con mala argumentación, malas comparaciones incluso hace referencia a un SCRUM Manager, que no</p>	

conozco y también menciona que en la reunión de planeación del sprint se decide si hay lanzamiento del incremento del producto.

Considero que es un artículo muy mal redactado y con pocas bases en lo que SCRUM es, también considero que las comparaciones hechas entre SCRUM e ISO/IEC 12207:2008 no tienen lugar en algunos lugares del artículo, por ejemplo el porcentaje de cumplimiento de SCRUM en las prácticas de Project Management de ISO/IEC 12207:2008 se ve reducido porque SCRUM “No utiliza recursos de otros proyectos o evidencia de proyectos anteriores”. En mi criterio esa comparación no tiene una base debido a que SCRUM es un marco de trabajo ágil para proyectos complejos, no es un estándar que una organización pueda adoptar. Por supuesto que una organización que utilice SCRUM puede re utilizar experiencia de otros proyectos anteriores pero no es necesario que en la documentación oficial de SCRUM diga “La organización puede o debe utilizar recursos de otros proyectos cuando sea posible”.

Título	Assessing XP at a European Internet Company
Publicación	IEEE Software, Volume 20, Issue 3
Año	2003
Autor(es)	Orlando Murri, Roberto Deias, Giampiero Mughedu
Resumen	Fst. Una pequeña empresa con aproximadamente 160 empleados. Diseña, construye y provee servicios de internet. Nuestras áreas de experticia son construir complejos portales con acceso multicanal y estrictos requerimientos de seguridad, soportando firmas digitales y el manejo de varas clases de transacciones financieras. Fst no es una casa de software, se enfoca en construir asociaciones con sus clientes al proveer soluciones a sus necesidades específicas.

Descripción	
Modelo de calidad y/o metodología ágil utilizadas	XP
Métodos de armonización utilizados	No aplica.
Nivel de detalle de la propuesta	No aplica.
Tipo de evaluación	Estudio de caso
Aspectos a destacar	
<p>El artículo es muy bueno respecto a la enorme información de retroalimentación que provee para una empresa que quiera adoptar la metodología de desarrollo ágil XP.</p> <p>Sin embargo, menciona la certificación ISO 9001 simplemente como un dato al inicio del estudio, pero no vuelve a relacionar el término en ninguna parte del documento.</p>	

Título	Balancing Agility and Discipline: Evaluating and Integrating Agile and Plan-Driven Methods
Publicación	ICSE'04 Proceedings of the 26th International Conference on Software Engineering
Año	2004
Autor(es)	Barry Boehm, Richard Turner

Resumen	El rápido cambio y el incremento en el software conllevan críticamente al desarrollo exitoso y requiere que las organizaciones balanceen la agilidad y la disciplina de sus procesos claves. El surgimiento de los métodos ágiles en la comunidad del software está elevando las expectativas de los clientes y la gestión, pero los métodos tienen problemas y su compatibilidad con métodos tradicionales planificados como aquellos representados por CMMI, ISO-15288, y UK-DefStan-00-55 son ampliamente inexplorados.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	El autor se refiere a dos grandes grupos de estrategias de desarrollo. Métodos ágiles Métodos dirigidos por la planificación.
Métodos de armonización utilizados	Risk based Spiral Model Anchor Points
Nivel de detalle de la propuesta	Muy alto nivel, muy general
Tipo de evaluación	No hay información de estudio de caso aunque el autor dice haber realizado 2.
Aspectos a destacar	
<p>El artículo es muy corto (2 páginas).</p> <p>El autor afirma que la agilidad y la disciplina de los métodos dirigidos por el plan son complementarias.</p> <p>Para destacar hay un trozo del texto.</p> <p>“Discipline is the foundation for any successful endeavor. Athletes train, musicians practice, craftsmen perfect techniques, and engineers apply processes. Without these basic skills there</p>	

may be an occasional success using natural talent, but professional consistency and long term prospects are limited. The strength and comfort which come from discipline support the endeavor when things are difficult, when the body or mind is under the weather, or when something new or unexpected arises and a response is required. Discipline creates well-organized memories, history, and experience.

Agility is the counterpart of discipline. Where discipline ingrains and strengthens, agility releases and invents. It allows the athlete to make the unexpected play, musicians to improvise and ornament, craftsmen to evolve their style, and engineers to adjust to changing technology and needs. Agility applies memory and history to adjust to new environments, react and adapt, take advantage of unexpected opportunities, and update the experience base for the future.

Every successful venture in a changing world requires both agility and discipline. This is as true in business and software development as it is in sports and art. In his bestseller Good to Great [3], Jim Collins presents a two dimensional scale that describes the characteristics of successful businesses. One dimension is a culture of discipline and the other, an ethic of entrepreneurship. In our context, we equate entrepreneurship with agility. If one has strong discipline without agility, the result is inflexible hierarchy and stagnation. Agility without discipline leads to the heady, unencumbered enthusiasm of a start-up company—before it has to turn a profit. Great companies, and great software projects, have both in measures appropriate to their goals and environment.”

Título	Certifying for CMM Level 2 and ISO 9001 with XP@Scrum
Publicación	ADC '03 Proceedings of the Conference on Agile Development

Año	2003
Autor(es)	Christ Vriens
Resumen	<p>Este informe de experiencia describe el camino seguido de conseguir la certificación tanto para CMM Level 2 e ISO 9001:2000 en una escala de tiempo de 2 años usando metodologías ágiles.</p> <p>Discutimos porque la combinación seleccionada de eXtreme Programming (XP) y Scrum como la base para nuestro proceso de desarrollo de software y cuál “ceremonia” tuvimos que agregar en orden de satisfacer los requerimientos de CMM L2 e ISO 9001. También nuestros mayores desafíos hasta el momento son descritos, y la forma como tratamos de resolverlos.</p> <p>Por lo tanto queremos compartir un número de inconvenientes y experiencias.</p>
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	CMM Level 2 ISO 9001:2000 XP@Scrum
Métodos de armonización utilizados	No especificado.
Nivel de detalle de la propuesta	Medio nivel de detalle, el autor menciona aspectos de alto nivel como también detalles de los estándares de codificación y las experiencias en cuánto la forma de trabajo del equipo de desarrollo
Tipo de evaluación	Estudio de caso
Aspectos a destacar	

El autor afirma que para conseguir el éxito en las certificaciones de ISO 9001:2000 y CMM Level 2 se identificaron puntos de inicio para maximizar los factores de éxito:

Mínima Burocracia en orden de construir tanto como sea posible.

No hacer “Reinvención de la rueda”, al usar prácticas que ya han sido probadas bajo ciertas circunstancias de entorno.

Utilizar un estado del arte en el proceso de desarrollo acompañado con las herramientas adecuadas que lo soportan.

Respecto a la metodología ágil XP el autor afirma que se obtuvo una mayor satisfacción tanto para los programadores como para los clientes.

Fue difícil para el equipo conseguir buenas metáforas del sistema.

Respecto al “Pair Programming” el autor comparte las reglas establecidas en el equipo:

- Se usó Pair Programming en todo el código de producción y los tests relacionados
- Cada nueva tarea debe comenzar con una pequeña etapa de diseño. Si la conclusión de esta sesión es que el código requerido es relativamente simple o de rutina, se permite programar sólo.
- Los reportes de problemas son siempre resueltos en pares ya que en la mayoría de casos los problemas ocurren sobre interfaces de dos o más módulos
- La documentación puede ser escrita por una persona y siempre será revisada usando la inspección de Fagan

El autor afirma: Pair programming es una actividad que consume mucha energía. Tratamos de limitarla a máximo 6 horas al día. Pero da un sentimiento de confianza al final del día respecto al hecho de que los pares han aprobado el código. También es bastante apreciado que es una forma de prevenir lesiones por la constante tensión.

El autor también menciona: Tanto los clientes como los programadores aprecian trabajar iterativamente haciendo pequeños incrementos. El cliente valora las entregas tempranas de valor para el negocio, y los programadores valoran la experiencia de aprendizaje. Hay una reducción de stress y aumento de confianza cuando la funcionalidad solicitada es entregada incrementalmente, en el orden definido por el cliente y en colaboración con los programadores.

Respecto a las “Stand-up” meetings el autor afirma que gradualmente han

reemplazado las tradicionales reuniones de reporte, las cuales se consideró que consumían mucho tiempo y había muchas personas quienes principalmente estaban fuera del alcance del proyecto. También se removió la sensación del “vigilante” del líder de proyecto, ya que este obtiene toda la información que necesita cada mañana y no necesita hacer sus rondas en orden de obtener esa información.

El autor afirma que también hubo algunos inconvenientes al usar XP.

- XP no nos ayudó respecto a la introducción de XP en una organización, ni como optimizar el proceso ni cómo mejorar la forma de trabajar, tampoco cómo lidiar con el personal de gestión
- No pudimos convencer a todos nuestros clientes de proveernos con requerimientos ambiguos en la forma de tests de aceptación. Lo mejor que pudimos obtener de ellos fueron escenarios para ejecutar
- Una longitud de iteración de dos semanas probó ser muy corta para nuestro pequeño equipo (1-5 en tamaño), en orden de agregar funcionalidad más significativa establecimos ahora una longitud de un mes
- Sentimos la necesidad de no sólo ingresar requerimientos funcionales en la lista de historias de usuario si no también los no funcionales.

Todos estos aspectos se solucionaron al implementar Scrum junto con XP, menciona el autor. Incluso ya había una fusión de las dos metodologías llamada XP@Scrum.

Respecto a los modelos de calidad el autor menciona:

En paralelo a la introducción de XP y Scrum en nuestra organización, decidimos tomar camino hacia la certificación ISO 9001 y CMM Level2. Por favor nótese que la certificación no era y nunca debe ser una meta por sí sola. La usamos en orden de medir el progreso de nuestras actividades de calidad usando las llamadas auto evaluaciones, nombradas como Interim Maturity Evaluations (IME) en CMM, y para ser capaces de medir y comprar nuestro proceso de desarrollo de software con otras organizaciones.

Respecto a ISO 9001 el autor afirma:

ISO 9001 tiene un nivel de abstracción superior y un mayor alcance comparado con CMM, el cual se enfoca estrictamente en el software. Esto significó que tuvimos que discutir cómo interpretar ciertos requerimientos de ISO para nuestra

organización. Para asistirnos en esta discusión y hacer claro qué grado de sofisticación es requerido para satisfacer un auditor CMM y de ISO. Contratamos un consultor para cada uno de los sistemas de gestión de calidad. El criterio de selección más importante para estos consultores fue su disposición a interpretar los requerimientos de CMM e ISO desde un punto de vista ÁGIL.

Respecto a la unión de las metodologías ágiles y los sistemas de gestión de calidad, el autor menciona un solo inconveniente:

Sólo para la gestión de la configuración y los procedimientos de aseguramiento de la calidad, no hubo soporte directo y adecuado (En términos de CMM e ISO) en XP y SCRUM.

En cuanto a las métricas el autor comparte las que la organización comenzó a medir.

- Cada 6 meses la satisfacción tanto de los clientes como de los programadores. En general se obtuvo entre 7 y 8 en una escala de 0-10
- La velocidad de varios equipos de desarrollo por cada iteración
- El número de líneas tanto de los tests como del código funcional agregado en cada iteración
- Los resultados de los tests
- Cada tres meses, el puntaje de una IME guiada
- El número de mayores y menores no conformidades por proyecto cada mes de iteración

Respecto a los desafíos enfrentados, se puede destacar según el autor.

Aseguramiento de la calidad: Tanto ISO 9001 como CMM requieren la existencia de un preferiblemente independiente grupo de aseguramiento de la calidad para proveer la gestión con visibilidad apropiada dentro del proceso que está siendo usado.

Sólo contrata a los mejores: Buenos ingenieros deben ser la mayor prioridad. Ningún proceso riguroso ni ágil, puede substituir el talento y las habilidades. Como un efecto secundario de nuestro innovador proceso basado en XP@Scrum, la de desarrollar software ha provado ser altamente interesante para ingenieros calificados de nuestra organización.

Dar espacio a la personalización: Diferentes proyectos tienen diferentes

necesidades de proceso por su criticidad, dominio del problema, tecnología usada, tamaño y distribución geográfica del equipo.

El encargado de la calidad debe ser un coach en lugar de un vigilante: La mejor forma de operar de un encargado de la calidad en un ambiente ágil, es trabajar sobre una base regular con el equipo de proyecto instruyéndolos a una forma definida de trabajar. No conformidades deben ser consideradas como una oportunidad de mejora. Actuar como un vigilante sólo diligenciando listas de chequeo de calidad es contraproducente.

Uso adecuado de los periodos de tiempo: Ingenieros y clientes deben aprender que el periodo de tiempo no es para presionarlos a trabajar contra el tiempo al final de cada iteración, si no para forzar a los clientes a tomar decisiones que retroalimenten el trabajo en orden de mantener el progreso.

Talleres de reflexión: Estos talleres, como son definidos por Scrum, al final de cada iteración son una forma ágil de incrementar la madurez y efectividad de los equipos de desarrollo.

Cliente en sitio: Experimentamos que era de lejos más simple acordar un espacio adecuado en la proximidad del cliente que convencer al cliente de sentarse con, o visitar a menudo el equipo de desarrollo. Esto resultó en equipos de desarrollo dispersados sobre nuestro campus, como consecuencia, menos transparencia en el conocimiento técnico.

XP no es para todos: XP y Scrum son metodologías altamente disciplinadas que se enfocan en lo verbal sobre la comunicación escrita. No todos los ingenieros o clientes aprecian esta forma de trabajo y algunos simplemente no se adherirán a ella. Hay que respetar esto y como consecuencia algunos ingenieros podrían dejar la organización y algunos clientes ser “mejor servidos” por un proveedor diferente.

Finalmente el autor concluye:

En Mayo de 2002 nos certificamos como proceso de desarrollo de software basado en XP@Scrum por ISO 9001:2000 y una rápida evaluación en diciembre de 2002 reveló que estamos en buen camino para la certificación CMM Level 2 en 2003. Pero como se expresó en los desafíos, Aún existe un número grande de inconvenientes a superar el en trayecto.

--

Título	Combining Extreme Programming with ISO 9000
Publicación	EurAsia-ICT 2002: Information and Communication Technology Volume 2510 of the series Lecture Notes in Computer Science pp 786-794
Año	2002
Autor(es)	Jerzy R. Nawrocki, Michal Jasiński, Bartosz Walter, and Adam Wojciechowski
Resumen	<p>Los principales motores del creciente mercado de las tecnologías de información y telecomunicación (ICT) son los productos software. El Observatorio de tecnología de información europeo estima que en el año 2002 el valor total de los productos software de la ICT en Europa del Oeste serán más de 70 billones de Euros. Desafortunadamente muy pocas personas están satisfechas con la calidad del software en cuanto a los productos y los procesos. Las herramientas de mejora de procesos como CMM e ISO 9000 estaban para curar esta situación, pero algunas personas se quejan de que son demasiado burocráticas e inflexibles. Como un nuevo resultado, las también llamadas metodologías ágiles aparecieron. Una de ellas es Extreme Programming (XP), un ligero, orientado al cambio y al cliente enfoque para el desarrollo de software. Aunque XP propone muchas prácticas interesantes, tiene algunas limitaciones. Más que todo no es claro como introducir XP a una organización certificada en ISO 9001:2000. El objetivo de este estudio es presentar una versión modificada de XP que sería aceptable desde el punto de vista de ISO 9000.</p>

Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001:2000 Extreme Programming
Métodos de armonización utilizados	XPMM
Nivel de detalle de la propuesta	Es un alto nivel de detalle que corresponde a un bajo nivel de abstracción. El autor menciona aspectos desde gestión de los procesos hasta el código del software, además provee un marco teórico antes del cuerpo de la investigación
Tipo de evaluación	Estudio de caso
Aspectos a destacar	
<p>El autor menciona al inicio del estudio que algunas compañías han decidido ir a través de la certificación ISO 9001. El peligro es que tal certificación podría resultar en un sistema bien documentado pero aún ineficiente. El entusiasmo inicial de los trabajadores por la mejora del proceso software pronto se disipará si ellos encuentran que ISO 9001 es sólo una fachada de marketing y que la compañía no tiene intención de introducir una mejora de procesos real.</p> <p>El autor afirma: “Una típica reacción de un programador a XP es: ‘Finalmente una metodología para las personas, no personas para la metodología’. Nuestra idea es usar este entusiasmo como un punto de partida para una mejora de procesos real y combinarla con los requerimientos de ISO 9000. Desafortunadamente, esa unión no es un camino sencillo. Lo que proponemos es una metodología modificada, basada en las prácticas de XP y conforme a ISO 9001:2000. Nuestro enfoque será en dos partes de ISO 9001:2000: La realización del producto y el monitoreo y medición.”</p> <p>El autor propone en orden de ganar flexibilidad un esquema similar a CMM, llamado XPMM. Sin embargo también menciona que XP aún tiene algunos puntos débiles, los problemas más importantes son con el mantenimiento del software. Ya</p>	

que los únicos artefactos son los casos de prueba y el código, después de algún tiempo puede ser muy difícil mantener el software. Lo que sería un problema también para la perspectiva de ISO 9001.

Respecto a la realización del producto de ISO 9001:2000.

Las áreas de proceso claves de XPMM asociadas con los niveles de madurez incluyen: Gestión de la relación con el cliente y aseguramiento de la calidad (Nivel 2), Pair Programming (Nivel 3), y Desempeño del Proyecto (Nivel 4).

Los niveles de madurez de XPMM son útiles ya que permiten introducir XP a una compañía gradualmente (nivel por nivel). Eso provee flexibilidad al proceso de mejora de software.

Gestión de la relación con el cliente (Importante para XP y para ISO 9001)

- Las historias de usuario son usadas para describir los requerimientos
- Un proceso de desarrollo es dividido en pequeñas entregas
- Planning Game es usado para crear el plan de entrega
- Una metáfora es seleccionada para facilitar la comunicación con el cliente
- No hay funcionalidades extras agregadas al inicio del proyecto

El radica en cómo introducir requerimientos documentados a una metodología ligera, para resolverlo uno tiene que notar que XP fue creado para ser ligero para los programadores y pone trabajo extra en los hombros de otras personas, por ejemplo el representante del cliente. Seguiremos esa ruta. Uno de los roles en el equipo de XP es el Tester, quien implementa los casos de prueba propuestos por el cliente. Nuestra sugerencia es hacer al Tester responsable por la documentación y gestión de los requerimientos.

Aseguramiento de la calidad del producto.

El aseguramiento de la calidad es dirigida por ISO 9000 en la cláusula 7.3.5

Diseño y Desarrollo.

- Primero las pruebas luego el código
- Todo el código debe tener pruebas unitarias
- Cuando un bug es encontrado, un caso de prueba debe ser creado
- Integración continua
- Optimización se hace hasta el último momento

Pair Programming.

Para implementar Pair Programming en una manera eficiente, un laboratorio de trabajo abierto es requerido. Lo cual está conectado con la cláusula 6.3

Infraestructura y 6.4 Entorno de trabajo

- El código debe ser escrito de acuerdo a estándares acordados
- Todo el código de producción es programado en pares
- Sólo un par integra el código a la vez
- Propiedad del código colectiva
- Usar sistema de gestión de versiones

Monitoreo y Medición.

En ISO 9001:2000 el monitoreo y medición son descritos en una términos muy generales. XP da algunas pistas sobre cómo implementar las cláusulas de ISO 9001 relacionadas con este asunto, 8.2.3 Monitoreo y medición de procesos. En el contexto de XP uno debería recolectar las siguientes métricas:

- Sobretiempo
- Disponibilidad del cliente
- Velocidad del proyecto
- Registro de integración para ver qué tan frecuentemente nuevas piezas de código son integradas con el sistema
- Modo de producción de cada pieza de código (Par o individual)
- Velocidad de programación (líneas de código por hora, casos de prueba por hora, tests de aceptación por hora)

Respecto a 8.2.4 Monitoreo y medición del producto: La principal medida son los reportes de los casos de pruebas y los tests de aceptación.

Respecto a 8.3 Control del producto no conforme: En XP antes de que una nueva versión es del sistema es aceptada para ser parte de la línea base, todos las pruebas de unidad deben ser pasados

Respecto a 8.5.2 Acción Correctiva: XP requiere crear casos de prueba para cada error detectado, Si el error persiste los casos de prueba lo descubrirán antes de que una nueva versión del sistema sea liberada.

Las cláusulas que quedan respecto a Monitoreo y medición de la satisfacción del cliente, auditorías internas, análisis de datos, mejora continua, y acciones preventivas son transparentes respecto a XP.

Título	Getting ISO 9001 Certified for Software Development Using Scrum and Open Source: A Case Study
Publicación	Tehnicki Vjesnik 22(6):1633-1640 · December 2015
Año	2015
Autor(es)	Tomo Popovic
Resumen	Este artículo presenta un caso de estudio de la adopción del proceso Scrum para el desarrollo Java usando herramientas el mundo del software de código abierto. El enfoque se concentra en pequeñas y medianas empresas de software, las cuales pueden algunas veces ser intimidadas por la introducción formal de procesos y sistemas de gestión de la calidad. Este estudio describe las prácticas ágiles, el proceso de Scrum, y selección de herramientas con el objetivo de desarrollar usando Java. Cada herramienta seleccionada es descrita y puesta en contexto del proceso Scrum y la implementación de un sistema de gestión de calidad. El estudio introduce es estándar ISO 9001:2008 y discute los beneficios de implementación de un sistema de gestión de calidad. Finalmente, el caso de estudio ilustra cómo las herramientas seleccionadas y los artefactos de Scrum pueden ser usados para la adopción de un sistema de gestión de la calidad, el cual conduce a una certificación exitosa en ISO 9001:2008
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001:2008 SCRUM
Métodos de armonización utilizados	Mapping con tablas comparativas
Nivel de detalle de	La propuesta tiene un nivel de detalle muy alto, con

la propuesta	ilustraciones explícitas desde las áreas de proceso para el sistema de gestión de calidad, las tablas de mapping entre los documentos requeridos por ISO 9001 y los que SCRUM provee, el diagrama de despliegue de los componentes necesarios para configurar el entorno físico de desarrollo incluyendo servidores, clientes, herramientas y todo muy bien explicado también textualmente
Tipo de evaluación	Estudio de caso en una pequeña empresa
Aspectos a destacar	
<p>Es importante destacar que el autor es un impulsor de la cultura del software de código abierto, al inicio del estudio afirma.</p> <p>El desarrollo de software de código abierto promueve el pensamiento evolutivo, y los procesos iterativos e incrementales los cuales están muy en línea con los métodos ágiles para el desarrollo de software.</p> <p>El autor cita uno de los estudios de sus referencias en el que combinan Scrum y XP en orden de alinearse para ISO 9001. Y comenta que eso realmente no es necesario, pero que los autores hicieron bien al notar que ISO 9001 no es igual a calidad, pero asegura que las prácticas ágiles son bien llevadas a cabo.</p> <p>Respecto al marco teórico de SCRUM, el autor dice:</p> <p>En las reuniones de revisiones de sprint el equipo demuestra que las características nuevas implementadas, errores corregidos u otros artefactos relacionados al producto. Las presentaciones, diapositivas y reportes de progreso no son consideradas de valor. El énfasis es el demo funcionando de las nuevas características y el valor para el usuario.</p> <p>Respecto a los malentendidos respecto a ISO 9001 y Scrum el autor afirma:</p> <p>Hay un común malentendido que la implementación ISO 9001 de un sistema de gestión de calidad tiene que tener documentación intensiva y no puede ser conseguido en ambientes de pequeñas y medianas empresas.</p> <p>Otro malentendido es que la documentación no es parte de los procesos ágiles, lo</p>	

cual es erróneo. Aunque el software funcionando es la prioridad, la correcta y necesaria documentación es requerida.

El autor menciona respecto a ISO 9001:2008:

Para pequeñas empresas todo esto puede parecer intimidante, pero el SGC de ISO 9001:2008 es de hecho más fácil de implementar que sus versiones anteriores. De cualquier forma es aconsejable obtener los materiales de entrenamiento, plantillas de documentos y si es necesaria la asesoría de firmas de consultoría. En nuestro caso usamos plantillas para crear políticas, manual de calidad y documentos de procedimiento listados a continuación

- Manual de calidad, definición de política de calidad y objetivos de calidad
- Control de documentos
- Control de registros
- Auditorías internas
- Control de producto no conforme
- Acción correctiva
- Acción preventiva

Las siguientes secciones introducen algunas peculiaridades que necesitaron consideraciones y esfuerzos adicionales para alinear el desarrollo de software con Scrum y los requerimientos de ISO 9001.

Diagrama organizacional.

Primero fue necesario identificar roles pertinentes al SGC y definir el diagrama organizacional. Pequeñas y medianas empresas tienen a verse agobiadas con esta tarea y a menudo optan por diagramas complicados. Suele suceder que en tales diagramas una persona cumple muchos roles, lo cual es innecesario. La clave aquí es mantenerlo simple, especialmente para aquellas empresas con pocos empleados.

Crear un diagrama de procesos.

Otro aspecto aquí es definir el diagrama de procesos. Para las empresas desarrolladoras de software pueden ser definidas tanto como proveedoras de servicio y de productos. Después de algunas consideraciones, fue decidido que para nuestro caso en particular la opción de proveedora de servicios es la adecuada. En el diagrama de procesos definimos a) Gestión de Calidad, b) Gestión contractual, c) Entrega de servicio, y d) Compras. Es importante notar que todos estos procesos y sus actividades correspondientes tienen que ser trazables a los requerimientos de ISO 9001:2008 especificados en las secciones 4-8 del estándar.

Identificando exclusiones.

Dependiendo de la situación, para los requerimientos de ISO 9001 que no son aplicables, es posible solicitar y documentar las exclusiones. Es importante leer cuidadosamente los requerimientos, identificar y justificar las posibles exclusiones.

[El autor muestra una tabla con las exclusiones para su empresa en particular]

Mapear artefactos de Scrum con los requerimientos de ISO 9001.

Nos preguntamos a nosotros mismos si había una posibilidad de mapear los artefactos de Scrum y los requerimientos de ISO 9001:2008. La respuesta corta a esta pregunta es SI, pero ciertas cosas deben ser consideradas. Adicional a los artefactos de Scrum es importante documentar bien.

Definición de 'Hecho'. (Para las historias de usuario terminadas y cerradas)

Definición de 'Listo'. (Para las historias de usuario bien documentadas y aceptadas listas para implementar)

[El autor presenta una tabla con el mapeo de los artefactos de Scrum y los requerimientos de ISO 9001]

Todo el equipo debe acordar colaborativamente que los requerimientos están claros.

Un ítem es testeable si hay una forma efectiva de determinar si la funcionalidad trabaja como se espera. Típicamente esto es provisto por los tests de aceptación.

Una consideración adicional es la selección de las herramientas software que soportan la gestión de los procesos y artefactos de Scrum. Un ejemplo interesante es el uso de un software de seguimiento con el objetivo de trazar errores e impedimentos. En adición a esta función primaria, puede ser muy exitoso gestionar y documentar las acciones preventivas y correctivas como es requerido por ISO 9001, estos pueden ser documentados usando formularios en papel pero también se pueden gestionar mucho más eficientemente si se usa una herramienta de seguimiento de impedimentos como 'Bugzilla'.

Las herramientas ilustradas y descritas condujeron a una pequeña empresa a una exitosa certificación ISO 9001:2008

La trazabilidad de errores e impedimentos en combinación con las herramientas de gestión del product Backlog pueden ver los errores como nuevos requerimientos descubiertos tales como trabajo relacionado con corregir errores, que pueden ser priorizados, estimados, y planeados por cada iteración.

Título	How can Agile and Documentation-Driven Methods be Meshed in Practice?
Publicación	Agile Processes in Software Engineering and Extreme Programming Volume 179 of the series Lecture Notes in Business Information Processing pp 62-77
Año	2014
Autor(es)	Lise Tordrup Heeager
Resumen	<p>Los métodos ágiles se están volviendo incrementalmente populares en el desarrollo de software; incluso para organizaciones que cumplen con estándares de calidad. La literatura reporta ejemplos dispersos de organizaciones que han tenido éxito en la unión de métodos ágiles y métodos dirigidos por documentación. De cualquier forma, debido a una falta de investigación empírica, no es bien entendido cómo implementar una práctica de desarrollo de software mixta. Para incrementar el entendimiento de cómo hacer esto, este estudio presenta dos casos de estudio del desarrollo software críticamente seguro. El primer caso de estudio presenta los desafíos de adoptar aseguramiento de la calidad en una práctica ágil de software. El segundo caso de estudio muestra cómo las prácticas ágiles son adoptadas en una práctica dirigida por documentación conforme con el estándar de US Food and Drug Administration. Basado en un framework que identifica 9 áreas de práctica en las cuales los métodos difieren, los desafíos y posibilidades en la implementación de una práctica de desarrollo mixta es presentado</p>
Descripción	

Modelo de calidad y/o metodología ágil utilizadas	GAMP FDA SCRUM
Métodos de armonización utilizados	El método es “Meshing” definido por el autor haciendo referencia a la definición del diccionario “Enredar o entrelazar”
Nivel de detalle de la propuesta	Propuesta de alto nivel, el autor presenta 9 áreas en las cuáles los métodos ágiles y los dirigidos por documentación pueden gestionan el mismo concepto de maneras diferentes
Tipo de evaluación	2 estudios de caso, uno es una pequeña organización que implementa métodos ágiles y busca hacer un “Mesh” debido a la demanda de una certificación de calidad. El otro caso es una gran organización que implementa un plan dirigido por documentación que busca implementar Scrum y hacer un “Mesh” con su ya implantado plan.
Aspectos a destacar	
<p>El autor no menciona explícitamente la metodología ágil que usa la pequeña empresa, sin embargo hace referencia a programación en pares lo cual llevaría a pensar que usan prácticas de XP, esta pequeña empresa busca implementar el estándar de calidad GAMP.</p> <p>En el segundo caso la gran organización utiliza estándares que cumplen con la FDA y busca incorporar SCRUM.</p> <p>Las 9 áreas de práctica que el autor presenta, en las cuales tanto los métodos ágiles como los dirigidos por documentación trabajan en formas diferentes son:</p> <ul style="list-style-type: none"> ● Estrategia de la dirección ● Relación con los clientes ● Relaciones entre las personas ● Documentación ● Requerimientos ● Estrategia de desarrollo ● Comunicación y transferencia de conocimiento ● Pruebas 	

- Cultura

Respecto al estado de las organizaciones estudiadas.

Caso1, SmallSoft (Pequeña empresa)

El primer caso se concentra en el proceso y desafío de adoptar aseguramiento de la calidad en una pequeña organización en Dinamarca. Fueron forzados a adoptar aseguramiento de la calidad por su cliente principal. El proyecto incluye la mayoría de 15 desarrolladores empleados en la organización.

El proceso de desarrollo se inició en 1998, en octubre de 2004 la organización decidió adoptar aseguramiento de la calidad, en 2007 tuvieron su primera auditoría externa y fallaron.

Caso2, LargeSoft (La gran organización)

El segundo caso describe como un producto clínico con crítica seguridad embebida es desarrollado en una compañía farmacéutica, se concentra en un grupo de 30 personas incluyendo managers, desarrolladores y testers. Debido a la necesidad de medidas críticas de seguridad el proyecto debía cumplir con el estándar de la FDA, después de varios años la organización quiso refinar el producto incorporando Scrum.

Después de haber estudiado los dos casos:

El “Meshing” promedio en los dos casos.

El objetivo declarado en ambos casos era conseguir un “Mesh”. En la práctica no todos los desarrolladores y managers estaban dispuestos a conseguirlo, lo cual es identificado en el análisis como un gran obstáculo.

Meshing en el caso1.

A pesar de los esfuerzos para conseguir el “Meshing” la organización permaneció primordialmente ágil. La auditoría externa mostró que la excesiva documentación requerida por el estándar de calidad fue la principal razón por la que el “Meshing” no fue satisfactorio. La cantidad y calidad de la documentación producida en los requerimientos y para compartir el conocimiento no fue suficiente.

Meshing en el caso2.

Durante el tiempo del estudio la organización incrementó la agilidad de las prácticas del software consiguiendo el “Meshing”, sin embargo permaneció

primordialmente dirigido por documentación.

El equipo tuvo éxito al implementar un desarrollo de software ágil embebido en el proyecto dirigido por documentación. El análisis mostró que los desarrolladores no fueron capaces de establecer contacto con el cliente y un buen desempeño del rol de Product Owner. Debido a la excesiva documentación de la FDA, la forma de manejar la documentación tuvo dificultades, también la forma de manejar los requerimientos tuvo dificultades por la naturaleza secuencial del proceso que tenían. La estrategia de comunicación pudo ser implementada más personalmente aunque soportada por la documentación que ya existía.

Respecto a las 9 áreas, en ambos casos se pudo analizar 6.

Relación con los clientes: En ambos casos no fue posible hacer un Mesh, y permanecieron con el enfoque que tenían. En el primer caso, la confianza que tenía el cliente con la empresa impidió implementar documentación de cantidad y calidad (El cliente encontró difícil el papeleo).

Documentación: No fue posible hacer un Mesh en ninguno de los dos casos, ambos permanecieron con la naturaleza que ya tenían.

Requerimientos: El primer caso no logró hacer un Mesh y permaneció ágil. El segundo caso pudo incorporar prácticas ágiles, sin embargo fue principalmente re ingeniería en etapas tardías del proyecto.

Estrategias de desarrollo: El caso 1 no intentó cambiar su estrategia iterativa incremental, en el caso dos fue posible hacer un Mesh, sin embargo prevaleció primordialmente dirigido por documentación porque aunque la estrategia iterativa fue ventajosa, hubo problemas en llegar a los metas establecidas por el plan inicial.

Comunicación y transferencia del conocimiento: En el primer caso no se logró un Mesh y las relaciones permanecieron ágiles sin mucha documentación. En el segundo caso se logró el Mesh, aunque permanecieron con gran documentación encontraron ventajas en la comunicación cara a cara y cercana.

Testing: En el primer caso permaneció el test-driven development, En el segundo caso no fue posible conseguir un Mesh, y las pruebas se realizaron en etapas tardías del proyecto.

Título	Investigation of the Capability of XP to Support the Requirements of ISO 9001 Software Process Certification
---------------	--

Publicación	Software Engineering Research, Management and Applications, ACIS International Conference on (2010)
Año	2010
Autor(es)	Malik Qasaimah, Alain Abran
Resumen	Para organizaciones de software que necesitan la certificación ISO 9001, es importante establecer un ciclo de vida del proceso software que pueda gestionar los requerimientos impuestos por el estándar de la certificación. Este estudio presenta un análisis de extreme programming (XP) desde las perspectivas de ISO 9001 e ISO 90003. El enfoque está en extraer los requerimientos relacionados con el proceso de realización del producto de ISO y determinar las fortalezas y debilidades de XP en la gestión de esos requerimientos.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001:2008, ISO 90003, ISO 12207 XP
Métodos de armonización utilizados	“Extracción” y Mapping, el autor utiliza una interface desde ISO 9001 e ISO 90003 hacia XP. Extrayendo las actividades de ISO 9001/90003 que corresponden a las fases definidas por ISO 12207 para el ciclo de vida del software, y posteriormente haciendo un mapping a XP.
Nivel de detalle de la propuesta	Alto nivel de detalle sin tocar hilos muy profundos, el autor se enfoca meramente en lo que al software respecta de ISO 9001, la sección “Diseño y desarrollo”
Tipo de evaluación	Análisis de las normas y la metodología.
Aspectos a destacar	
Respecto a la planeación de la realización del producto. ISO 90003 especifica directrices para ser aplicadas a la planeación del desarrollo	

de software clasificadas en 3 categorías.

- Actividades del desarrollo
 - Identificación de las entradas requeridas y salidas para para cada fase
 - Identificación de las prácticas y procedimientos usados para verificar que las fases del desarrollo de software son bien llevadas a cabo y el producto ha sido testeado en varias fases
- Actividades de la gestión
 - Identificación del cronograma de entregables, tiempo, recursos y presupuesto para cada entregable
 - Identificación de las interfaces técnicas y organizacionales
 - Identificación de cualquier ley reguladora que pueda afectar el proceso o el producto
- Métodos y herramientas
 - Identificación de herramientas y técnicas para el desarrollo

Finalmente el autor presenta el mapping para esta sección.

ISO 9001: Requerimientos en la fase de planeación

ISO 12207: Implementación del proceso (Sección 5.3.1)

XP: Planning game, iterative planning

Respecto a la recolección de los requerimientos.

ISO 90003 muestra actividades específicas para ser aplicadas en esta etapa:

- Método para trazar los cambios a los requerimientos durante el desarrollo iterativo, y un método para registrar los cambios
- Matrices de trazabilidad para los requerimientos hasta el producto final
- Detalles sobre cualquier otra interfaz, herramienta o algoritmo necesitado durante el desarrollo.

Respecto a los requerimientos de ISO durante las actividades de validación.

ISO 90003:

- Los desarrolladores deberían usar un método adaptable para la evaluación de los requerimientos.
- Los requerimientos deberían ser evaluados en la presencia de los clientes y en una reunión cerrada. Todos los requerimientos ambiguos e inconsistentes deben ser registrados.

El Mapping para esta sección se presenta así:

ISO 9001: Requerimientos ISO en la fase de recolección de los requerimientos,

Requerimientos ISO en la fase de validación de los requerimientos.

ISO 12207: Análisis de requerimientos de software (secciones 5.3.4.1 y 5.3.4.2)
XP: Historias de usuario, Tests de aceptación del cliente, Prototipado, Cliente en sitio.

Respecto a la fase de construcción.

ISO 9001 y 90003 incluyen varios requerimientos en esta etapa. Se puede resumir así:

- Asegurar que los métodos destinados a ser usados, durante la fase de planeación fueron de hecho usados durante la fase del desarrollo
- Asegurar que los requerimientos del software son cumplidos por el producto software

De acuerdo a ISO 90003, la revisión del diseño y desarrollo debería ser desempeñada de acuerdo al plan definido. El plan debería identificar los elementos importantes tales como:

- Que necesita ser revisado y las responsabilidades en el equipo
- El propósito del proceso de revisión y las herramientas y técnicas usadas
- Identificación de los pasos necesarios para resolver cualquier anomalía encontrada durante la revisión

El Mapping para esta etapa:

ISO 9001: Requerimientos ISO en la revisión del diseño y desarrollo.

ISO 12207: Revisión conjunta del proceso (Sección 6.6)

XP: Pair Programming.

Respecto a la verificación y validación del diseño y desarrollo.

ISO 90003 especifica directrices en esta etapa clasificadas así:

- Los desarrolladores deberían seleccionar los métodos de verificación adecuados dependiendo del tamaño complejidad del proyecto software. (Prototipado, simulaciones, testing)
- La versión final del producto software debería ser validada antes de ejecutar los tests de aceptación del cliente
- La validación puede desempeñarse al hacer pruebas de software a diferentes niveles (Pruebas unitarias, pruebas de integración, tests de calificación, tests de aceptación y tests de regresión)
- Cualquier problema o anomalía encontrada debe ser registrada y las acciones necesarias deben ser tomadas.

El Mapping para esta etapa:

ISO 9001: Requerimientos ISO en la verificación y validación del diseño y desarrollo

ISO 12207: Proceso de verificación (Sección 6.4), Proceso de validación (Sección 6.5)

XP: Pruebas unitarias, Pruebas de integración, Tests de aceptación.

Al final del estudio el autor concluye:

- La principal técnica para documentar los requerimientos en XP es la historia de usuario, de cualquier manera estas proveen pocos detalles respecto a lo que se especifica por ISO 9001 e ISO 90003.
- Las historias de usuario son principalmente escritas en un lenguaje natural y las especificaciones formales no son provistas, así los requerimientos son evaluados por prototipos, clientes en sitio. Las evaluaciones formales como validación del modelo no son soportadas por extreme programming.
- ISO 90003 requiere que los desarrolladores seleccionen métodos adecuados de revisión durante el diseño y desarrollo. Las revisiones de XP son principalmente basadas en Pair Programming. Las actividades de revisión en XP carecen de métodos formales y falla al proveer los documentos requeridos de ISO 9001
- XP no registra los pasos al resolver anomalías encontradas durante las pruebas unitarias, tests de aceptación y de integración, no proveen suficiente evidencia a los auditores de ISO 9001 sobre cómo estas actividades han sido planeadas, puestas en el cronograma y llevadas a cabo. Como resultado fallan al satisfacer los requerimientos de ISO 9001 en esta etapa.

Título	ISO 9001 and Agile Development
Publicación	AGILE '07 Proceedings of the AGILE 2007 Pages 262-265
Año	2007
Autor(es)	Bill McMichael, Marc Lombardi

Resumen	El pensamiento convencional concluiría que ISO y la agilidad no deben ser compatibles. Después de todo, ISO es a menudo caracterizado por ser pesado en el proceso y pesado en la documentación – lo opuesto a la agilidad. Sólo con la asunción que la agilidad trata sobre no documentación es erróneo. Entonces también lo son las asunciones que ISO necesita ser un proceso muy pesado. ISO 9001:2000 no sólo es compatible con la agilidad, sino que también puede proveer suficiente estructura para ayudar a asegurar que la agilidad de los procesos es bien llevada a cabo.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001:2000 SCRUM/XP
Métodos de armonización utilizados	El autor no menciona explícitamente ningún método de armonización, sólo cuenta la experiencia de su organización
Nivel de detalle de la propuesta	Muy alto nivel, el autor no menciona explícitamente ningún detalle de bajo nivel como por ejemplo temas respecto al código, ni procesos.
Tipo de evaluación	Estudio de caso “Primavera Systems, Inc.”
Aspectos a destacar	
<p>El objetivo del estudio según el autor, es describir como Primavera Systems, una empresa desarrolladora de proyectos de software para la gestión empresarial. Estableció un sistema de gestión de calidad que fue alineado con ISO 9001 mientras aún mantuvo todos los beneficio de sus prácticas ágiles SCRUM/XP.</p> <p>El autor menciona que la organización había seguido prácticas ágiles de SCRUM y XP y decidieron ir a través de la experiencia de como proveer apropiados niveles de documentación sobre los procesos existentes para satisfacer solicitudes de clientes y proveer referencias útiles a los desarrolladores.</p>	

El autor, debido a que la organización no buscaba la certificación por motivos no justificables de negocio, introduce el concepto de “Alineado” diciendo que el sistema de gestión de calidad sería “alineado” a ISO 9001:2000.

Para la implementación del sistema de gestión de calidad, la organización contrató a una consultora “RCG IT” para conducir y evaluar el proceso con los objetivos:

- Proveer una opinión externa del actual estado del desarrollo de software, las políticas de tecnologías de información, procedimientos e instrucciones de trabajo y como se relacionan en alineación con ISO 9001:2000
- Crear un espacio para reportar el análisis que identifica los vacíos entre el actual estado de la organización y el estándar ISO 9001:2000

La primera evaluación concluyó que los procesos de desarrollo de la organización, aunque no se ha documentado al pie de la letra de ISO 9001:2000, fuertemente refleja su intención tanto en espíritu como en ejecución. Debido a que el desarrollo de software esta estructurado alrededor de equipos altamente evolucionados que se enfocan en los requerimientos dirigidos por el cliente para implementar funcionalidad completa, probada y funcional en cortos periodos de tiempo. De cualquier forma carecía de la documentación formal necesaria para estar alineada con ISO.

La Organización usaba prácticas ágiles de SCRUM y XP y tenían un deseo de establecer un marco de trabajo en el cual direccionar la mejora de sus procesos. Además de que documentación adicional sería útil para ayudar a los nuevos miembros del equipo.

Debido a que no había razón de negocio para buscar la certificación. El sistema de gestión de calidad de la organización no fue creado al pie de la letra del estándar. Más que eso la alineación con ISO 9001:2000 significa que hay políticas y procedimientos documentados en un formato consistente con ISO.

El proceso que estableció el SGC buscaba simplemente documentar procesos que ya estaban institucionalizados como resultado del trabajo realizado por equipos auto dirigidos.

Se determinó que los procedimientos claves a documentar eran:

- Pruebas de software
- Gestión de la configuración
- Trazabilidad de defectos

- Internacionalización
- Mantenimiento del producto

Como adición también debía ser documentado los procedimientos que describirían los métodos ágiles de la organización.

Se trabajó sobre 30 talleres para documentar los procedimientos. Cada taller incluía una sección transversal del equipo de desarrollo y era facilitado por un consultor. El proceso fue ejecutado sobre un periodo de 10 meses, resultando en un impacto mínimo sobre el cronograma de desarrollo y dando suficiente tiempo para la revisión y validación de los documentos finales. La información fue recolectada usando las siguientes preguntas a los miembros del equipo.

¿Qué tareas están asociadas con esta función?

Y por cada tarea;

¿Qué tiene que pasar antes de comenzar?

¿Qué entradas necesitas y de donde las obtienes?

¿Quién es responsable por asegurar que la tarea está hecha?

¿Cómo sabes cómo llevar a cabo la tarea?

¿Qué salidas produces y dónde las pones?

¿Cómo sabes que has terminado?

Finalmente quien escribió los documentos fue el consultor externo lo que permitió a los miembros enfocarse en describir los procesos y revisar los documentos. El autor menciona que tener alguien externo para escribir y ser un facilitador fue un factor clave de éxito, también que todo el SGC estaba totalmente soportado por la dirección.

Conclusión:

El autor dice “ISO no es igual a calidad. Simplemente ayuda a asegurar que las prácticas ágiles se realizan correctamente. Había preocupaciones al violar los principios ágiles. Nuestro mantra fue proveer sólo la documentación necesaria para ser una referencia útil y ayudar a aplicar correctamente los procesos existentes.

Título	Reconciling the irreconcilable? A software development approach that combines Agile with Formal.
---------------	--

Publicación	Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06)
Año	2006
Autor(es)	José Goncalo A. Oliveira Basto da Silva, Paulo Rupino da Cunha
Resumen	<p>Requerimientos de software inestables son comunes y piden prácticas ágiles. En la otra mano, los estándares de calidad y recomendaciones, a las cuales las compañías se deben adherir para entrar en mercados seleccionados, aplica la formalidad en el proceso de desarrollo. Presentamos un enfoque, adaptado a una compañía real, la cual reconcilia estos extremos. Hemos usado como entradas el proceso original de desarrollo de software y sus restricciones de formalidad, las dificultades de la compañía y varios enfoques ágiles. La salida ha sido probada en varios casos con resultados motivantes respecto a la conciencia sobre el progreso del proyecto, gestión de los requerimientos, control del esfuerzo y tiempo, y calidad del producto. Aunque el enfoque fue personalizado para una compañía en particular. Puede dar una visión significativa y aguda para otros que enfrentan desafíos similares.</p>
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	<p>El autor no especifica explícitamente el modelo de calidad usado para el proyecto presentado, sin embargo menciona que la organización cumple con algunos estándares.</p> <p>ISO 9001 ISO 12207 SPICE ECSS (European Community for Space Standardization) NASA-SEL (NASA Software Engineering Laboratory Recommendations)</p> <p>Respecto a lo métodos ágiles el autor dice que se usa una</p>

	<p>adaptación para la organización de prácticas de</p> <p>SCRUM</p> <p>XP</p> <p>ASD (Adaptive Software Development)</p>
Métodos de armonización utilizados	Tailoring
Nivel de detalle de la propuesta	Es una propuesta muy detallada, el Autor es muy específico en el qué y cómo es la versión final de la adaptación de las prácticas ágiles conservando la formalidad necesaria para cumplir con los estándares, respecto al ciclo de vida del desarrollo de software.
Tipo de evaluación	Estudio de caso
Aspectos a destacar	
<p>La organización es una gran empresa desarrolladora de software internacional que trabajaba inicialmente desarrollando para el sector aero espacial, cumpliendo con todos los estándares, sin embargo posteriormente incursionó en el sector industrial, y debido al crecimiento de la empresa y las particularidades del sector industrial se presentaron los siguientes inconvenientes:</p> <ul style="list-style-type: none"> • Requerimientos cambiantes • Falta de compromiso del cliente • Fechas límite estrictas, alta demanda, poca tolerancia y bajo presupuesto • Problemas de comunicación entre desarrolladores y clientes <p>La organización decidió implementar prácticas ágiles debido a las circunstancias del sector industrial, sin embargo el autor enfatiza que ninguna de las metodologías ágiles podía ser implementada sin cambios. Era necesario reconciliar la agilidad con el grado de formalidad impuesto por los estándares y recomendaciones del sistema de gestión de calidad.</p> <p>El autor menciona que utilizaron una herramienta CASE “Enterprise Architect” para ayudar a gestionar los requerimientos más efectivamente a través de matrices</p>	

relacionales y documentación auto generada. También se utilizó un software para pruebas unitarias automáticas.

Para auditar las nuevas prácticas ágiles, la organización utilizó una técnica llamada “Action-Research” en varios proyectos durante 3 ciclos, mejorando el proceso iterativa e incrementalmente. El estudio se enfoca en cómo es el proceso resultante con prácticas ágiles y que conserva la rigurosidad y formalidad para cumplir con los estándares.

Se buscaron prácticas ágiles de SCRUM, XP y ASD.

Se eliminaron algunas características debido a las condiciones de la organización, como pair programming y daily stand-up meetings debido a que tenían grupos de desarrollo de 2 programadores. El autor también menciona que el Product Backlog de Scrum no pudo ser utilizado eficientemente sino hasta el tercer ciclo de la investigación. Y sólo arrojó resultados satisfactorios en 3 de 4 proyectos debido a la falta de disciplina en uno de ellos.

El enfoque ágil diseñado soporta casos de requerimientos inestables y está compuesto por 4 fases:

- Fase de requerimientos
- Fase de diseño, ingeniería y validación
- Fase de aceptación
- Fase de mantenimiento y operación.

Es importante que todo el equipo del proyecto sea identificado. Consiste no sólo del personal de la organización sino también de los interesados del cliente que se hagan corresponsable por el proyecto. Incluso cláusulas del contrato de penalidad para el cliente fueron incluidas.

Las funcionalidades deberían ser descritas simples no ambiguas usando la técnica SMART

- Specific
- Measurable
- Attainable
- Realizable
- Traceable

La herramienta CASE ayudó también a relacionar funcionalidades con componentes para analizar el impacto de solicitudes de cambio y tests de pruebas. Si el cliente demanda modificaciones a los requerimientos iniciales, las matrices ayudan a identificar el impacto y las implicaciones económicas de forma adecuada.

También se usó Cruise-Control una herramienta de software libre altamente configurable para integración continua.

El autor presenta una tabla comparativa de dos proyectos del sector industrial, uno con el viejo proceso formal y otro con el proceso ágil resultante conforme a los estándares, los resultados son extremadamente exitosos en cuanto a calidad, costos y tiempo.

Finalmente el autor concluye:

“Presentamos un proceso de desarrollo de software que es simultáneamente, suficiente ágil para soportar requerimientos inestables y suficiente formal para cumplir con las certificaciones de calidad. Ha sido confeccionado para la compañía cuyos clientes tradicionales demandan estrictas metodologías para manejar sus estables pero críticos requerimientos. Pero también aquellos nuevos clientes que demandan requerimientos cambiantes sin perder las fechas límite. El enfoque ha sido probado en varios casos reales con resultados motivantes. En aquellos casos en que las fechas límite fueron incumplidas, el cliente reconoció su responsabilidad por no haber respondido rápidamente al equipo de desarrollo. El enfoque propuesto cumple con la mayoría de desafíos presentados en la introducción, es decir la necesidad de un enfoque iterativo e incremental que permite enfrentar requerimientos cambiantes en una manera responsiva. La participación del cliente en la especificación de requerimientos, revisiones y talleres no sólo en los momentos clave del proyecto sino también entre sprints le da una sensación de propiedad del producto en proceso.

Título	Requirements of an ISO Compliant XP Tool
Publicación	Extreme Programming and Agile Processes in Software Engineering Volume 3092 of the series Lecture Notes in Computer

	Science pp 266-269
Año	2004
Autor(es)	Marco Melis, Walter Ambu, Sandro Pinna, Katuscia Mannaro
Resumen	En los últimos años, unos pocos estudios y experiencias han sido publicados sobre la compatibilidad entre Extreme Programming y la certificación ISO 9001:2000. El problema real no es demostrar si es posible certificar un proceso XP sino explicar cómo un proceso XP puede mantener su agilidad en tal contexto. Pensamos que el uso de una herramienta adecuada que soporte tanto las prácticas de XP y el estándar ISO 9001 puede simplificar este proceso de integración. En este estudio proveemos los requerimientos esenciales para una herramienta con tales funcionalidades.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001:2000 XP
Métodos de armonización utilizados	Ninguno
Nivel de detalle de la propuesta	Muy alto nivel
Tipo de evaluación	Ninguno
Aspectos a destacar	
<p>El artículo presenta algunos “Requerimientos” de muy alto nivel para una aplicación que soporte la gestión del proceso de desarrollo de software con XP e ISO 9001.</p> <p>El autor describe algunos requerimientos de alto nivel únicamente para la sección</p>	

de realización del producto en diseño y desarrollo de la norma ISO 9001.

- Planeación del diseño y desarrollo: La herramienta debe soportar la planeación de los procesos del desarrollo de software. Tal como la definición de entregables e iteraciones. Sobretodo debería soportar la definición de las responsabilidades y autoridades relacionadas. En adición, debe comunicar los resultados correspondientes a estas actividades de planeación dentro de la organización como también el control de las actividades planeadas.
- Entradas del diseño y desarrollo: La herramienta debe permitir registrar este tipo de actividades y mantener los registros relacionados. Debería proveer información y sugerencias recolectadas en proyectos previos similares.
- Salidas del diseño y desarrollo: Las principales salidas de un proceso de desarrollo XP son el código y sus tests relacionados. Por medio de la herramienta, debe ser posible verificar automáticamente que los elementos de las salidas son adecuadas para satisfacer los requerimientos de entrada: en práctica, la herramienta debería permitir la asociación de cada artefacto de salida con el correspondiente elemento de entrada y su test relacionado. En esta manera, gracias a una apropiada interface con una aplicación de testing, será posible verificar la satisfacción del requerimiento.
- Revisión del diseño y desarrollo: En un proceso XP, las reuniones de iteración son una revisión formal del diseño y desarrollo. El informe de esta actividad y las salidas resultantes son registros del sistema de gestión de calidad que deben ser integradas en la herramienta. Pair-Programming con refactorización y testing son otras formas de revisión del diseño y desarrollo. La herramienta debe identificar y registrar las salidas de estas actividades para documentar y dar evidencia de que la revisión continua es llevada a cabo en un ambiente XP.
- Verificación del diseño y desarrollo: En un proyecto XP, la verificación del diseño y desarrollo es hecha continuamente a causa de la integración continua y el testing, pero puede ser ejecutada formalmente en momentos específicos como el fin de cada iteración. Consiste de verificar la efectividad de las salidas para satisfacer los requerimientos de entrada. Puede ser principalmente hecha por medio de tests de aceptación automáticos. La herramienta debe permitir la ejecución de estos tests en el que cada uno esté relacionado a su respectiva historia de usuario. Luego debe registrar salidas e identificar cada historia de usuario correctamente implementada como también testeada y verificada.
- Validación del diseño y desarrollo: La validación del diseño y desarrollo es un conjunto de actividades hechas por la organización, a menudo con el

cliente, cuyo objetivo es evaluar que el producto es conforme a los requerimientos del cliente para el uso esperado. En proyectos XP, la validación está a cargo del cliente (o su delegado). Por esta razón la herramienta debe restringir el permiso de modificación del estado de validación de cada entidad del proceso (Historia de usuario, entregable,...) únicamente al cliente (o su delegado).

- Control de los cambios del diseño y desarrollo: Las metodologías ágiles aceptan el cambio continuo a los requerimientos. De cualquier forma, estos deben ser controlados y cada actividad/artefacto involucrado en los cambios a requerimientos debe ser identificado. Con este objetivo la herramienta puede tener un rol fundamental gracias a su característica de trazabilidad: puede correlacionar cada historia de usuario con su código y tests específicos. Estas funcionalidades pueden ser implementadas gracias a la integración con una aplicación de gestión de la configuración. De esta manera es inmediato encontrar todas aquellas entidades involucradas en cambios, identificar el nivel de cambio y luego monitorear y controlar todas estas actividades. Después de un cambio la herramienta debe re activar los procesos de revisión, verificación y validación para cada entidad involucrada. Sobre todo, debe proveer, por medio de la lista de distribución apropiada, distribuir la información relacionada a los cambios realizados.

Título	Standards and Agile Software Development
Publicación	SAICSIT '03 Proceedings of the 2003 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology Pages 178-188
Año	2003
Autor(es)	W.H. Morkel Theunissen, Derrick G. Kourie, Bruce W. Watson

Resumen	Este estudio investiga la adaptabilidad de las metodologías ágiles a los estándares del desarrollo de software establecidos por ISO. En la base de este análisis, directrices son propuestas para el desarrollo de software de tal forma que el proceso de desarrollo sea conforme a los estándares requeridos por quien adquiere el software, reteniendo aún sus características ágiles. Extreme Programming (XP) es usado como el primer representativo de varias metodologías ágiles.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 12207:1995 XP
Métodos de armonización utilizados	Mapping
Nivel de detalle de la propuesta	El autor se enfoca únicamente en las secciones de la norma ISO 12207:1995 correspondientes al proceso de desarrollo del software, con un nivel de detalle medio
Tipo de evaluación	Ninguno
Aspectos a destacar	
<p>El motivo del estudio según el autor, es que casi no hay directrices para incorporar a las metodologías ágiles, procesos que aseguren el cumplimiento con estándares específicos, el estudio por tanto sugiere algunas directrices, basadas en el análisis del estándar ISO usado actualmente (en 2003).</p> <p>El autor, divide la propuesta en varias secciones.</p> <p>Comentarios generales:</p> <ul style="list-style-type: none"> • Para asegurar que un equipo ágil es conforme a ISO 12207:1995, es importante asignar la tarea de asegurar el cumplimiento de la norma a uno o más individuos quienes puedan orientar al equipo en el objetivo. • Se sugiere, que similar al modelo organizacional de Brooks, también 	

conocido como “Surgical Team” sea seguido. El punto en particular valioso es la noción de que los programadores deberían estar aislados de tareas administrativas provenientes de la generación de la documentación.

Posteriormente el autor presenta varias tablas que muestran los procesos principales de ISO 12207:1995, entre los cuales está “Supporting Life Cycle Processes” que a su vez contiene 8 subprocesos: Documentación, gestión de la configuración, aseguramiento de la calidad, verificación, validación, revisión conjunta, auditoría y resolución de problemas. Según lo anterior el autor presenta el mapping a XP.

Verificación y validación se encuentra inherentemente gestionados por la práctica ágil de escribir el código específicamente para aprobar los casos de prueba, cada parte del código no es aceptado hasta que ha sido verificado contra los casos de prueba. También se hace verificación a través de las pruebas unitarias y funcionales.

La noción de revisión conjunta es fuertemente construida con XP. Reuniones de planeación regulares, Pair Programming, rotación de tareas de codificación, cliente en sitio. Esto asegura que las revisiones conjuntas son continuas en varios niveles de actividades y detalle.

La auditoría, puede ser respecto a un proceso donde un agente independiente revisa las actividades de la parte auditada, reportando los hallazgos en una manera formal. La necesidad de auditoría nace en un contexto donde las actividades son llevadas a cabo en una forma privada o independiente. Pero en un contexto XP, esta es la antítesis de la manera en que el código es desarrollado. En lugar de lo anterior, el código es visto como una propiedad colectiva y todos tienen acceso, además de que las actividades se rotan asegura que todos los diferentes miembros están constantemente expuestos y por lo tanto se vuelven familiares con el código desarrollado por otros.

Posteriormente el autor se enfoca en cláusulas más específicas.

“5.3.4.2 El desarrollador debe evaluar los requerimientos de software considerando los criterios listados abajo. Los resultados de las evaluaciones deben ser documentados.

- a) Trazabilidad a los requerimientos y diseño del sistema
- b) Consistencia externo con los requerimientos del sistema

- c) Consistencia interna
- d) Testabilidad
- e) Factibilidad del diseño de software
- f) Factibilidad de operación y mantenimiento”

XP explícitamente toma lo de arriba en consideración durante el “Planning Game”, en colaboración con los clientes en sitio. También se usan historias de usuario para los requerimientos.

Durante la implementación, la práctica de Pair Programming obliga al programador que no escribe para evaluar los criterios mencionados.

La Testabilidad es conducida explícitamente en términos del desarrollo dirigido por los tests, que es un principio aplicado en XP.

Cualquier otro asunto que pueda surgir, debería ser discutido en el contexto del equipo y si es necesario escrito como comentarios en el código fuente, bien sea para la lectura de otros o para la fácil extracción de documentación utilizando una herramienta.

“5.3.5 Diseño arquitectural de software

5.3.5.1 El desarrollador debe transformar los requerimientos de software a una arquitectura que describa con alto nivel la estructura y los componentes del software.

5.4.5.2 El desarrollador debe desarrollar y documentar un diseño de alto nivel para las interfaces externas al software y entre los componentes del mismo”

XP se basa en el uso de metáforas para facilitar el diseño arquitectónico del sistema. Dado que XP propone que el código fuente debería ser la documentación, existen tecnologías que extraen la documentación desde el código en una manera amigable al usuario.

“6.1. Proceso de documentación

- a) Implementación del proceso
- b) Diseño y desarrollo
- c) Producción
- d) Mantenimiento”

Lo que deba ser documentado es acordado contractualmente entre el adquiridor y el proveedor. La documentación requerida por quien adquiere el software junto con el que la organización requiere internamente, forman conjuntamente el documentación que debe ser generada.

“6.2.3 Control de configuración”

Actualmente hay un diverso rango de herramientas software para la gestión de la

configuración que puede permitir al equipo cumplir con el estándar, llevando a cabo un proceso de personalización según las necesidades.

“6.3 Aseguramiento de la calidad

- a) Proceso de implementación
- b) Aseguramiento del producto
- c) Aseguramiento del proceso
- d) Aseguramiento de los sistemas de calidad”

El aseguramiento de la calidad (QA) está construido dentro de XP a través tests funcionales, de aceptación y unitarios como también con la presencia del cliente en sitio. Sin embargo, una persona o sub equipo debería ser asignado para coordinar el QA en conjunto con el cliente en sitio.

“6.8 Resolución de problemas”

Aunque XP no especifica un proceso llamado “Resolución de problemas”, sus ciclos cortos de desarrollo iterativo y su fuerte interacción dentro del equipo naturalmente lleva a una detección temprana de problemas y su resolución. Si estas tareas deben ser documentadas se recomienda que lo haga como se mencionó anteriormente, un equipo encargado de la documentación para no sobre cargar a los desarrolladores.

Finalmente el autor hace un aporte sobre “Documentación incremental y concluye”.

Documentación incremental.

Las directrices de IEEE 12207.1-1997; IEEE 12207.2-1997 para ISO 12207:1995 establecen que cuando el desarrollo es incremental o evolutivo la documentación puede también ser incremental o evolutiva. Así, el uso de herramientas para generar la documentación desde el código fuente tienen el beneficio de asegurar que la documentación está actualizada y por lo tanto permite que se refleje la verdadera naturaleza del software.

Conclusión.

Las directrices provistas en este estudio no tienen la intención de ser exhaustivas. Más que eso proveen un punto de partida para desarrolladores ágiles quienes requieren cumplir con ISO 12207:1995. Las directrices deberían por supuesto ser probadas en práctica y personalizadas para necesidades de proyecto específicas. El

principio ágil de estresar el desarrollo de software funcionando más que grandes volúmenes de documentación no necesariamente son un rechazo a la producción de documentación per se.

Título	Structure of a Multi-model Catalog for Software Projects Management Including Agile and Traditional Practices
Publicación	Trends and Applications in Software Engineering Volume 405 of the series Advances in Intelligent Systems and Computing pp 87-97
Año	2015
Autor(es)	Andrés Felipe Bustamante, Jesús Andrés Hincapié, Gloria Piedad Gasca-Hurtado
Resumen	Los proyectos de desarrollo de software pueden ser gestionados bajo una gran variedad de metodologías y frameworks. El uso de frameworks tradicionales puede llevar a etapas de planeación extendida que toman una gran cantidad de tiempo. Las metodologías ágiles están diseñadas para acelerar la creación de valor a través de un proceso evolutivo incremental, donde actividades que crean más valor para el cliente son priorizadas. El uso de metodologías ágiles puede relegar factores importantes en gestión de proyectos, si no son incluidos en la fase de planeación, ya que las metodologías ágiles no proponen dimensiones que las metodologías tradicionales sí. Pretendemos identificar si es posible combinar metodologías ágiles y modelos tradicionales para definir un catálogo de mejores prácticas para la planeación de proyectos de desarrollo de software. Proponemos reducir la complejidad de implementar un modelo integrado ágil/tradicional, a través de un catálogo metodológico para la planeación de proyecto. Presentamos

	un diseño de alto nivel como solución, incluyendo conceptos iniciales del modelo para un futuro desarrollo del catálogo.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	El autor hace referencia a dos grandes sectores, los frameworks y modelos tradicionales y los ágiles. En los tradicionales el autor menciona a: PMI, CMMI, ISO 9001, TSP. En los ágiles el autor menciona a: Scrum, Kanban, XP.
Métodos de armonización utilizados	Catálogo multi modelo para la gestión de proyectos software
Nivel de detalle de la propuesta	La propuesta no es sobre cómo utilizar una metodología ágil con una tradicional si no cómo seleccionar las prácticas de ambos lados que se adapten a la organización para conformar una metodología a la medida
Tipo de evaluación	Ninguna
Aspectos a destacar	
<p>El objetivo del estudio según el autor: “Intentamos proveer una directriz en términos de dimensiones que tienen en cuenta prácticas específicas del mundo ágil y el tradicional”</p> <p>El efecto buscado del estudio según el autor: “Identificar las mejores prácticas de gestión de proyectos de desarrollo de software a partir de la integración de metodologías ágiles y frameworks tradicionales de referencia.”</p> <p>Posterior a una revisión de la literatura en las mejores prácticas del mundo ágil, tradicional y ambos, el autor destaca.</p> <p>“Spunda establece un análisis claro, identificando la idea básica detrás de los enfoques tradicionales, donde ‘se espera que los proyectos sean relativamente</p>	

simples, predecibles y lineales, con límites claramente definidos que permiten planeación detallada y ejecución sin muchos cambios”

“Varios estudios coinciden en que ‘La planeación en proyectos ágiles es muy consciente de la necesidad de incluir temas sobre sistemas sociales, enfocándose en la importancia de la cultura, el desarrollo de las personas, auto gestión, auto disciplina, toma de decisiones participativa, enfoque al usuario y menos burocracia”

“Es explicado por Torrecillas Salinas et al., donde los frameworks de referencia tradicionales contribuyen para establecer qué necesidades deben ser satisfechas, y las metodologías ágiles establecen cómo hacerlo. El análisis de los estudios seleccionados muestra que hay claras ventajas cuando se integran metodologías ágiles y frameworks de referencia tradicionales, como también hay algunas oportunidades de mejora”

El autor destaca los beneficios tanto de los frameworks tradicionales como de las metodologías ágiles.

“De las metodologías ágiles resaltamos cosas como la rápida retroalimentación, gran tolerancia al cambio, gestión de riesgos y un enfoque social que motiva a los clientes y los miembros del proyecto. De los frameworks tradicionales de referencia podemos resaltar que tienen una buena estructura de proyecto, artefactos claros y precisos y un buen entendimiento de las actividades clave y conjuntos de buenas prácticas para cubrir completamente el ciclo de vida del proyecto”.

Posteriormente el autor presenta el “Catálogo Multi modelo para la Gestión de Proyectos Software”

“Por mucho, hemos mostrado que la integración entre metodologías ágiles y frameworks de referencia tradicionales es una realidad que está siendo contemplada en la industria hoy e día. Con enfoques que analizan soluciones mezcladas, modelos abstractos y conjuntos de las mejores prácticas en ambos lados.”

“La solución propuesta a este problema es incluir catálogo de homologación entre diferentes metodologías. Tales catálogos considerarán categorización de prácticas por framework en orden de guiar el proceso de decidir qué usar y cómo usarlo para ensamblar una metodología auto definida, de acuerdo a las necesidades específicas

de la compañía y sin importar el modelo o estándar usado en ella. ...Proponemos 4 dimensiones.”

Dimensión 1 – Tipo de actividades.

Esta dimensión categoriza las actividades en dos tipos diferentes:

Estructurales: Entrega tangible de artefactos de planeación

Comportamiento: Dinámica de actividades para resolver una situación.

Dimensión 2 – Tamaño del proyecto.

Esta dimensión ajusta o evita la implementación de actividades, de acuerdo al tamaño del proyecto, el cual puede ser medido por horas, esfuerzo o complejidad.

Dimensión 3 – Actividades redundantes.

Esta dimensión identifica y unifica actividades con el mismo objetivo, haciendo explícitas las actividades y en lo que se centran.

Dimensión 4 – Actividades complementarias.

Esta dimensión agrupa y correlaciona actividades entre diferentes frameworks, identificando conectores de actividades que no pertenecen al mismo framework.

Las dimensiones buscan conducir problemas identificados en el estudio previo, tales como la necesidad de discernir entre qué hacer y cómo hacerlo, dando relevancia a la dimensión 1; la ejecución de tareas sin valor en cierto tipo de proyectos debido al modelo rígido, dando relevancia a la dimensión 2; y la descomposición de actividades sin una estructura clara, dando relevancia a las dimensiones 3 y 4.

Finalmente el autor concluye:

“En este estudio, hemos desempeñado un estudio de enfoques ágiles y tradicionales para la gestión de proyectos de software que nos llevaron a creer que es posible combinar actividades y técnicas de ambos lados. De acuerdo a esto, hemos hecho una propuesta inicial para la estructura de un catálogo metodológico multi modelo para la gestión de proyectos software, el cual establece rutas de actividades guiadas por diferentes dimensiones que son definidas en el catálogo mismo.

Título	The Agile and the Disciplined Software Approaches: Combinable or Just Compatible?
Publicación	Information Systems Development pp 35-49
Año	2013
Autor(es)	Lise Tordrup Heeager
Resumen	<p>A simple vista, los enfoques de desarrollo de software ágil y disciplinado parecen contradictorios. Más y más, organizaciones desarrolladoras de software de cualquier forma luchan al implementar un enfoque de desarrollo ágil aun siendo conformes a un estándar de aseguramiento de la calidad. Investigadores están discutiendo la combinabilidad y compatibilidad de estos dos enfoques. A través de una revisión de la literatura, el propósito de este capítulo es determinar si los enfoques de desarrollo de software ágil y disciplinado son combinables o sólo compatibles, en particular para identificar los principales desafíos de usar un enfoque de desarrollo de software ágil en una configuración de disciplina. La revisión muestra que los enfoques ágil y disciplinados son compatibles, pero no combinables. Es posible implementar prácticas ágiles y principios en un proceso de desarrollo conforme con un estándar de calidad, pero las regulaciones del estándar hacen imposible implementar un completo proceso ágil de desarrollo de software sin comprometer la agilidad. Los principales desafíos, al balancear la agilidad y la disciplina en un proyecto, son cómo determinar el nivel correcto de documentación y cómo superar las deferencias e la forma en que los requerimientos son manejados.</p>
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	El autor no analiza específicamente modelos o metodologías, sin embargo analiza dos grandes grupos. Metodologías ágiles (Scrum, XP)

	Metodologías tradicionales de disciplina (CMMI, ISO)
Métodos de armonización utilizados	Ninguno
Nivel de detalle de la propuesta	Propuesta de muy alto nivel
Tipo de evaluación	Revisión sistemática de la literatura
Aspectos a destacar	
<p>El objetivo del estudio según el autor es averiguar si las metodologías ágiles y las tradicionales son combinables o compatibles. Para esto el autor presenta las definiciones para cada término.</p> <p>Combinar es mezclar dos o más cosas o grupos juntos para formar una sola cosa, significa que uno es capaz de hacer dos cosas al mismo tiempo sin comprometer las dos.</p> <p>Ser compatible es ser capaz de “existir juntos”, significa que algo es capaz de trabajar al mismo tiempo con otra cosa.</p> <p>Así, combinabilidad es un concepto booleano mientras que compatibilidad puede ser calificada.</p> <p>Los enfoques del software ágil y disciplinado son combinables si es posible usar un enfoque ágil de desarrollo de software y al mismo tiempo cumplir con un estándar de aseguramiento de la calidad sin comprometer la agilidad del enfoque.</p> <p>Los enfoques del software ágil y disciplinado son compatibles si es posible usar algunas de las prácticas y principios ágiles y al mismo tiempo ser capaz de cumplir con un estándar de aseguramiento de la calidad.</p> <p>La revisión de la literatura muestra que investigaciones previas no distinguen entre combinabilidad y compatibilidad de los enfoques ágiles y disciplinados. Varios estudios establecen que los enfoques ágiles y disciplinados son combinables; de cualquier forma, cuando se lee el análisis y los resultados presentados en los</p>	

estudios adoptando las definiciones de este capítulo, la conclusión debería ser en lugar que los enfoques ágiles y disciplinados son compatibles, pero no combinables.

El autor revisa 79 estudios que analizan las ventajas de los enfoques ágiles y tradicionales y las mejores prácticas, a través de un proceso bien definido, encuentra que principalmente los estudios analizan las metodologías ágiles y los estándares ISO 9000 e ISO 9001. Como también analizan las metodologías ágiles y los modelos CMM y CMMI.

Respecto a las metodologías ágiles e ISO.

Los métodos ágiles pueden ser adaptados para asegurar compatibilidad con los estándares ISO. Basado en evidencia empírica, Wright concluye que las compañías que usan XP pueden cumplir con los requerimientos de ISO 9001. McMichael y Lombardi concluyen que ISO 9001:2000 es capaz de ayudar a asegurar que los procesos ágiles son seguidos correctamente. Melis concluye que varios requerimientos de ISO 9001:2000 son implementados por las herramientas existentes en la gestión del proyecto de XP, mientras que Stalhane y Hanssen sugieren algunos cambios a ambos, ISO y el enfoque ágil. Lami y Falcini mostraron que ISO/IEC 15504 en principio es aplicable a contextos ágiles, pero en la práctica pueden ocurrir problemas, por ejemplo al crear un modelo de referencia de proceso ágil y en orden de eso encontrar un asesor que tenga experiencia en el desarrollo de software en contextos ágiles.

El autor hace 4 proposiciones.

Proposición 1: Los enfoques ágiles y disciplinados son compatibles, no combinables

Proposición 2: Obtener compatibilidad de los enfoques ágiles y disciplinados de desarrollo de software requieren una extensión de tanto uno o los dos enfoques.

Los desafíos al obtener compatibilidad:

1. La documentación: Una diferencia entre los enfoques controlados por los estándares de calidad y los enfoques ágiles es la cantidad de documentación requerida.

Proposición 3: Para obtener compatibilidad entre los enfoques ágiles y disciplinados, el enfoque ágil debe concentrarse no sólo en software funcionando sino también en algo de documentación.

2. Los requerimientos: La metodología ágil XP por ejemplo, basa los

requerimientos en historias de usuario creadas por el cliente, estas diferencias en cómo los enfoques ágiles y disciplinados manejan los requerimientos pueden causar problemas al obtener compatibilidad, sin embargo hay varios estudios que presentan algunas alternativas (Ver el estudio)

Proposición 4: Las diferentes estrategias para gestionar los requerimientos propuestos por los enfoques ágiles y disciplinados pueden causar problemas al tratar de obtener compatibilidad.

Finalmente el autor concluye:

El análisis ha mostrado que los enfoques ágiles y disciplinados son altamente compatibles, pero no combinables. En la práctica, es posible implementar varias prácticas y principios ágiles en un proceso de desarrollo de software y al mismo tiempo ser capaz de cumplir con un estándar de aseguramiento de la calidad; de cualquier forma, la agilidad de un proyecto y/o las prácticas y principios ágiles en el proceso de desarrollo de software serán afectados en cierto grado por las regulaciones del modelo de proceso de un estándar de calidad.

Título	The Application of ISO 9001 to Agile Software Development
Publicación	PROFES '08 Proceedings of the 9th international conference on Product-Focused Software Process Improvement Pages 371-385
Año	2008
Autor(es)	Tor Stalhane, Geir Hjtíl Hanssen
Resumen	En este estudio, discutimos cómo reconciliar el enfoque ágil en velocidad y desarrollo ligero con la necesidad de documentación, trazabilidad y control de ISO 9001. Vemos que no hay necesidad de cambiar ni ISO 9001 ni el concepto ágil. En lugar de eso, vemos una necesidad de ser flexible al

	usar términos como planeación y evidencia de conformidad. Es verdad que podemos incluir todo en el desarrollo ágil al hacerlo un requerimiento, pero hay un límite en cuantos documentos requerimos de un proceso ágil sin destruir el concepto de agilidad.
Descripción	
Modelo de calidad y/o metodología ágil utilizadas	ISO 9001. SCRUM/XP
Métodos de armonización utilizados	Mapping
Nivel de detalle de la propuesta	El autor revisa minuciosamente todos los ítems relevantes de la norma ISO 9001.
Tipo de evaluación	Evaluación de expertos, incluyendo investigadores, estudios y auditores de la norma
Aspectos a destacar	
<p>Inicialmente, el autor revisa la literatura y hace comentarios sobre algunos estudios revisados.</p> <p>“En su tabla ISO 9001 vs XP, Wright ha revisado el ítem 7.3.4 de ISO 9001 ‘En etapas adecuadas, revisiones sistemáticas del diseño y desarrollo deben ser realizadas en siguiendo acuerdos planeados’. El autor dice que Pair Programming es una revisión continua del código. Estas palabras, de cualquier forma no se mantienen vigentes según la mayoría de definiciones disponibles de una revisión del código”.</p> <p>“En la tabla de Wright de TickIT vs XP, el dice “las historias de usuario y tests de aceptación totalmente definen los requerimientos de software”. Hay dos problemas con esta afirmación, Primero, que los tests de aceptación definen los requerimientos es incorrecto. Los tests de aceptación están escritos basados en los requerimientos, no de otra forma, aunque la nueva tendencia de testing de aceptación automatizado puede cambiar esto, segundo que las historias de usuario</p>	

son bastante imprecisas para servir como requerimientos. Son las historias de usuario más la aceptación del cliente, a menudo no descrito así, lo que define los requerimientos.”

“Erharuyi revisa la parte 7 y 8 de ISO 9001, igual que Nawrocki. Cómo debería esperarse, sus conclusiones son bastante iguales que las de Nawrocki. De cualquier forma, su estudio contiene algunos conceptos evidentemente malentendidos, tales como la afirmación que las actualizaciones al plan de testing es parte de las acciones correctivas- ISO 9001, ítem 8.5.2.”

Posteriormente el autor afirma.

“Un tipo de crítica contra las metodologías ágiles es la deliberada evitación de la documentación ya que puede ser considerada como desperdicio; documentos no son software y el desarrollo de software debería desarrollar software, no documentos. Este es un concepto estricto y probablemente explica la percepción común que el desarrollo ágil de software es incompatible con los bien conocidos estándares de calidad como ISO 9001. Como mostraremos más tarde, esto de cualquier forma no necesariamente es verdad”

El autor establece algunas premisas relevantes para el estudio.

“Para el propósito principal de este proyecto – decidir qué es conforme a ISO 9001 y que no, usamos 3 enfoques:

1. Lo que ha sido aceptado por otros auditores
2. Nuestro propio juicio, basado en desarrollar un proceso conforme a ISO 9001
3. La experiencia de dos auditores de DnV- “Detnorske Veritas”

El proceso nos dejó con 15 ítems donde las metodologías ágiles sólo son parcialmente capaces de cumplir con los requerimientos de ISO 9001 y 4 ítems donde las metodologías ágiles no pudieron cumplir los requerimientos en absoluto. Como debería esperarse, la parte 7 de ISO 9001 domina en ambos casos – 9 fuera de los 15 de los ítems parcialmente cumplidos y 2 de los 4 ítems que no fueron cumplidos. En el lado positivo – de los 50 ítems en ISO 9001, 31 no necesitarán ningún cambio o mejora por parte de ningún lado”

¿Qué se puede hacer para conseguir conformidad?

4.2.1.d: La documentación del sistema de gestión de la calidad debe incluir documentos necesitados por la organización para asegurar la planeación efectiva,

operación y control de sus procesos.

ACCIÓN/ Como para cualquier clase de metodología de desarrollo, un proyecto de desarrollo ágil siempre comienza por definir cómo la metodología será usada en el proyecto dado. La planeación de un proyecto ágil puede fácilmente ser documentada en una forma simple especificando por ejemplo la longitud de iteración, como registrar y trazar requerimientos etc.

4.2.4: Registros deben ser establecidos y mantenidos para proveer evidencia de conformidad con los requerimientos y la operación efectiva del sistema de gestión de la calidad. Registros deben permanecer legibles, identificables y entregables. Un procedimiento documentado debe ser establecido para definir los controles necesarios para la identificación, almacenamiento, protección, entrega, tiempo de retención y disposición de los registros.

ACCIÓN/Entre iteraciones, conformidad con los requerimientos es evaluada por el Product Owner y los registros de los resultados son mantenidos como evidencia de conformidad.

5.3: La alta dirección debe asegurar que la política de calidad

5.3 a: es apropiada para el propósito de la organización

5.2 b: incluye un compromiso para cumplir con los requerimientos y la mejora continua de la efectividad del sistema de gestión de la calidad.

ACCIÓN/ Un proceso ágil bien implementado, con las condiciones necesarias en el lugar correcto soportará el propósito de la organización. Que es entregar software que funciona bien dentro del tiempo y costo establecido. Un buen proceso ágil asegurará un compromiso con los requerimientos ya que son continuamente evaluados basados en la experiencia del desarrollo y el testing.

5.4.1. La alta dirección debe asegurar que los objetivos de calidad, incluyendo aquellos necesarios para cumplir los requerimientos del producto, son establecido a funciones relevantes y niveles dentro de la organización. Los objetivos de calidad deben ser medibles y consistentes con la política de calidad.

ACCIÓN/ El método ágil EVO enfatiza objetivos de calidad medibles y esta práctica puede ser fácilmente aplicada en otros métodos ágiles y cumplir con el requerimiento de la norma.

5.6.2: La entrada para revisión por la dirección debe incluir información de

5.6.2 a: Resultados de auditorías

5.6.2 a: Retroalimentación del cliente

5.6.2 c: Desempeño del proceso y conformidad del producto.

ACCIÓN/ Después de cada iteración, el desempeño del proceso es revisado y potencialmente mejorado en la siguiente iteración. Las revisiones están basadas en las entradas de los desarrolladores y la retroalimentación del cliente. Las salidas de tales retrospectivas pueden en el contexto de ISO 9001 ser usadas como entradas para las revisiones por la dirección.

7.1: La organización debe planear y desarrollar los procesos necesarios para la realización del producto. Planeación de la realización del producto debe ser consistente con los requerimientos de otros procesos del sistema de gestión de la calidad. En la planeación de la realización del producto, la organización debe determinar lo siguiente como sea apropiado:

7.1 a: objetivos de calidad y requerimientos del producto

7.1 b: La necesidad de establecer procesos, documentos y proveer recursos específicos para el producto

ACCIÓN/ La adaptación de un proceso ágil al inicio del desarrollo del proyecto cubre este requerimiento. Adaptación, o planeación del proceso puede incluir la decisión de la longitud de la iteración, estrategias para la documentación de los requerimientos, personal etc.

7.2.1 a: La organización debe determinar los requerimientos especificados por el cliente incluyendo los requerimientos para la entrega y actividades post-entrega.

ACCIÓN/ Los procesos ágiles incluyen prácticas para determinar los requerimientos pero son usualmente enfocadas en características y calidad del producto como tal. Para cubrir el requerimiento de determinar los requerimientos para entrega y actividades post-entrega, esto necesita ser incluido. Una forma de manejar esto es agregar sprints adicionales después de la entrega para enfrentar cualquier actividad post-entrega.

7.2.2: La organización debe revisar los requerimientos relacionados al producto. La revisión debe ser conducida antes del compromiso de la organización a proveer el producto al cliente y asegurar que:

7.2.2 a: Los requerimientos del producto son definidos

7.2.2 c: La organización tiene la habilidad para cumplir con los requisitos definidos

7.2.2 x2: Donde el cliente no provea documentación de los requerimientos, estos deben ser confirmados por la organización antes de ser aceptados.

ACCIÓN/ Este requerimiento es el más problemático si no es interpretado estrictamente. Requiere que organización tenga los requerimientos completos definidos desde el inicio. De cualquier forma, las metodologías ágiles especifican

que los requerimientos deberían ser recolectados al inicio. Sin embargo no completos y sin todos los detalles. Esto basado en la asunción que es imposible obtener una completa vista general de todos los detalles desde el inicio; en lugar de eso los aspectos más importantes deben ser documentados. En Scrum esto es documentado en el Product Backlog el cual es establecido antes de la primera iteración, en este momento constituye el mejor posible entendimiento de los requerimientos. Comparado a los requerimientos tradicionales difieren en la forma en que están anticipados al cambio, basados en la experiencia del desarrollo. La conclusión de este asunto es que si un auditor acepta esta vista general de los requerimientos, los métodos ágiles cumplen el requerimiento, Sino encontramos que el principio fundamental de requerimientos evolutivos en los métodos ágiles está en conflicto con ISO 9001.

7.3.1 a: La organización debe planear y controlar el diseño y desarrollo del producto. Durante la planeación del diseño y desarrollo, la organización debe determinar las etapas del diseño y desarrollo

ACCIÓN/ Los métodos ágiles cubren la planeación y control del diseño del producto. Difiere de los enfoques tradicionales en que es hecho iterativa e incrementalmente, todavía son gestionados. Esto aún producirá documentos que puede ser usados como prueba de conformidad con las actividades mencionadas en 7.3.1.a Ejemplos de documentos que han sido aceptados como prueba de conformidad son por ejemplo fotos de las reuniones de requerimientos planeados.

7.3.2 a: Requerimientos funcionales y de desempeño

7.3.2 x1: Estas entradas deben ser revisadas cuando sea adecuado: Los requerimientos deben ser completos, no ambiguos y no estar en conflicto entre ellos

ACCIÓN/Esto es manejado a través del proceso de requerimientos en los métodos ágiles. Al inicio de cada iteración, nuevos requerimientos son recolectados o los requerimientos existentes son actualizados debido a la retroalimentación del cliente. Estos requerimientos son luego revisados y registrados a través de la cooperación entre el cliente y el equipo de desarrollo. Esto incluye ambos tanto requerimientos funcionales como de desempeño.

7.3.3: Las salidas del diseño y desarrollo deben ser provistas en una forma que permita la verificación contra las entradas del diseño y desarrollo y deben ser aprobadas antes de la liberación. Las salidas del diseño y desarrollo deben:

7.3.3 a: Cumplir los requerimientos de las entradas del diseño y desarrollo

ACCIÓN/Parece ser que hay cierta cantidad de desacuerdo en el campo ágil como

si por ejemplo XP requiere el diseño pero no todo al mismo tiempo, no es problema incluir una actividad de diseño de alto nivel en la primera planeación y una de diseño de bajo nivel al inicio de cada iteración.

7.3.4: En las etapas adecuadas, las revisiones sistemáticas del diseño y desarrollo deben ser llevadas a cabo de acuerdo con lo planeado

7.3.4 a: Para evaluar la usabilidad de los resultados del diseño y desarrollo para cumplir los requerimientos

ACCIÓN/El diseño y desarrollo es revisado en cada transición entre iteraciones como un esfuerzo conjunto entre el equipo de desarrollo y el cliente. Al cliente se le da una particular responsabilidad de evaluar la usabilidad y el cumplimiento de los requerimientos.

7.3.5: La verificación debe ser llevada a cabo de acuerdo a lo planeado (ver 7.3.1) para asegurar que las salidas del diseño y desarrollo han cumplido con los requerimientos del diseño y desarrollo. Los registros de los resultados de la verificación y cualquier acción necesaria deben ser mantenidos.

ACCIÓN/No es un problema incluir una actividad de diseño de alto nivel en la primera planeación y una de bajo nivel al inicio de cada iteración.

7.3.7: Los cambios del diseño y desarrollo deben ser identificados y los registros mantenidos. Los cambios deben ser revisados, verificados y validados como sea apropiado y aprobados antes de la implementación. La revisión de los cambios del diseño y desarrollo debe incluir la evaluación del efecto de los cambios en partes constituyentes y productos ya liberados. Los registros de los resultados de la revisión de los cambios y cualquier acción necesaria deben ser mantenidos (ver 4.2.4).

ACCIÓN/La revisión hecha después de cada iteración se encarga de esto. Es hecho como un esfuerzo conjunto entre el equipo, el cliente o Product Owner y otros posibles interesados los cuales entre otros asuntos consideran los cambios al diseño y desarrollo. Las decisiones son documentadas en la forma de un Product Backlog actualizado. Si es necesario, actas de reuniones formales y firmadas pueden ser hechas para mantener el rastro de la historia del diseño y desarrollo.

8.1: La organización debe planear e implementar el monitoreo, medición, análisis y los procesos de mejorar necesarios.

8.1 a: Para demostrar conformidad con el producto

8.1 b: Para asegurar conformidad del sistema de gestión de la calidad

ACCIÓN/Esto es cubierto por la planeación y la adopción del método ágil que está

siendo usado. Una parte central de todos los métodos ágiles es monitorear de cerca el progreso para descubrir potenciales problemas rápidamente. Las revisiones hechas entre iteraciones también incluyen una evaluación del proceso de desarrollo como tal para potencialmente identificar iniciativas de mejora del proceso software.

8.2.3: La organización debe aplicar métodos adecuados para monitorear y, donde sea aplicable, medir el proceso del sistema de gestión de la calidad. Estos métodos deben demostrar la habilidad del proceso para lograr los resultados planeados. Cuando los resultados planeados no son conseguidos, acciones correctivas deben ser tomadas, como sea apropiado para asegurar la conformidad del producto. ACCIÓN/ Además de la evaluación de los incrementos del producto, la revisión entre iteraciones también puede incluir una retrospectiva. Esto tiene la misma función que una evaluación tradicional del desempeño del proceso, potencialmente llevado a que acciones para la mejora del proceso sean implementadas en las iteraciones siguientes.

8.2.4: La organización deben monitorear y medir las características del producto para verificar que los requerimientos del producto han sido cumplidos. Esto debe ser llevado a cabo en etapas apropiadas del proceso de realización del producto de acuerdo a lo planeado (ver 7.1).

Evidencia de conformidad con los criterios de aceptación deben ser mantenidos. Los registros deben indicar la(s) persona(s) que autorizan la liberación del producto (ver 4.2.4).

ACCIÓN/Esto es manejado por las revisiones de iteración. La aceptación de los requerimientos es documentada por ejemplo en el Product Backlog o similar.

8.5.2: La organización debe tomar acciones para eliminar las causas de la no conformidad en orden de prevenir la recurrencia. Las acciones correctivas deben ser apropiadas para los efectos de la no conformidad encontrada.

Un procedimiento documentado debe ser establecido para definir los requerimientos para:

8.5.2 c: evaluar la necesidad de acciones para asegurar que las no conformidades no son recurrentes

8.5.2 d: determinar e implementar las acciones necesarias

8.5.2 e: los registros de los resultados de las acciones tomadas (ver 4.2.4)

8.5.2 f: revisar las acciones correctivas tomadas

ACCIÓN/ Las revisiones de iteración tratan de descubrir no conformidad con los requerimientos. Esto es causado tanto por pocos recursos en la iteración previa

debido a dificultades no previstas, poco entendimiento de los requerimientos o un mal proceso. Sólo el último caso es de interés aquí. Las causas del proceso son registradas y posteriormente serán usadas como entrada a una actividad de mejora del proceso.

8.5.3: La organización debe tomar acciones para eliminar la causa de potenciales no conformidades en orden de prevenir su recurrencia. Las acciones preventivas deben ser apropiadas para los efectos de los potenciales problemas.

Un procedimiento documentado debe ser establecido para definir los requerimientos para:

8.5.3 a: determinar las potenciales no conformidades y sus causas

8.5.3 b: evaluar la necesidad de acciones para prevenir la ocurrencia de no conformidades

8.5.3 c: determinar e implementar las acciones necesarias

8.5.3 e: revisar las acciones preventivas tomadas

ACCIÓN/ La intención de tener revisiones frecuentes del progreso del desarrollo, los requerimientos y el desempeño del proceso en cooperación con el cliente es, entre otras cosas, para eliminar las causas de potenciales no conformidades. Tanto el producto software como el conocimiento relacionado crece, el equipo de desarrollo y el cliente continuamente mejoran su habilidad para descubrir potenciales causas de no conformidad. Tales revisiones puede, si es relevante, producir acciones de mejora de proceso – ambos para reactivamente tomar acciones inmediatas y como un medio para proactivamente mejorar el proceso de desarrollo para proyectos posteriores.

Finalmente el autor concluye:

“Hay formas de tratar con muchos de los documentos que ISO 9001 requiere.

Podemos agregar actividades como reuniones de revisión, escribir documentos de diseño y así. El proceso aún mantendrá las ideas ágiles más importantes, tales como iteraciones cortas, construir por incrementos, incluir al cliente, re priorizar los requerimientos cuando sea necesario, y ajustar el constantemente el alcance, el tiempo y costo dentro de los límites del contrato del proyecto. Un slogan usado a menudo en la comunidad ágil es “Haz la cosa más simple que podría posiblemente funcionar”. El término “simple” no significa que está prohibido agregar artefactos o actividades extra al proceso. Hay, de cualquier forma, límites a cuántos artefactos pueden ser agregados a un método ágil y aún insistir en catalogarlo ágil. Los cambios necesarios para ser conforme a ISO 9001 están, de cualquier forma, dentro de tales límites.

Título	The effect of moving from a plan-driven to an incremental software development approach with agile practices An industrial case study.
Publicación	Empirical Software Engineering December 2010, Volume 15, Issue 6, pp 654–693
Año	2010
Autor(es)	Kai Petersen, Claes Wohlin
Resumen	<p>Por mucho, algunos pocos estudios en profundidad se han enfocado en la comparación directa de modelos de proceso en general, y en particular entre enfoques dirigidos por el plan y ágiles/incrementales. Eso es, que no se ha hecho explícito el efecto de moverse de un modelo a otro. Por lo tanto, hay evidencia limitada sobre las ventajas y los problemas encontrados en el desarrollo ágil de software, esto es específicamente cierto en el contexto de del desarrollo a gran escala. El objetivo de este estudio es investigar cómo la percepción del cuello de botella, trabajo innecesario, y re trabajo (de aquí en adelante referido como problemas) cambia cuando se migra de un enfoque de desarrollo dirigido por el plan a uno con prácticas ágiles (Product Backlog flexible, interacción cara a cara e integración frecuente), y cómo son percibidas comúnmente estas prácticas a través de los diferentes roles del sistema y del desarrollo. El contexto en el cual el objetivo debería ser conseguido es un desarrollo a gran escala con un enfoque dirigido por el mercado. La selección del contexto fue basada en la observación del trabajo relacionado que en su mayoría ha investigado pequeños proyectos de desarrollo de software y la investigación se ha enfocado en un modelo ágil (eXtreme programming). Un caso de estudio fue llevado a cabo en un sitio de desarrollo de Ericsson AB, ubicado en Suecia al final de 2007. En total 33 entrevistas fueron realizadas en orden de investigar el cambio percibido cuando migraron de</p>

un desarrollo de software dirigido por el plan a uno ágil e incremental, siendo las entrevistas la primera fuente de evidencia. Para propósitos de triangulación las mediciones realizadas por Ericsson fueron consideradas, las mediciones relacionadas al trabajo innecesario (cantidad de requerimientos descartados) y el re trabajo (datos de eficiencia del testing y esfuerzo de mantenimiento). La triangulación en este contexto significa que las mediciones fueron usadas para confirmar los cambios percibidos con una fuente adicional de datos. En total 64 problemas fueron identificados. 24 siendo de naturaleza general y los 40 restantes siendo locales y por lo tanto únicos a opiniones individuales o a un sistema en específico. Los más comunes fueron documentados y analizados en detalle. La comunalidad se refiere a cuántas personas en diferentes roles y a través de los sistemas estudiados han mencionado los problemas para cada uno de los modelos de proceso. La mayoría de los problemas más comunes se relacionan con el desarrollo dirigido por el plan. También identificamos problemas comunes que permanecieron en la agilidad después de la migración, los cuales fueron relacionados al tiempo de espera por el testing, el cubrimiento del test, la liberación de software, y la coordinación. Las mejoras fueron identificadas ya que muchos problemas que aparecieron por el enfoque dirigido por el plan no volvieron a aparecer en el enfoque ágil e incremental. Se concluye que la introducción reciente (comienzo en 2005 con el estudio siendo llevado a cabo al final del 2007) de las prácticas ágiles brinda valor agregado en comparación al enfoque dirigido por el plan, el cual es evidente desde la ausencia de problemas críticos que están apareciendo en el desarrollo dirigido por el plan.

Descripción

Modelo de calidad y/o metodología ágil utilizadas	Métodos dirigidos por el plan. Métodos ágiles (Scrum, XP)
--	--

Métodos de armonización utilizados	No especificado
Nivel de detalle de la propuesta	El autor toca temas de ingeniería del software con gran profundidad, sin embargo no menciona aspectos relacionados a la programación, estándares, infraestructura etc. Es una propuesta de alto nivel
Tipo de evaluación	Estudio de caso en Ericsson AB, en Suecia.

Aspectos a destacar

En la introducción el autor menciona:

“El enfoque dirigido por el plan fue usado en Ericsson por varios años, Debido a las referencias de la industria y de igual forma por problemas de desempeño (como tiempos de espera) Ericsson adoptó primero prácticas incrementales comenzando en la mitad de 2005. Posteriormente prácticas ágiles fueron agregadas a finales de 2006 y comienzos de 2007. En general vamos a mostrar que el modelo de Ericsson comparte prácticas con el desarrollo incremental, Extreme Programming y Scrum.”

“Hemos identificado los problemas y ventajas en el desarrollo dirigido por el plan, y cómo esos problemas y desventajas han cambiado después de la migración a prácticas ágiles e incrementales.”

De una revisión de estudios empíricos en la comparación de modelos el autor menciona varias conclusiones de los diferentes autores, algo relevante fue.

“Un problema del desarrollo ágil es que los miembros no son fácilmente intercambiables como en el modelo de desarrollo orientado en cascada (Baskerville et al 2003).”

Además el autor menciona respecto a los diferentes estudios que comparan modelos y el actual estudio.

“Aun cuando este estudio no se enfoca en productividad, las observaciones en los estudios revisados sobre productividad mostraron un significativo efecto positivo de la introducción al desarrollo ágil.”

Al final de la introducción el autor concluye, con base en la revisión de la literatura que hacen falta estudios empíricos que investigue la situación del modelo base antes de la migración al enfoque ágil y así pueda comparar ambas situaciones para emitir un juicio

“En respuesta al vacío de investigación, este estudio investiga la situación base para juzgar el efecto de la migración hacia el enfoque ágil incremental.”

Respecto al enfoque dirigido por el plan en Ericsson el autor menciona.

“Los principales pasos del proceso eran ingeniería de requerimientos, diseño e implementación, testing, entrega y mantenimiento.”

Respecto al enfoque ágil e incremental en Ericsson el autor menciona:

“El proceso yace en un conjunto de prácticas específicas para la compañía que han sido introducidas

1. Product Backlog
2. Plan de anatomía: Basado en las dependencias entre partes del sistema que son implementadas en cada proyecto (proyectos pequeños de 3 meses). Las dependencias eran un resultado de la arquitectura del sistema problemas técnicos y dependencias de requerimientos. El plan de anatomía resultó en un número de líneas base llamadas LSV (última versión del sistema) que necesitaban ser desarrolladas
3. Equipos pequeños y línea de tiempo: Los paquetes de requerimientos fueron implementados por equipos pequeños en proyectos cortos de aproximadamente 3 meses. La duración de cada proyecto determinó el número de requerimientos seleccionados para un paquete.
4. Uso de LSV (última versión del sistema) como entregable a producción: Si un proyecto era integrado con la última base del sistema, una nueva base era creada (LSV). Por tanto sólo una base existía a la vez, ayudando a reducir el esfuerzo de mantenimiento del producto.
5. Desacoplado de desarrollo del entregable del cliente: Si cada entregable hubiera sido puesto en producción, habrían sido muchos entregables en uso por clientes necesitando soporte. En orden de evitar esto, no todos los LSV eran puestos en producción.”

Respecto a los modelos de proceso generales (Métodos oficiales), el autor comenta.

“Para poder generalizar los resultados de este estudio, las características del modelo ágil e incremental usado en Ericsson fueron mapeadas a los modelos existentes del desarrollo iterativo incremental XP y SCRUM.

El modelo usado en Ericsson comparte 5 de 12 principios con XP y 6 de 10 con Scrum.

- Iteraciones e incrementos
- Entregables interno y externos
- Time Boxing
- No cambios a proyectos iniciados
- Interacción cara a cara frecuente
- Product Backlog flexible
- Integración continua”

Posteriormente respecto al contexto del estudio de caso el autor menciona:

“Ericsson es una de las compañías de telecomunicaciones más grandes en el mundo ofreciendo productos y servicios en este dominio incluyendo soluciones para teléfonos móviles, soluciones multimedia y de infraestructura de red. La compañía está certificada con ISO 9001:2000. El desarrollo de Ericsson es dirigido por el mercado y caracterizado por un mercado cambiante. Por lo tanto, el mercado demanda soluciones altamente personalizadas”.

“Durante el proceso del diseño de la investigación el autor muestra 3 subsistemas a estudiar en la organización:

- Subsistema 1: C++, 300k líneas de código, 43 personas
- Subsistema 2: C++, 850k líneas de código, 53 personas
- Subsistema 3: Java, 24k líneas de código, 17 personas

las áreas de preguntas para el levantamiento de problemas en cada modelo:

- Cuellos de botella: componentes que estorban en el desarrollo del proceso
- Trabajo innecesario: actividades que no contribuyen a la creación de valor para el cliente
- Re trabajo evitable: el re trabajo puede ser evitado haciendo las cosas por completo, consistentes y correctamente

La división de inconvenientes entre locales y globales es así:

- Inconvenientes globales: Expresados por los entrevistados que representaban más de un rol y más de un subsistema

- Inconvenientes locales: Expresados por uno o varios entrevistados representando un rol o un subsistema.

Respecto a los inconvenientes globales se clasificaron así:

- Inconvenientes generales: Más de 1/3 de entrevistados
- Inconvenientes muy comunes: Más de 1/5 de entrevistados
- Inconvenientes comunes: Más de 1/10 de entrevistados
- Otros inconvenientes: Menos de 1/10 de entrevistados

Respecto a las mejoras percibidas se clasificaron así:

- Comúnmente percibidos: Más de 1/10 de entrevistados
- Observación: Menos de 1/10 de entrevistados”

Los resultados cualitativos de la investigación arrojaron como el autor menciona:

“En total 64 inconvenientes fueron identificados de los cuales 24 fueron de naturaleza general y los 30 restantes fueron problemas locales relacionados con las experiencias de individuos o subsistemas específicos. ...Todo esto dio 13 inconvenientes para análisis detallados de los cuales 2 se consideraron generales, 3 se consideraron muy comunes y 8 se consideraron comunes.”

Los inconvenientes generales fueron respecto a los requerimientos documentados y validados que no fueron implementados y respecto a la verificación en cuanto al reducido tiempo de testing al final del proyecto, ambos en el modelo dirigido por el plan.

Los inconvenientes muy comunes fueron respecto a la verificación en cuanto a la cantidad de errores que se incrementó debido por la etapa tardía de testing y los errores encontrados que eran muy difíciles de reparar, estos dos en el modelo dirigido por el plan. En el enfoque ágil e incremental se encontró un inconveniente con la verificación debido a que los tiempos de espera del testing de LSV hacían esperar los proyectos que necesitaban el entregable como entrada.

Respecto a los datos cuantitativos el autor menciona:

“Una vista general de los datos cuantitativos usados para confirmar o contradecir los hallazgos del análisis cualitativo.

En cuanto a los requerimientos descartados respecto al método dirigido por el plan 1856 implementados, 649 descartados (26% desperdiciado)

Respecto al método ágil

1614 implementados, 73 descartados (4% desperdiciado).

Además, hubo un 31% de errores que se escaparon antes de la fase de testing en el desarrollo dirigido por el plan.

Y un 19% de errores que se escaparon antes del testing de cada LSV.”

Respecto a los errores reportados por los clientes que generaron costo de mantenimiento el autor presenta una gráfica con una línea que indicaba que los costos de mantenimiento llegarían a ser 40% más en 2007 de lo que eran en 2005, sin embargo después de la migración los costos fueron casi iguales en 2007 que en 2005 e incluso mostraban una tendencia a seguir reduciendo.

En el análisis de la investigación el autor menciona:

“Una clara mejora puede ser vista en la reducción de basura, mostrada en el análisis cualitativo y soportada por el análisis cuantitativo, además el número de solicitudes de cambio han sido reducidos lo cual es un indicador de que los requerimientos son un mejor reflejo de las necesidades del cliente que lo que eran en el enfoque dirigido por el plan. ...Finalmente, el alcance reducido en el nuevo enfoque de desarrollo ayuda a tener estimaciones más precisas, significando que el alcance de los requerimientos es establecido de manera más apropiada para cada incremento, así es menos probable que los requerimientos tengan que ser descartados debido a planeación no apropiada.”

Respecto a la mejora en la calidad del software el autor menciona:

“La mejora en la calidad de software se reflejó en una reducción de los errores escapados antes del testing en comparación con el enfoque dirigido por el plan, el esfuerzo de mantenimiento constantemente creciente fue reducido después de la introducción de las prácticas ágiles, ...el testing fue mejorado debido a la temprana detección de fallos y retroalimentación de las pruebas. ...El estudio muestra que incluso aunque ha habido mejoras en el testing, problemas muy importantes se relacionan al usar prácticas ágiles e incrementales.”

Respecto a la mejora en la comunicación el autor menciona:

“Los datos cualitativos sugieren una mejora en la comunicación después de haber puesto a las personas juntas. ..., la cantidad de documentación puede ser reducida porque mucha de ella estaba relacionada al cambio de roles entre fases que necesitaba mucha documentación para que cada miembro entendiera el sistema, ahora la comunicación directa reemplaza partes de la documentación. ...tener

equipos pequeños requiere un gran conocimiento por parte de los miembros del equipo, ...la interacción cara a cara ayuda a los miembros del equipo a aprender de los otros y ganar visión y entendimiento del proceso de desarrollo en general”

Respecto a la verificación el autor menciona:

“Para la verificación, la mejora de los tiempos de espera reducidos para el testing ha sido identificada, de cualquier forma el inconveniente se relaciona al concepto de los ciclos de tiempo de LSV que no ha sido optimizado, hay una oportunidad de mejora aquí para reducir los tiempos de espera del testing.”

Respecto a la sobrecarga de la dirección el autor menciona:

“De cualquier forma, muchos proyectos trabajando hacia la misma meta tienen que ser coordinados. Como se discutió antes, esto requiere mucha comunicación y planeación involucrando muchas personas diferentes.”

Respecto al proyecto de entrega el autor menciona:

“Las personas del proyecto de entrega son involucradas muy tarde en el proceso de desarrollo y así el producto no es bien visto desde una perspectiva comercial. Consecuentemente, una acción para la mejora sería integrar a la gente del proyecto de entrega desde la fase de requerimientos.”

Finalmente el autor concluye:

“...El estudio muestra que los problemas más comúnmente percibidos en los modelos de desarrollo pueden ser encontrados en el desarrollo dirigido por el plan, e introducir prácticas ágiles e incrementales permite mejorar sobre los problemas más comunes. Para responder las preguntas de investigación podemos concluir.

- Inconvenientes, varios problemas fueron identificados para ambos el enfoque dirigido por el plan y el uso de prácticas ágiles e incrementales, de cualquier forma los problemas más comúnmente percibidos entre los roles fueron del enfoque dirigido por el plan.
- Inconvenientes generales, Los dos más comúnmente percibidos fueron identificados para el enfoque dirigido por el plan: 1) cambio de requerimientos y re trabajo; y 2) reducción del cubrimiento del testing debido a tiempo limitado de testing al final del proyecto.