



CIMAT

Centro de Investigación en Matemáticas, A.C.

---

# MÉTODO DE EVALUACIÓN Y EVOLUCIÓN EN EL USO DE MÉTODOS DE DESARROLLO DE SOFTWARE ÁGIL

**T E S I S**

Que para obtener el grado de

**Maestro en Ciencias**

con Orientación en  
**Ingeniería de Software**

**Presenta**

Brisia Nectali Corona Tirado

**Director de Tesis:**

Dra. Mirna Ariadna Muñoz Mata

---

**Autorización de la versión final**



**CENTRO DE INVESTIGACIÓN EN MATEMÁTICAS, A.C.**

**BIBLIOTECA**

**AUTORIZACION**  
**PUBLICACION EN FORMATO ELECTRONICO DE TESIS**

El que suscribe

Autor(s) de la tesis: Brisia Nectali Corona tirado  
Institución y Lugar: Centro de Investigación en Matemáticas, A. C. Sede Zacatecas, Zac.  
Grado Académico: Licenciatura ( ) Maestría ( X ) Doctorado ( ) Otro ( ) -----  
Año de presentación: 2021-----  
Área de Especialidad: Ingeniería de Software-----  
Director(es) de Tesis: Dra. Mirna Ariadna Muñoz Mata-----  
Correo electrónico: brisiacorona@gmail.com-----  
Domicilio: Celaya, GTO-----  
Palabra(s) Clave(s): Métodos ágiles, herramientas de evaluación de entorno ágil, PYMEs -----  
ORCID: 0000-0003-4077-1179 -----

Por medio del presente documento autorizo en forma gratuita a que la Tesis arriba citada sea divulgada y reproducida para publicarla mediante almacenamiento electrónico que permita acceso al público a leerla y conocerla visualmente, así como a comunicarla públicamente en la Página WEB del Repositorio Institucional y Repositorio Nacional.

La vigencia de la presente autorización es por tiempo indefinido a partir de la firma de presente instrumento, quedando en el entendido de que si por alguna razón el alumno desea revocar la autorización tendrá que hacerlo por escrito con acuse de recibo de parte de alguna autoridad del CIMAT

Atentamente

\_\_\_\_\_  
Brisia Nectali Corona Tirado

# Resumen

Hoy en día una cantidad considerable de microempresas (MiPYMEs), así como pequeñas y medianas empresas (PYMEs) prefieren el uso de métodos ágiles. Sin embargo, la falta de conocimiento en cómo utilizar estos métodos, da como resultado que éstos se adopten de manera empírica, por lo tanto, sin implementarlas de manera adecuada.

En este trabajo se presenta un método de evaluación en relación a la implementación de los métodos ágiles, en específico de Scrum. El método de evaluación dirigido a las MiPYMEs y PYMEs, está basado en un análisis que mide el nivel de madurez en el cual una organización adopta y usa Scrum de manera adecuada. Para la implementación del método de evaluación, se ha desarrollado una herramienta que permita evaluar a las organizaciones, ofreciendo además información requerida para implementar Scrum e incrementar el nivel de madurez de la empresa en el uso de Scrum para su proceso de desarrollo de software.

Palabras Clave – métodos ágiles, herramientas de evaluación de entorno ágil, PYMEs.

# Abstract

Nowadays, small and medium enterprises are using agile methods in an effort to produce software to meet the time requested by the market. However, the lack of knowledge on how to use these methods results in an empirical adoption of them, without proper implementation and use of them.

This work presents an evaluation method for the implementation of agile methods, specifically Scrum. The method is addressed to small settings, as well as small and medium-sized enterprises (SMEs). It is based on an analysis that measures the maturity level in which an organization adopts and uses Scrum in an adequate way. Besides, for the method implementation, a tool has been developed that allows evaluating organizations, offering the required information to implement Scrum, and increase the organization's maturity regarding the use of Scrum in the software development process.

Keywords: Agile assessing tools, agile methods, SMEs, agile methods implementation and use.

# Agradecimientos

Agradezco a mi madre Ma. Lourdes Tirado Mendoza por toda su ayuda y apoyo incondicional, por impulsarme a alcanzar todo lo que deseo.

Agradezco a mis hermanas Cyndi e Ivonne porque, aunque no siempre tuve la oportunidad de estar para apoyarles saben que siempre estaremos juntas y que cuentan con mi apoyo infinito.

A mi familia por tanto apoyo, tanto amor. Los quiero.

Agradezco infinitamente a Mayolo, por acompañarme durante esta etapa, por impulsarme y enseñarme que si puedo. Gracias por todo.

Agradezco a todos y cada uno de los doctores y profesores, que me instruyeron en mi educación durante mi periodo en CIMAT, gracias por compartir y por impulsarme. Especialmente a la Dra. Mirna Muñoz por creer en mí a pesar de las circunstancias y apoyarme hasta el final, muchas gracias por su paciencia y por compartir conmigo su amplio conocimiento en este proyecto.

Agradezco a mis compañeros por el apoyo, por impulsarnos a mejorar, por el compañerismo gracias a la vida, por permitirme coincidir con tan maravillosas personas.

# Índice

<b>Resumen</b> .....	2
<b>Abstract</b> .....	3
<b>Agradecimientos</b> .....	4
<b>Introducción</b> .....	12
<b>Capítulo 1. Antecedentes</b> .....	13
1.1 Marco teórico.....	13
1.1.1 Métodos.....	14
1.1.2 Modelos del Ciclo de Vida.....	14
1.1.2.1 Modelos tradicionales de desarrollo de software.....	15
1.1.3 Estándares.....	19
1.1.3.1 Estándares de calidad de software.....	21
1.1.4 Madurez en el desarrollo de software.....	22
1.1.4.1 Modelos de Evaluación y Mejora de Procesos.....	23
1.1.4.2 CMMI Dev v1.3.....	24
1.1.4.3 ISO 15504: Software Process Improvement and Capability determination (SPICE).....	27
1.1.5 Métodos ágiles.....	29
1.1.5.1 Análisis de Métodos ágiles.....	32
1.1.5.2 SCRUM.....	33
1.1.5.3 XP (eXtreme Programamming).....	36
1.1.6 Comparación de metodologías tradicionales y métodos ágiles.....	38
1.1.7 Scrum checklist.....	40
1.1.8 Modelo de Madurez Ágil (AMM).....	40
1.1.9 Planteamiento del problema.....	41
1.1.10 Hipótesis.....	43
1.1.11 Objetivos de la investigación.....	43
1.1.11.1 Objetivo general.....	43
1.1.11.2 Objetivos específicos.....	43
1.1.12 Justificación.....	44
<b>Capítulo 2. Estado del arte</b> .....	45

2.1 Revisión sistemática de la literatura .....	45
2.1.1 Planificación .....	46
2.1.1.1 Identificación de la necesidad de la RSL.....	46
2.1.1.2 Preguntas de la investigación.....	47
2.1.1.3 Fuentes de datos .....	47
2.1.1.4 Cadenas de búsqueda .....	47
2.1.1.5 Criterios de selección .....	48
2.1.2 Ejecución .....	48
2.1.2.1 Estrategia de selección de estudios primarios.....	48
2.1.2.2 Extracción de datos .....	50
2.1.3 Resultados .....	50
2.1.3.1 Modelos, metodologías o estándares utilizados por las PYMES.....	51
2.1.3.2 Marcos de trabajo, métodos y metodologías para la evaluación de métodos ágiles...	51
2.2 Encuesta .....	52
2.2.1 Desarrollo de la Encuesta.....	53
2.3 Análisis de Herramientas.....	56
2.3.1 Procedimiento de Análisis .....	56
2.3.2 Tipos de análisis .....	56
2.3.3 Establecimiento de Criterios .....	58
2.3.4 Análisis de resultados.....	59
2.3.4.1 Tipos de Herramientas .....	59
2.3.5 Aspectos Enfocados.....	59
2.3.5.1 Herramienta tipo checklist.....	60
2.3.5.2 Herramienta de software tipo Encuestas.....	62
2.3.5.3 Herramientas tipo Cuestionario .....	64
2.3.5.4 Herramienta de software tipo Prueba .....	65
2.3.5.5 HERRAMIENTAS DE TIPO: Kit de Herramientas de software .....	66
2.3.5.6 Mejores herramientas.....	68
2.3.5.7 Hallazgos .....	70
2.3.5.8 Necesidades .....	70
2.4 Trabajos Relacionados.....	71
2.4.1 Metodología para la evaluación de software ágil en enfoques de desarrollo.....	71
2.4.2 Un Framework para Evaluación de Métodos Ágiles .....	72

2.4.3 Scrum Checklist .....	72
2.4.4 Modelo de Madurez Ágil (AMM) .....	73
2.5 Análisis de tecnologías .....	74
2.5.1 Arquitectura en Capas.....	74
2.5.2 Arquitectura tres capas .....	75
2.5.3 Base de Datos.....	75
2.5.3.1 ORM (object-relational mapping) .....	76
2.5.3.2 Doctrine.....	76
2.5.4 LENGUAJES .....	76
2.5.4.1 Lenguaje HTML 5.....	76
2.5.4.2 Lenguaje PHP.....	78
2.5.4.3 Lenguaje ASP.NET.....	78
2.5.4.4 Justificación de la elección de tecnología .....	80
<b>Capítulo 3. Desarrollo de la propuesta del método .....</b>	<b>81</b>
3.1 Metodología para desarrollo de la tesis.....	81
3.2 Desarrollo del método .....	83
3.2.1 Introducción .....	83
3.2.2 Fundamentos del método.....	84
3.2.3 Fase de Evaluación .....	86
3.2.4 Fase de Evolución.....	87
3.3 Descripción detallada del método .....	88
3.3.1 Estructura para descripción general de los niveles del método .....	88
3.3.2 Elementos comunes .....	89
3.3.3 Nivel 1: Overview .....	90
3.3.4 Nivel 2: Planeación .....	92
3.3.5 Nivel 3: Comunicación.....	94
3.3.6 Nivel 4: Comportamiento.....	95
3.3.7 Nivel 5: Comportamiento Extendido.....	97
3.4 Desarrollo de la herramienta .....	98
3.4.1 Requisitos Funcionales.....	99
3.4.2 Diagramas de flujo: Interacción sistema-usuarios .....	99
3.4.3 Diseño y Arquitectura de la herramienta.....	102
3.4.4 Diseño de prototipo .....	104



3.4.4.1	Diseño de la Herramienta .....	104
3.4.4.2	Pantalla Inicio de Sesión.....	105
3.4.4.3	Pantalla registro de un nuevo usuario .....	106
3.4.4.4	Pantalla para los niveles.....	107
3.4.4.5	Vista de Resultados .....	108
3.4.4.6	Videos de recomendaciones .....	108
<b>Capítulo 4.</b>	<b>Desarrollo del estudio de caso .....</b>	<b>109</b>
4.1	Diseño del estudio de caso.....	109
4.1.1	Objetivo del estudio de caso .....	109
4.1.2	Conocimiento Actual .....	110
4.1.3	Marco conceptual .....	110
4.1.4	Objeto de Estudio.....	112
4.1.4.1	Construcción de la muestra.....	113
4.1.4.2	Preguntas de investigación .....	114
4.2	Preparar la recolección de datos.....	114
4.2.1	Objetivos de mejora .....	115
4.2.2	Encuesta de validación .....	115
4.3	Recolección de datos.....	116
4.3.1	Métricas.....	117
4.4	Análisis de datos.....	119
4.4.1	Descripción de las inferencias de diseño .....	119
4.4.2	Adoptar las inferencias.....	120
4.5	Resultado del análisis .....	121
4.5.1	Resultados del estudio del caso .....	125
<b>Capítulo 5.</b>	<b>Conclusiones.....</b>	<b>130</b>
5.1	Conclusiones.....	131
5.2	Trabajo Futuro.....	133
5.3	Logros .....	133
5.3.1	Publicaciones.....	134
5.3.2	Estancia .....	135
5.3.3	Presentación CIMPS .....	135
<b>Referencias</b>	<b>.....</b>	<b>136</b>

# Índice de Figuras

Figura 1. Marco teórico y subtemas.....	13
Figura 2. Modelo Cascada .....	16
Figura 3. Modelo en V .....	17
Figura 4. Modelo Iterativo .....	18
Figura 5. Modelo de desarrollo incremental .....	18
Figura 6. Modelo de desarrollo espiral .....	19
Figura 7. Porcentaje de Métodos ágiles 2019.....	33
Figura 8. Ciclo Scrum .....	34
Figura 9. Proceso XP.....	37
Figura 10. Practicas XP .....	38
Figura 11. Modelo de Madurez ágil .....	41
Figura 12. Actividades en el protocolo de revisión sistemática .....	46
Figura 13. Estrategia de selección de estudios .....	49
Figura 14. Cantidad de resultados durante la selección de estudios.....	50
Figura 15. Ubicación de los resultados de la encuesta.....	54
Figura 16. Tamaño de la empresa de desarrollo de software .....	55
Figura 17. Métodos utilizados .....	55
Figura 18. Clasificación de herramientas por tipo.....	59
Figura 19. Modelo en capas .....	75
Figura 20. Metodología utilizada en la investigación.....	81
Figura 21. Habilidades que se identifican .....	84
Figura 22. Esquema de la metodología propuesta .....	85
Figura 23. Evaluación del uso de métodos ágiles .....	87
Figura 24. Evolución en el uso de métodos ágiles .....	87
Figura 25. Interacción Administrador Sistema.....	100
Figura 26. Interacción Usuario-Sistema-Nivel .....	101
Figura 27. Arquitectura de la aplicación.....	102
Figura 28. Esquema de BD.....	103
Figura 29. Página principal.....	104
Figura 30. Pantalla inicio de sesión.....	106

Figura 31. Pantalla de registro de nuevo usuario .....	107
Figura 32. Pantalla de niveles.....	107
Figura 33. Pantalla gráfica de resultados.....	108
Figura 34. Pantalla recomendaciones.....	108
Figura 35. Pasos del proceso de un estudio de caso .....	109
Figura 36. Aspecto evaluado en la herramienta.....	111
Figura 37. Total, de conocimiento por nivel .....	117
Figura 38. Promedio de intentos por nivel .....	118
Figura 39. Total, obtenido por niveles .....	119
Figura 40. Ubicación de los participantes .....	121
Figura 41. Resultados de la pregunta 1.....	122
Figura 42. Resultados de la pregunta 2.....	123
Figura 43. Resultados de la pregunta 3.....	124
Figura 44. Total, de respuestas de participantes de la muestra del nivel 1.....	126
Figura 45. Total, de respuestas de individuos de la muestra del nivel 2.....	127
Figura 46. Total, de respuestas de individuos de la muestra para el nivel 3.....	128
Figura 47. Total, de respuestas de individuos de la muestra para el nivel 4.....	129
Figura 48. Total, de respuestas de individuos de la muestra para el nivel 5.....	130

# Índice de Tablas

Tabla 1. Estructura del modelo CMMI Dev v1.3 .....	26
Tabla 2. Modelo de capacidad de proceso.....	28
Tabla 3. Métodos ágiles existentes .....	30
Tabla 4. Comparación de métodos ágiles y tradicionales .....	39
Tabla 5. Criterios de selección.....	48
Tabla 6. Clasificación de artículos.....	51
Tabla 7. Tipo de propuesta desarrollada para evaluar los métodos ágiles.....	52
Tabla 8. Encuesta para conocer el estado de las PYMEs de México.....	53
Tabla 9. Criterios establecidos.....	58
Tabla 10. Herramientas de tipo checklist.....	60
Tabla 11. Herramienta tipo encuesta.....	62
Tabla 12. Herramienta tipo cuestionario.....	64
Tabla 13. Herramienta de tipo prueba.....	65
Tabla 14. Herramienta de tipo software.....	67
Tabla 15. Mejores herramientas.....	69
Tabla 16. Trabajos Relacionados.....	73
Tabla 17. Características HTML5 .....	77
Tabla 18. Características PHP .....	78
Tabla 19. Características ASP .NET .....	79
Tabla 20. Comparación de Tecnologías.....	80
Tabla 21. Elementos para el diagnóstico del Nivel 1.....	91
Tabla 22. Elementos para el diagnóstico del Nivel 2.....	93
Tabla 23. Elementos para el diagnóstico del Nivel 3.....	94
Tabla 24. Elementos para el diagnóstico del Nivel 4.....	96
Tabla 25. Elementos para el diagnóstico del Nivel 5.....	97
Tabla 26. Tipos de usuario.....	99
Tabla 27. Actividades de usuario.....	105
Tabla 28. Empresas piloto.....	112
Tabla 29. Encuesta para la recolección de información.....	116
Tabla 30. Observaciones de individuos de la muestra.....	124

# Introducción

En los últimos años el crecimiento del desarrollo de software ofrece la oportunidad de producir productos y servicios a bajos costos, lo cual es atractivo para las PYMEs, suponiendo cerca del 99% de los negocios en Latinoamérica [1-2]. Estas organizaciones se enfrentan a una frecuente necesidad de mejorar su proceso de desarrollo de software. Y, por lo tanto, la calidad de sus productos, en una búsqueda por mantenerse en el mercado y lograr un crecimiento constante.

De acuerdo al “Chaos Manifiesto 2019” [3], informe elaborado por el “Chaos Report” [4], se menciona que, analizando los resultados del rendimiento de los proyectos, el porcentaje de proyectos que utilizan métodos ágiles obtuvieron un 42% de término exitoso, 8% de fallo y 50% de alto grado de dificultad, en contraste, con los proyectos que aplicaron metodologías tradicionales, que obtuvieron un 26% de termino exitoso, 21% de fallo y 53% de alto grado de dificultad.

En este contexto, las PYMEs han recurrido al uso de los métodos ágiles como una solución a los problemas del rendimiento de las metodologías tradicionales. Sin embargo, aún cuando existen principios para el desarrollo de software con métodos ágiles, que son una guía en su implementación, establecidos en el manifiesto ágil [5], las organizaciones que han implementado métodos ágiles, no lo han hecho de manera adecuada, debido a que basan su implementación en los beneficios que se presentan en otras organizaciones [6-7], provocando que se desconozca su correcta implementación, esto se ve reflejado en productos software de baja calidad. *La presente investigación tiene como objetivo establecer el estado del arte enfocado en: los métodos para evaluar el uso e implementación de los métodos ágiles que usan en las PYMEs en Latinoamérica.*

# Capítulo 1. Antecedentes

En este capítulo se mencionan los conceptos relacionados con el trabajo de tesis, como son los métodos ágiles y formales, madurez en el desarrollo de software, además de la definición del problema y los objetivos propuestos para ser alcanzados en la investigación.

## 1.1 Marco teórico

En el siguiente apartado se muestra la información acerca de los conceptos clave para una mejor comprensión de la investigación. A continuación, se muestran en la estructura del capítulo como: modelo tradicional de desarrollo de software, niveles de madurez, concepto de método ágil, manifiesto ágil, comparación de métodos ágiles y tradicionales, métodos de evaluación de procesos de software.

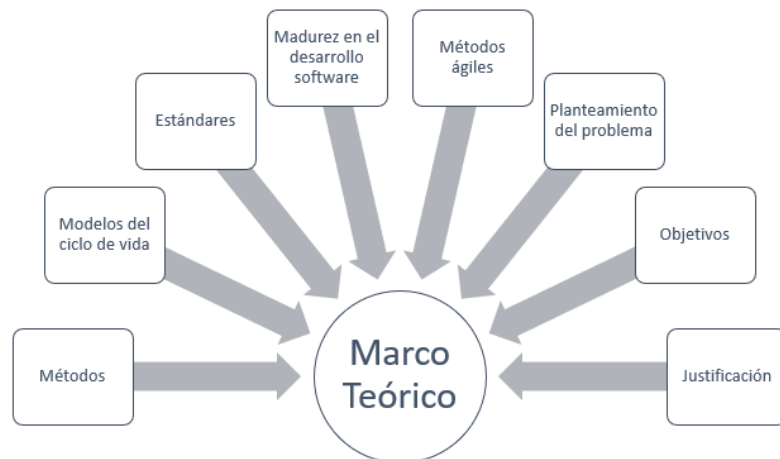


Figura 1. Marco teórico y subtemas.

### 1.1.1 Métodos

Un método, impone un proceso disciplinado sobre el desarrollo de software con el objetivo de hacer el desarrollo de software más predecible y eficiente. Por tanto, se plantea que un método define un camino reproducible para obtener resultados confiables. Un método de desarrollo de software describe como modelar y construir un sistema de software de una forma confiable y reproducible (M. Cossío et al. 2012 & Rina M Aguilera, 2013).

Los métodos existentes del proceso de software son (M. Cossio et al. 2012 & Rina M Aguilera, 2013):

- El modelo en cascada.
- Desarrollo evolutivo.
- Ingeniería del software basada en componentes.

### 1.1.2 Modelos del Ciclo de Vida

Un modelo de ciclo de vida de software, es una lista que ayuda a determinar el orden de las actividades y criterios entre las etapas del desarrollo de software (S. INTECO, 2009 & Tomohiro H. et al, 2011).

Entre las funciones de un ciclo de desarrollo de software se puede destacar (Ramírez, 2018):

- Determinar el orden de las fases del proceso.
- Establecer los criterios de transición entre fases.
- Definir entradas y salidas para cada fase.
- Describir los estados por los que el producto pasa.
- Describir las actividades a realizar para transformación del producto.

- Definir un esquema base para la planificación, organización, coordinación y desarrollo.

Un ciclo de vida de un proyecto se compone de un conjunto de fases sucesivas que se componen de tareas que se pueden planificar y cuentan con actividades genéricas. (Ramírez, 2018).

- Especificación: un ciclo de vida se compone de un conjunto de fases sucesivas con tareas que se planifican.
- Desarrollo: producción de software.
- Validación: comprobación de los requerimientos del sistema.
- Evolución: adaptar el software de acuerdo a los cambios.

Un modelo de ciclo de vida de software, tiene como características (S. INTECO, 2009):

- Descripción de las etapas principales del desarrollo de software.
- Establecer las etapas principales durante la ejecución.
- Apoyar a gestionar el progreso del desarrollo de software.
- Se cuenta con un espacio de trabajo.

En cada una de las etapas, se pueden establecer un conjunto de objetivos, tareas y actividades que lo caractericen. Existen distintos modelos de ciclo de vida, y su elección para un determinado tipo de proyecto es realmente importante (S. INTECO, 2009 & M. del Carme Gómez, 2011).

### 1.1.2.1 Modelos tradicionales de desarrollo de software

Los modelos tradicionales se basan en el ciclo de vida del desarrollo de software, de los cuales surgen modelos como: el modelo “cascada”, modelo en V y como el propio el ciclo de vida de software, entre otros. Estos modelos surgen por la necesidad de administrar proyectos de desarrollo de software, ya que, hasta entonces todo desarrollo se realizaba de manera artesanal, estos modelos surgen como procesos robustos que ayudan a mejorar el desarrollo, haciéndolo eficientes y predecibles (S. INTECO, 2009 & M. del Carmen Gómez, 2011).



### A) Modelo Cascada

Modelo que implementa desarrollos con un proceso secuencial, en el que el desarrollo de software se divide en fases separadas una tras otra, las cuales componen el ciclo de vida. Modelo reconocido como el primer modelo de proceso y adoptado ampliamente en ingeniería del software. En la Figura 2 se muestra el modelo cascada (S. INTECO, 2009 & Tomohiro H. et al, 2011).

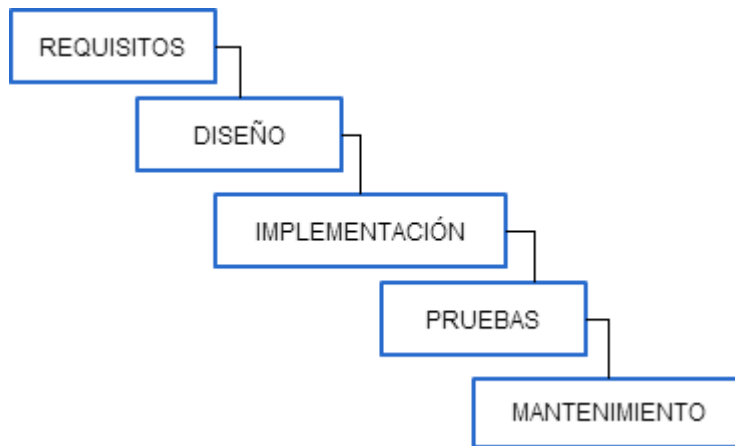


Figura 2. Modelo Cascada.

Modelo que sostiene una perspectiva metodológica de las etapas del ciclo de vida del desarrollo de software, debiendo ser por cada inicio de las etapas, siendo que no deberá comenzar la siguiente hasta concluir la anterior, de esta forma se visualiza de manera secuencial (S. INTECO, 2009 & Tomohiro H. et al, 2011).

### B) Modelo en V

Modelo que representa pasos consecutivos en el desarrollo del ciclo de vida de un proyecto de software. Donde se especifican las actividades y los resultados son utilizados durante el desarrollo. La parte izquierda de la V, se enfoca en el análisis de los requisitos y especificaciones del sistema. La parte derecha de la V, se enfoca en la unión de las partes y su verificación, y finalmente la V, es

“Validación y Verificación”. La Figura 3 muestra el modelo en V (S. INTECO, 2009 & M.del Carmen Gómez, 2011).

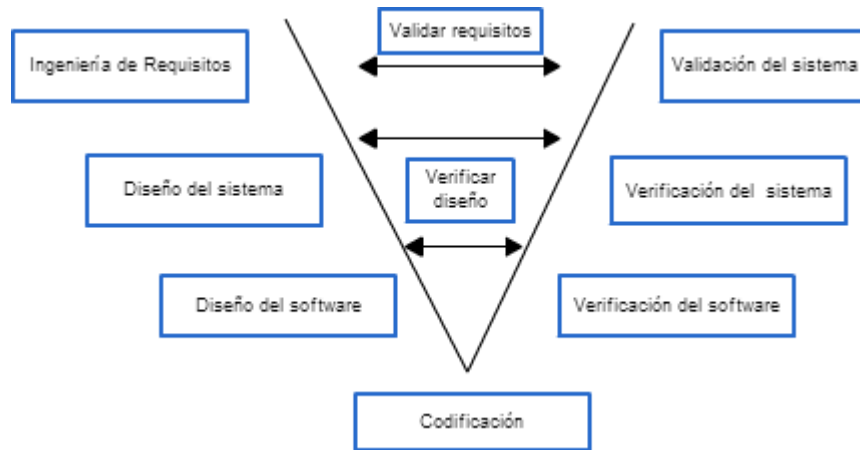


Figura 3. Modelo en V.

### C) Modelo iterativo

Modelo derivado del ciclo de vida en cascada. Cada iteración lleva a cabo un ciclo de vida en cascada. Al final de cada iteración se le entrega al cliente una versión mejorada del producto o con mayor funcionalidad. El cliente es quien después de cada iteración evalúa el producto y lo corrige o propone mejoras. Estas iteraciones se repiten hasta obtener un producto que satisfaga las necesidades del cliente. En la Figura 4 se muestra el modelo iterativo (S. INTECO, 2009 & M.del Carmen Gómez, 2011).

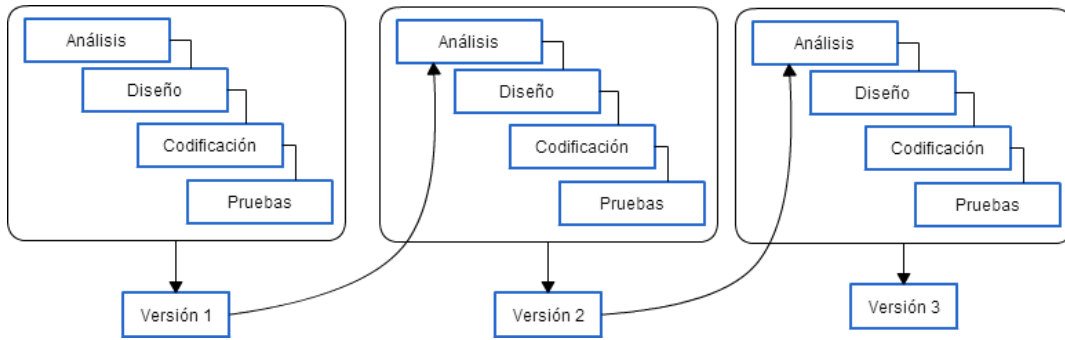


Figura 4. Modelo Iterativo.

*D) Modelo de desarrollo incremental*

El modelo incremental combina elementos del modelo en cascada y filosofía interactiva de construcción de prototipos. Este modelo construye para incrementar las funcionalidades del programa. Este modelo aplica una secuencia lineal escalonada mientras se va progresando en el tiempo del calendario. En cada secuencia lineal, se produce un incremento del software. En la Figura 5 se muestra el modelo de desarrollo incremental (S. INTECO, 2009 & M.del Carmen Gómez, 2011).

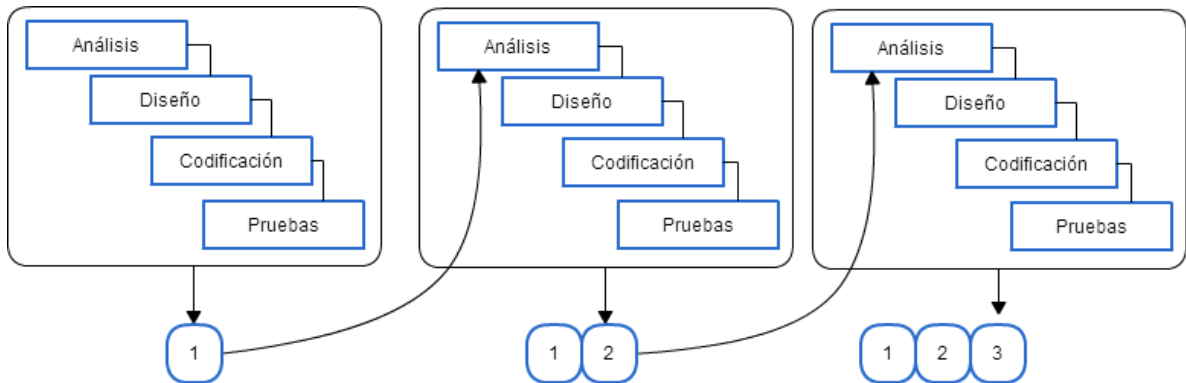


Figura 5. Modelo de desarrollo incremental.

### E) Modelo Espiral.

Es un modelo de ciclo de vida desarrollado por Barry Boehm en 1985, utilizado de forma general para la Ingeniería del Software. Las actividades se conforman en forma de espiral, donde cada bucle representa un conjunto de actividades. Las actividades no están fijadas a priori (S. INTECO, 2009 & M.del Carmen Gómez, 2011). Este modelo combina las características del modelo de prototipos y el modelo en cascada. Este modelo está pensado para proyectos largos, caros y complicados. Este modelo fue el primero en explicar el concepto “iteraciones” (S. INTECO, 2009 & M.del Carmen Gómez, 2011). En la Figura 6 se muestra el modelo en espiral.

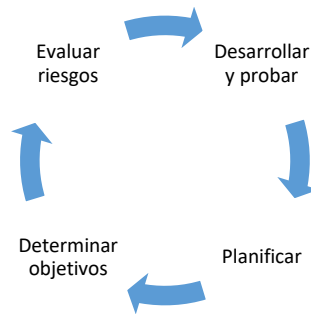


Figura 6. Modelo de desarrollo espiral.

### 1.1.3 Estándares

El crecimiento del comercio y de las sociedades, ha impulsado a las organizaciones a buscar mejorar continuamente sus productos, servicios y bienes debido a la demanda. Es por eso que la mejora de sus procesos ha tomado parte primordial para la optimización de sus recursos. A partir de este planteamiento, se hace necesaria la creación de una organización dedicada a uniformizar el proceso de las organizaciones, así como para la creación de estándares o normas de calidad aplicables a todas las organizaciones productivas (I. ICDE, 2015 & Moreno, 2012).

Los estándares que contienen lineamientos o criterios precisos constituidos como reglas, guías o definiciones de características, documentados de común acuerdo, para asegurar que materiales,

productos, procesos y servicios cumplen su propósito con interoperabilidad y calidad (I. ICDE, 2015 & Moreno, 2012).

La estandarización persigue fundamentalmente tres objetivos (Moreno, 2012 & ISO, 2004):

- Simplificación: seleccionar únicamente los modelos necesarios.
- Unificación: permite la integración de un nivel determinado.
- Especificación: identificar un lenguaje claro y preciso, para aminorar errores en la identificación.

Los estándares brindan elementos de calidad, seguridad, fiabilidad y eficiencia a los productos. Si se cumplen los lineamientos, los elementos obtenidos cumplen los objetivos para cumplir con necesidades. Esto brinda la oportunidad a las organizaciones para incursionar en mercados nacionales o internacionales para ofrecer sus productos, servicios y tecnología, provocando una competencia desarrollada en igualdad de condiciones, si los requerimientos de ingreso son basados en el cumplimiento de estándares (Moreno, 2012 & ISO, 2004).

La ISO, como parte de sus funciones es elaborar proyectos que apliquen estándares técnicos internacionales; así como la coordinación y la cooperación entre países unificando criterios; elaboración de las normas internacionales; y colaboración activa con organizaciones internacionales que impulsan la normalización (I. ICDE, 2015).

Estandarización es el proceso de formulación, elaboración, aplicación y mejora de estándares existentes. Hoy en día, forman una parte muy importante en los procesos de las organizaciones (I. ICDE, 2015).

### 1.1.3.1 Estándares de calidad de software

Uno de los principales propósitos de un estándar es promover un intercambio de productos con base a ciertos lineamientos comunes para la existencia de una industria de software robusta. De acuerdo a (I. ICDE, 2015 & ISO, 2004-2008) deben de establecerse la definición de características y reglas que serán utilizados constantemente por la organización. La Organización Internacional para la Estandarización (ISO, por sus siglas en inglés) fue creada en 1947 con la finalidad de hacer más fácil la coordinación internacional y la unificación de estándares industriales. (Moreno, 2012 & ISO, 2004-2008). Como resultado de la estandarización de los procesos de software surge la ISO 9001:1994, la ISO/IEC 12207.

- A continuación, se listan los estándares más utilizados: ISO 9001:1994: La Organización Internacional para la Estandarización (ISO) (Moreno, 2012 & ISO, 2004) creó un apartado de calidad del software y fue la primera organización encargada de la creación de un estándar enfocado en la creación de software. ISO 9001:1994, fue creado como respuesta al Capability Maturity Model (CMM) por parte de ISO. Ésta se encarga de los “Sistemas de calidad”. Modelo para asegurar la calidad en diseño o desarrollo, producción, instalación y servicio”. Tres años después se publica al mercado el ISO 9000-3:1997, que es una expansión de una guía para el ISO 9001:1994 (Moreno, 2012 & ISO, 1994).

- **ISO/IEC 12207:** La Organización Internacional para la Estandarización (ISO) comienza a trabajar con la Comisión Electrónica Internacional (IEC, por sus siglas en inglés) para generar una nueva norma que reuniera los estándares relacionados a la electrónica y a los estándares de proceso de software. Norma que dicta lineamientos para los procesos del ciclo de vida del software, determinando una terminología bien definida que permitan citar correctamente en la industria del software. Ésta describe los procesos para adquirir, desarrollar, operar y controlar software, así como sus características (Moreno, 2012 & ISO, 2008).

#### 1.1.4 Madurez en el desarrollo de software

En 1984, el departamento de defensa de Estados Unidos de América, formó el instituto de ingeniería de software (SEI por sus siglas en inglés) solicitando la creación del mismo a la universidad Carnegie Mellon. El objetivo del SEI es el establecimiento de estándares de excelencia en la Ingeniería de Software para acelerar la transición de la tecnología y métodos en las prácticas (W. Humphrey, 2012).

Watts S.Humphrey, en su artículo “un framework para caracterizar el proceso del software” (W. Humphrey, 2012), define un marco de trabajo de cinco niveles para la caracterización del estado actual del proceso de software, donde estos niveles representan las fases de mejora evolutiva de las organizaciones reales, a través de objetivos establecidos dentro de la organización. Los cinco niveles definidos se describen a continuación (W. Humphrey & M. Llaneza et al., 2012,2013):

- *Inicial.* Se debe mantener un control estadístico sobre el proceso. Mientras esto no suceda la mejora en el proceso no sucederá.

- *Repetible*. Se ha alcanzado un proceso estable en la implementación del proceso de control estadístico. La gestión de proyectos da un notable cambio en la gestión de costos, compromisos, cronograma y cambios del proyecto de software.
- *Definido*. Una vez estable la gestión de proyectos la organización establece, define e informa el proceso. En este nivel la adaptación a nuevas tecnologías se da adecuadamente.
- *Administrado*. La organización mantiene las mediciones de todos los procesos de manera estable. Se da las mejoras más significativas en el desarrollo y aplicación del proceso en el desarrollo de software.
- *Optimización*. La organización puede adaptar y optimizar el proceso definido.
- controlar software, así como sus características (Moreno, 2012 & ISO, 2008).

#### 1.1.4.1 Modelos de Evaluación y Mejora de Procesos

Desde los inicios de la Ingeniería de Software, ha existido una gran preocupación enfocada en que los artefactos alcancen un nivel de calidad óptimo dentro del límite de tiempo y coste marcados en el proyecto. Por lo tanto, la estandarización de los modelos de evaluación y mejora de procesos ha adquirido un papel decisivo para identificar, integrar, medir y optimizar buenas prácticas en las organizaciones desarrollo software (M. De La Vila et al., 2004 & S. Soundararajan, 2011).

Evaluar un proceso se refiere a examinar determinadamente los procesos a través de un conjunto de criterios que apoya a determinar su capacidad dentro de objetivos de calidad, planificación y costo. El objetivo es identificar las practicas actuales, permitiendo identificar debilidades y fortalezas del



proceso para controlar las incidencias que provocan baja calidad, así como las desviaciones en el costo o en la planificación (M. De La Vila et al., 2004 & S. Soundararajan, 2011).

El modelo de evaluación de procesos de software ISO 15504, es una adaptación con la norma ISO/IEC 20000 y ISO/IEC 2700, así como se encuentra alineado con los métodos ágiles (SCRUM, XP, etc.) (ISO, 2020).

- **ISO 15504:** norma que permite evaluar el ciclo de vida del software y los procesos relacionados con el desarrollo de los servicios TIC. Esta norma es importante para las empresas de desarrollo de sistemas informáticos que quieren posicionarse en el mercado (Riesco, 2018).

- **ISO 20000:** norma de calidad enfocada en los Servicios de Tecnologías de la Información, que se encarga de asegurar la implementación de buenas prácticas para la seguridad de la información, así como para evitar riesgos y mejorar procesos de información. (Riesco, 2018).

- **ISO 27001:** norma enfocada en la Seguridad de la Información, para garantizar las buenas prácticas en seguridad de información. Esta norma establece los requisitos para implantar con éxito un Sistema de Gestión de Seguridad de la Información o SGSI. (Riesco, 2018).

#### 1.1.4.2 CMMI Dev v1.3

El modelo de madurez y de capacidad (Capability Maturity Model Integration CMMI), ha evolucionado hasta su actual versión 2.0 lanzada en abril del 2018, sin embargo, en la investigación se abarca únicamente la versión 1.3, que a continuación se describe:

El modelo CMMI Dev v1.3, marco de referencia que permite medir el desempeño de los diferentes procesos en las organizaciones de software, por medio de fundamentos para generar una evaluación que permita determinar la madurez de sus procesos; concediendo una guía para aplicar estrategias de mejora continua.

CMMI Dev v1.3 se enfoca en la mejora de procesos en una organización, analizando los procesos de desarrollo y permitiendo una evaluación de la madurez de la organización con base en una escala de cinco niveles (inicial, repetible, definido, dirigido y optimizado) (M. De La Vila et al., 2004 & SEI, 2010).

- **Estructura del modelo.**

El modelo CMMI Dev v1.3 está compuesto por niveles de madurez que contienen áreas de proceso, metas, prácticas y subprácticas. Cada nivel de madurez contiene un conjunto de áreas de proceso que apoyan a una organización a enfocar la mejora de sus procesos. Cada área de proceso se describe a nivel de prácticas que contribuyen a satisfacer un conjunto de objetivos (M. De La Vila et al., 2004 & SEI, 2010).

A continuación, la Tabla 1 se presentan las áreas de proceso cubiertas por el modelo CMMI Dev v1.3.

Tabla 1. Estructura del modelo CMMI Dev v1.3.

N de Madurez de la Organización	Centrado en	Áreas del proceso	Categoría
5. Optimizado	Mejora continua del proceso	<ul style="list-style-type: none"> <li>• Análisis y resolución de causas de desviaciones.</li> <li>• Innovación y despliegue a toda la organización</li> </ul>	SopORTE
			G. Proceso
4. Gestionado cuantitativamente	Control cuantitativo del proceso	<ul style="list-style-type: none"> <li>• Gestión cuantitativa de los proyectos.</li> <li>• Entendimiento cuantitativo del rendimiento de los procesos de la organización.</li> </ul>	G. Proyecto
			G. Proceso
3. Definido	Proceso caracterizado por la organización y proactivo	<ul style="list-style-type: none"> <li>• Desarrollo de los requisitos</li> <li>• Soluciones técnicas</li> <li>• Integración de productos</li> <li>• Verificación</li> <li>• Validación</li> <li>• Enfoque de procesos en organización</li> <li>• Definición de procesos en organización.</li> <li>• Entrenamiento y formación</li> <li>• Gestión integrada de proyectos</li> <li>• Gestión del riesgo</li> <li>• Análisis y resolución de las decisiones</li> <li>• Entorno organizativo para la integración</li> <li>• Equipo para desarrollo integrado</li> </ul>	Ingeniería
			Ingeniería
			Ingeniería
			Ingeniería
			Ingeniería
			G. Proceso
			G. Proceso
			G. Proceso
			G. Proyecto
			G. Proyecto
			SopORTE
2. Gestionado	Gestión básica del proyecto	<ul style="list-style-type: none"> <li>• Gestión de requisitos</li> <li>• Planificación de proyectos</li> <li>• Monitorización y control de proyectos</li> <li>• Gestión de acuerdos con proveedores.</li> <li>• Medición y análisis</li> <li>• Aseguramiento de la calidad del producto y del proceso</li> <li>• Gestión de la configuración</li> </ul>	Ingeniería
			G. Proyecto
			G. Proyecto
			G. Proyecto
			SopORTE
			SopORTE
			SopORTE
			SopORTE
1. Inicial	Proc.impredecible, control reactivo		

La representación continua, toma una perspectiva en la capacidad de cada área del proceso de desarrollo de software y establecer lineamientos que permiten medir la mejora individual por área.

modelo que contiene prácticas definidas en las áreas del proceso las cuales permiten soportar el crecimiento en la mejora del proceso individual (M. De La Vila et al., 2004 & SEI, 2010).

### 1.1.4.3 ISO 15504: Software Process Improvement and Capability determination (SPICE)

ISO/IEC 15504 es un estándar internacional para evaluar y determinar la capacidad y mejora continua de procesos de Ingeniería del Software. Este estándar desarrolla un conjunto de medidas de capacidad estructuradas para todos los procesos del ciclo de vida y sus participantes (W. Humphrey & VersionOne, 2012, 2014 & ISO, 2008).

- **Estructura del estándar**

ISO/IEC desarrolla un modelo 2-D de evaluación de la capacidad del proceso, para valorar la organización de desarrollo software para la dimensión del proceso contra los atributos del proceso en la dimensión de capacidad. A partir de la versión (2004) se reducen las etapas del estándar a cinco partes: Cliente-Proveedor, Ingeniería, Proyecto, Soporte y Organización. (M. Llana et al., 2013 & ISO, 2008).

- Parte 1. Conceptos y Vocabulario. En publicación (7/10/04)
- Parte 2. Realizando una Evaluación (Requisitos, normativa). Publicado (30/10/04).
- Parte 3. Guía para Realización de Evaluaciones. Publicada (6/1/04)
- Parte 4. Guía para el Uso de Resultados de Evaluaciones. Publicada (6/7/04)
- Parte 5. Un Modelo de Evaluación de Procesos Ejemplar. Supera votación CD2

Con la idea de aplicar el estándar sin evitar restringirlo a un determinado ciclo de vida, la evolución del estándar para aceptar Modelos de Referencia de Procesos (PRM's) elimina la dimensión inicial

de procesos. La medida de capacidad se aplicable a cualquier modelo de procesos plasmado en un PRM compatible con ISO 12207(véase Tabla 2) (C. Ramírez, 2012 & ISO, 2008).

Tabla 2. Modelo de capacidad de proceso.

Id	Nivel de Capacidad	Atributos de Proceso y Descripción
CL [0]	Incompleto	El proceso no está implementado o falla en alcanzar su propósito. No es fácil identificar los productos o salidas de los procesos.
CL [1]	Realizado	El propósito del proceso se logra generalmente, aunque no sea rigurosamente planificado ni llevado a cabo. Hay productos identificables que testifican el alcance del propósito.
<b>PA.1.1 Realización del Proceso.</b>		
CL [2]	Gestionado	El proceso es gestionado y los resultados de los entregables de procedimientos específicos, planificados y seguidos, con requisitos de calidad, tiempo y recursos.
PA.2.1		Gestión de la Realización.
PA.2.2		Gestión de los Productos del trabajo
CL [3]	Establecido	Un proceso realizado y gestionado aplicado un proceso definido, basado en un principio de buenas prácticas de ingeniería del software.
PA.3.1		Definición del Proceso.
PA.3.2		Despliegue del Proceso.
CL [4]	Predecible	El proceso definido es puesto consistentemente en práctica dentro de límites de control establecidos para alcanzar metas del proceso ya definidas. Entendimiento cuantitativo de la capacidad del proceso y habilidad mejorada de predecir y gestionar el rendimiento.
PA.4.1		Medición del Proceso.
PA.4.2		Control del Proceso.
CL [5]	En optimización	Realización del proceso optimizada en la búsqueda de las necesidades actuales y futuras del negocio. Objetivos cuantitativos de eficiencia y efectividad se establecen en función de los objetivos de la organización. La optimización puede llevar a estudiar y adoptar ideas innovadoras o productos tecnológicos novedosos que incluyan y modifiquen el proceso definido.
PA.5.1		Innovación del Proceso.
PA.5.2		Optimización del proceso.

### 1.1.5 Métodos ágiles

De acuerdo a (S. INTECO, 2009), los métodos ágiles promueven un proceso de gestión de proyectos fomentando el trabajo en equipo, la organización y auto responsabilidad. Contienen un conjunto de mejores prácticas de ingeniería permitiendo una rápida entrega de software de alta calidad, teniendo enfoque de negocio alineado el desarrollo con las necesidades del cliente y los objetivos de la organización.

Los métodos ágiles se enfocan en el individuo y en el producto, es decir, al desarrollador y al cliente para generar un ambiente propicio para el desarrollo, al darse cuenta que si el ambiente es más flexible para el individuo los resultados son más eficientes. Para la mejor interpretación del concepto de agilidad se establecieron principios los cuales se encuentran en el manifiesto ágil (T. Hayata et al. & B. Kent et al., 2011). La tabla 3 muestra los métodos ágiles existentes (G. Mariana, 2014).

Tabla 3. Métodos ágiles existentes.

<b>Metodologías ágiles existentes</b>	
<b>Español</b>	<b>Ingles</b>
<b>Desarrollo de Software Adaptativo</b>	Adaptative Software Development
<b>Modelado Ágil</b>	Agile Modeling
<b>Desarrollo Dirigido por Modelo Ágil</b>	Agile Model Driven Development
<b>Gestión Ágil de Proyectos</b>	Agile Project Management
<b>Proceso Unificado Ágil</b>	Agile Unified Process
<b>Metodologías Crystal</b>	Crystal Methods
<b>Desarrollo Basado en Funcionalidades</b>	Feature Driven Development
<b>Desarrollo Rápido en Internet</b>	Internet Speed Development
<b>Metodología Lean</b>	Lean Development
<b>Programación Pragmática</b>	Pragmatic Programming
<b>Scrum</b>	Scrum
<b>XBreed</b>	XBreed
<b>Programación Extrema</b>	Extreme Programming
<b>Modelo Espiral</b>	Win Spiral
<b>Gestión de Proyectos Evolutivo</b>	Evolutionary Project Management
<b>Desarrollo Basado en Historias de Usuario</b>	Story Cards Driven Development
<b>Proceso Unificado Abierto</b>	Open Unified Process
<b>Kanban</b>	Kanban
<b>Incepción Ágil</b>	Agile Inception
<b>Lanzamiento de diseño (Google)</b>	Design Sprint (Google)

- **Manifiesto ágil**

Los principios del manifiesto ágil (T. Hayata et al.,2011), se basan en un cambio en la valoración de los siguientes aspectos (P. José et al. & B. Kent et al., 2011):

- **Valores:**

- La intercomunicación del equipo de desarrollo y las personas, por encima del proceso y las herramientas.

- El factor primordial para el éxito o fracaso en un proyecto de software, es el personal.
- La documentación no es tan importante como el desarrollo de software, lo primordial es generar software de desempeño efectivo; se genera documentación si es importante para la toma de decisiones, pero estos documentos deben de ser cortos para centrarse en lo fundamental.
- Agilidad en el desarrollo de software se busca la colaboración del cliente, la cual se centra en la interacción del equipo de desarrollo y el cliente. Esta relación, marcara el éxito o el fracaso del desarrollo de software.

Como resultado se espera una respuesta efectiva a los cambios en el desarrollo, para responder eficazmente se debe contar con la habilidad de responder a los problemas que se presenten durante el desarrollo, eso de ninguna manera afecta a lo planificado (B. Kent et al. & P. José et al., 2011).

- Principios

Los valores antes mencionados inspiraron los doce principios del manifiesto ágil (Think Big, A.S,2013-2016):

- I. La prioridad se enfoca en satisfacer al cliente a través de entregas tempranas y continuas de software que aporta valor al cliente.*
- II. Bienvenida a los cambios. Recibir cambios para dar una ventaja competitiva al cliente.*
- III. Entregar de manera frecuente software funcionando, con un intervalo menor de tiempo entre entregas (entre 2 semanas y hasta 2 meses).*
- IV. Impulsar trabajo en conjunto entre el cliente y los desarrolladores a lo largo del proyecto.*



- V. *Construir el proyecto en un ambiente de individuos motivados. Darles un ambiente de confianza, así como el apoyo y confianza necesarios para lograr la conclusión del trabajo.*
- VI. *Dialogo cara a cara es el método más eficiente y efectivo para la comunicación de información en un equipo de desarrollo.*
- VII. *El software funcionando es la medida principal del progreso del proyecto.*
- VIII. *Los procesos ágiles promueven un desarrollo sostenible.*
- IX. *Atender continuamente a la calidad técnica y al buen diseño para mejorar la agilidad.*
- X. *La simplicidad es fundamental.*
- XI. *Las mejores arquitecturas, requisitos y diseños provienen de los equipos auto organizados.*
- XII. *El equipo analiza cómo ser más efectivo en intervalos regulares para ajustar su comportamiento.*

Los principios antes listados rigen y ayudan a mejorar el ambiente entre el desarrollador y el cliente. Todo esto para alcanzar el desarrollo de software eficaz.

#### 1.1.5.1 Análisis de Métodos ágiles

En una encuesta del 2018 de (VersionOne, 2018), Esta encuesta muestra que los cinco métodos ágiles más utilizadas en las empresas y desarrolladores son: Scrum 66%, ScrumBan 6%, Scrum con XP Híbrido 6%, Kanban 6%, Iterativo 4%, Extreme programming (XP) 1%, Lean Startup 1%, other 5% y no se identifica 5%. La Figura 7 muestra el porcentaje de utilización de los métodos.

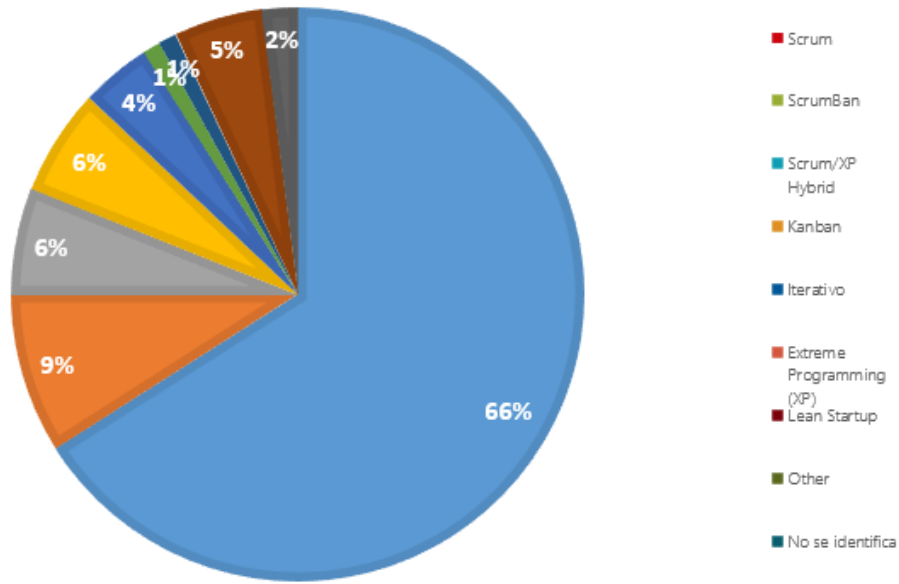


Figura 7. Porcentaje de Métodos ágiles 2018.

### 1.1.5.2 SCRUM

Método que permite establecer un proceso compuesto de un conjunto de buenas prácticas en el desarrollo de software basadas en la colaboración y el equipo, de manera que permita obtener el mejor resultado para un proyecto (Proyectos ágiles, 2012). Se realizó un análisis sobre los procedimientos que aplican los equipos altamente productivos, y como resultado se eligieron estas buenas prácticas. (J. Pérez, 2012 & B. M. Schwaber Ken, 2004). En la Figura 8 se muestra el ciclo Scrum.

Scrum es un método ágil en el que un proyecto se ejecuta mediante bloques temporales (llamadas iteraciones o sprint) que pueden tener una duración de un mes natural (sin embargo, pueden ser de dos o tres semanas, si así se requiere). En cada iteración se desarrolla un producto completo, que debe

ser un incremento del producto susceptible a ser entregado con el mínimo esfuerzo, cuando el cliente lo solicite (J. Pérez, 2012 & B. M. Schwaber Ken, 2004).

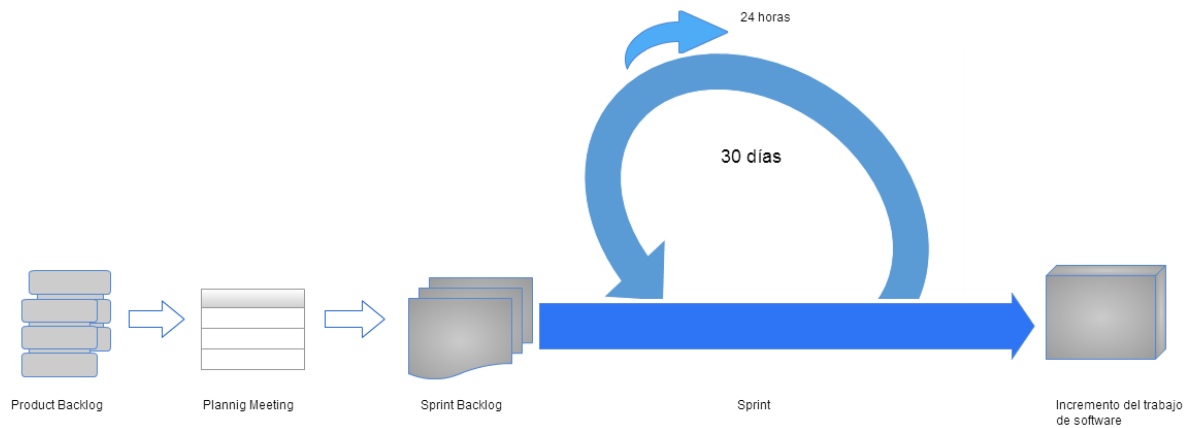


Figura 8. Ciclo Scrum.

El proceso scrum inicia con los requisitos enlistados y priorizados por el cliente, en esta lista se muestran segmentados en iteraciones y entregas esperadas, cumpliendo como parte del plan de proyecto.

- Actividades del método Scrum

El Sprint de los ciclos de Scrum se delimita por las reuniones de planificación del sprint y de retrospectiva. Cabe resaltar, que una vez fija la duración del Sprint, ésta es inamovible, un periodo de duración tradicional tiene una duración de entre dos a cuatro semanas. Durante el sprint, el equipo de desarrollo realiza diariamente una reunión de seguimiento muy breve. Además, al finalizar el sprint se entrega un incremento del producto al cliente con base en la funcionalidad planificada al inicio del sprint. A continuación, se detallan las actividades Scrum (J. Pérez, & J. Ruiz et al., 2011, 2012):

1. Planificación de la iteración. El primer día se lleva a cabo una reunión para la planificación de las siguientes iteraciones. La reunión está compuesta dos partes: 1)

elección de requisitos (4 horas máximo), 2) planificación de la iteración (4 horas máximo)

2. Ejecución de la iteración (sprint), periodo de tiempo para la realización de un producto funcionando al cliente.
3. Reuniones diarias (Scrum daily meeting), son reuniones del equipo donde se enfoca en el canal de comunicación y la participación de los integrantes de manera que se mejore la productividad. Estas reuniones para sincronización de las actividades realizadas (15 minutos máximo).
4. Demostración de requisitos terminados (Sprint Demonstration), es una reunión informal donde el equipo presenta al cliente los requisitos acordados y terminados durante la iteración.
5. Retrospectiva (Sprint Retrospective). Reunión en la que el equipo analiza cómo ha sido su manera de trabajar durante la iteración (J. Pérez, 2012).
6. Re planificación del proyecto. Durante el transcurso de cada iteración, el propietario del producto en conjunto con el cliente va trabajando la lista de requisitos priorizada (añadiendo requisitos, modificándolos, eliminándolos o re priorizándolos), pudiendo cambiar el contenido de las iteraciones y definiendo un calendario de entregas ajustado a las nuevas necesidades (J. Ruiz et al., 2011 & B. M. Schwaber Ken, 2004).

### 1.1.5.3 XP (eXtreme Programming)

Enfoque planteado por Kent Beck de ingeniería de software basado en la programación extrema (XP), Aplicar dinámicamente las mejores prácticas durante el ciclo de vida del software mientras se lleva un proyecto de software, se le considera como programación extrema (G. Mariana, 2014 & Mejhem, 2010).

Los planes en XP se diferencian de las metodologías tradicionales en tres aspectos (J. Pérez, 2012 & Mejhem, 2010):

- Simplicidad en el plan. Un plan que no requiera de un “gurú” con complicados sistemas para gestionar el proyecto.
- Los planes deben ser realizados por personas que realizarán el proyecto.
- Los planes debe ser la mejor estimación de cómo saldrán las cosas. Los planes son útiles, pero necesitan ser evolucionar cuando las circunstancias lo requieren.

El ciclo de desarrollo consiste (de manera resumida) en los siguientes pasos (G. Mariana, 2014 & Mejhem, 2010):

- El cliente define el valor de negocio a implementar.
- El programador estima el esfuerzo necesario para su implementación.
- El programador construye ese valor.
- Vuelve al paso inicial.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden. El ciclo de vida ideal de XP consiste en 6 fases: exploración, planificación de la entrega, iteraciones, producción, mantenimiento y conclusión del proyecto (M. Llana et al. & G. Mariana, 2013, 2014).

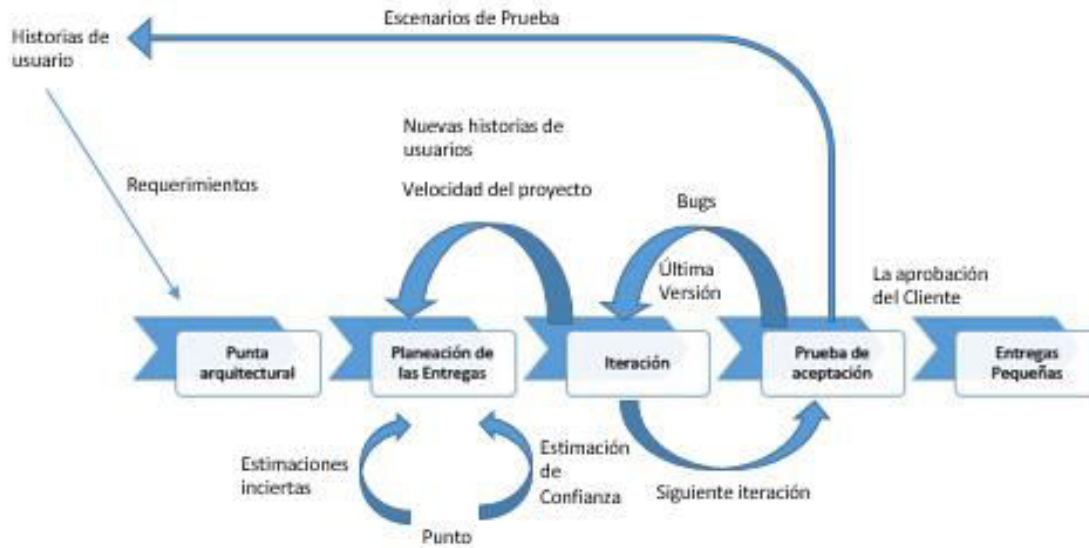


Figura 9. Proceso XP.

- Prácticas XP

XP permite la posibilidad de disminuir la curva exponencial del costo del cambio a través del proyecto para un diseño evolutivo, conseguido por las tecnologías disponibles para el desarrollo de software y a la aplicación disciplinada de las prácticas para (M. Llana et al. & S. INTECO, 2013, 2009): planificación, diseño, pruebas, refactoring, programación en pares, entregas pequeñas, integración continua, entre otras. Utilizar estas prácticas conjuntamente y equilibrada permitirá un mayor beneficio en su aplicación. La forma en cómo se relacionan las prácticas de XP se definen en la Figura 10.

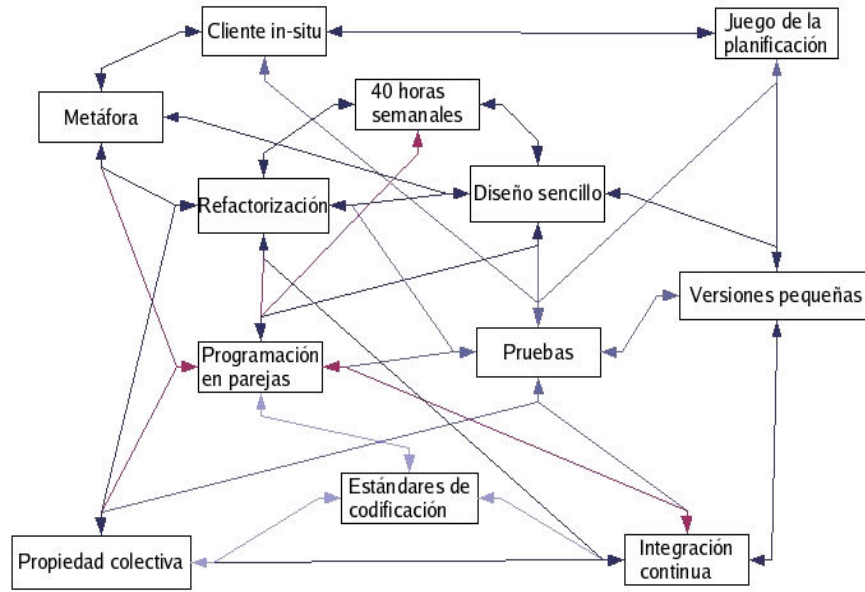


Figura 10. Practicas XP.

### 1.1.6 Comparación de metodologías tradicionales y métodos ágiles

En el desarrollo de software se requiere de un proceso definido para obtener resultados de calidad en los productos (D. Brenda et al., 2013 & S. Overhage et al, 2011). Hoy en día es de vital importancia mejorar los procesos de desarrollo, así como mejorar su proceso de gestión en los proyectos (D. Brenda et al., 2013 & S. Overhage et al, 2011). Los beneficios que obtiene una organización al implementar mejoras en los procesos de desarrollo de software en la calidad, el incremento del aumento en la cartera de clientes, por la pronta entrega del producto e impulso a la competitividad.

En Ingeniería de Software las metodologías para el desarrollo de software se dividen en dos grupos:

- Metodologías tradicionales (J. Ruiz et al. & S. INTECO, 2011, 2009), se enfocan en seguir un plan desde el inicio del proyecto hasta el final, además de mantener la documentación adecuada para el avance del proyecto.

- Métodos ágiles (J. Ruiz et al. & S. INTECO, 2011, 2009), se enfocan en los individuos del equipo y en la adaptabilidad de los procesos de desarrollo.

La diversidad de las metodologías permite aplicar una o varias metodologías a un proyecto para la obtención de mejores resultados. En este contexto los métodos ágiles pueden ser utilizados en conjunto, debido a que se rigen por procesos o pasos similares (J. Ruiz et al. & S. INTECO, 2011, 2009). Las principales diferencias de los métodos ágiles y las tradicionales basado en los siguientes estudios (J. Ruiz et al., W. Humphrey, P. José et al., 2012), en los cuales para realizar la comparación entre el método ágil y las tradicionales establecieron los siguientes atributos comunes en el desarrollo de software: normas, cambios, actividades impuestas, control, contrato, interacción del cliente, grupos de trabajo, artefactos, roles y arquitectura. Como se muestra en la Tabla 4.

Tabla 4. Comparación de metodologías tradicionales y métodos ágiles.

Actividades	Metodologías Ágiles	Metodologías Tradicionales
<b>Normas</b>	Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
<b>Cambios</b>	Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
<b>Actividades</b>	Impuestas internamente (por el equipo)	Impuestas externamente
<b>Control</b>	Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/normas
<b>Contrato</b>	No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
<b>Interacción del cliente</b>	El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
<b>Grupos de trabajo</b>	Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio	Grupos grandes y posiblemente distribuidos
<b>Artefactos</b>	Pocos artefactos	Más artefactos
<b>Roles</b>	Pocos roles	Más roles
<b>Arquitectura</b>	Menos énfasis en la arquitectura del software	La arquitectura del software es esencial y se expresa mediante modelos

Aunque existen diferencias muy marcadas entre una y otra metodología. Cada una otorga beneficios en su implementación en desarrollo de software.



### 1.1.7 Scrum checklist

Scrum checklist es un framework con reglas simples, que espera ayudar a recordar las reglas diarias del método Scrum. Este framework muestra cómo crear un divertido y productivo ambiente de trabajo con el equipo, mantiene diferentes estrategias Scrum de acuerdo a el grado de aprendizaje que se tenga, por medio de esta clasificación permite adoptar el framework de acuerdo a tres niveles (G. Boris, 2012):

- 1. Principiantes de Scrum:** este checklist muestra como iniciar con 3 o 4 sprints exitosamente. Para facilitar la adopción de scrum en la organización.
- 2. Intermedio Scrum:** una vez que la adopción de scrum se ha dado por el checklist para principiantes, el siguiente proceso utiliza una guía del checklist que permite avanzar de manera individual.
- 3. Scrum Master con Experiencia:** utiliza el scrum checklist en esta parte como una guía para resolver problemas derivados de situaciones de estrés con el equipo.

### 1.1.8 Modelo de Madurez Ágil (AMM)

El modelo de madurez ágil (AMM) (Think Big, A. S & C. Patel et al., 2013, 2009), está diseñado para mejorar y realzar el método de desarrollo de software ágil e impulsar los principios y objetivos ágiles como el costo más bajo, la satisfacción del cliente, la calidad del software. En este modelo se muestra como las prácticas de desarrollo de software ágil maduran desde un nivel inicial basado en los principios y prácticas ágiles para la mejora continua. Cada nivel tiene objetivos predefinidos para ayudar al enfoque de la organización de sus actividades de mejora. La Figura 11 muestra el modelo y sus niveles, en este modelo cada nivel tiene pre definidos las metas que ayudan al usuario o a la organización a enfocarse en mejorar sus actividades.



Figura 11. Modelo de Madurez ágil.

El modelo de madurez ágil realiza una evaluación de la mejora del proceso de software para el ambiente de desarrollo de software ágil, la adaptabilidad del método, las prácticas, la problemática y las metas de los niveles de madurez (C. Patel et al., 2009).

### 1.1.9 Planteamiento del problema

En los años recientes el aumento de la importancia del software proporciona una oportunidad para incrementar la producción en las organizaciones de desarrollo de software, para satisfacer una amplia demanda de necesidades de diversos sectores industriales como tecnologías de Información, servicios, automotriz o de aviación, entre otros.

Esta situación hace que el desarrollo de software sea muy atractivo para las PYMEs de este sector (G. Piedad et al. & G. Gómez et al., 2013, 2014). En Latinoamérica alrededor del

99.5% son PYMEs (OCDE/CAF, 2019), las pequeñas y medianas empresas (PYMEs) la mayoría de veces no tienen acceso a recursos necesarios para iniciar proyectos tecnológicos o de innovación. desde el punto de vista estratégico, las PYMEs desarrollan sus recursos y capacidades limitadamente debido a la falta de prácticas gerenciales, habilidades y entrenamiento de su personal (S. Estrada et al, 2019). En este contexto, las PYMEs necesitan mejorar constantemente sus procesos en desarrollo de software, de manera que conserven el control y la calidad de sus productos, permitiéndoles mantenerse en el mercado con un crecimiento constante.

Por lo que las organizaciones han preferido el uso de los métodos ágiles en un esfuerzo por desarrollar software y ponerlo en el mercado tan pronto como es posible para satisfacer a sus clientes.

Desafortunadamente, la mayor parte de las implementaciones de los métodos ágiles se basan en los beneficios que ocurren en otras organizaciones (SEI, 2016) (LASHCP & VersionOne, 2015, 2014), dando como resultado que exista un periodo crítico entre el primer y segundo año en el que las PYMEs interrumpen el uso de estos métodos (LASHCP, 2014), y regresen sus malas prácticas anteriores. Esta situación se debe principalmente a la falta de conocimiento en la correcta implementación de un método ágil, reflejándose en productos de baja calidad y/o empresas frustradas por la falta de habilidad en el uso correcto de un método ágil.

### 1.1.10 Hipótesis

Si se dispone de un método de evaluación y evolución del conocimiento de una persona en un método ágil entonces se logra facilitar el diagnóstico y adquisición de conocimiento y habilidades en el uso del método ágil.

### 1.1.11 Objetivos de la investigación

En la siguiente sección se muestra los objetivos generados para la investigación.

#### 1.1.11.1 Objetivo general

Desarrollar un método de evaluación y evolución en el uso de métodos ágiles para pequeñas y medianas empresas de desarrollo de software.

#### 1.1.11.2 Objetivos específicos

- Analizar el estado del arte de marcos de trabajo, métodos, metodologías y/o estándares desarrollados para evaluar el uso de métodos ágiles en las PYMEs.
- Analizar el uso de los métodos ágiles en México.
- Definir un método de evaluación del uso de métodos ágiles en una PYME.
- Generar una herramienta web para la automatización del método de evaluación propuesto.

- Validar el método de evaluación propuesto a través de un estudio de caso en PYMEs.

### 1.1.12 Justificación

El crecimiento tecnológico es impulsado por la competencia y cambios globales en todos los aspectos organizacionales, lo cual provoca que las empresas dedicadas al desarrollo de software busquen alcanzar la eficiencia en sus procesos a costos menores. Adoptar nuevas formas de trabajo que permitan comunicar, entender, controlar, predecir para mejorar y con esto legitimar el producto ofertado (S. Buzz, 2004).

Por lo tanto, las organizaciones tienen un mayor interés en ser competitivas, rápidas y equidistantes con el cliente, impulsando cambios en las personas y procesos para mejorar la productividad en el desarrollo de software y competir con la creciente demanda de innovación (Pulse PMI, 2020).

Sin embargo, aun cuando existe una alta posibilidad de que los proyectos de desarrollo de software se terminen dentro del presupuesto usando un método ágil, la mayoría de las PYMES, no la ejecutan adecuadamente, dando como resultado que algunas empresas abandonen la implementación o uso del mismo método ágil.

Como solución a esta situación, se propone la definición de un método de evaluación del uso de los métodos ágiles a través de una herramienta, de manera que se conozca si ésta es ejecutada correctamente. Además, se tiene identificada la necesidad de asegurar la madurez en el uso de los métodos ágiles.

# Capítulo 2. Estado del arte

Este capítulo describe el estado del arte relacionado con la evaluación de los métodos ágiles, así como un análisis de trabajos relacionados con el trabajo de tesis. Además, se incluye una encuesta para conocer el uso de los métodos ágiles en México y un análisis de herramientas y los trabajos relacionados.

## 2.1 Revisión sistemática de la literatura

La Revisión Sistemática de Literatura (RSL), es un protocolo formal que permite la identificación, el análisis e interpretación de una investigación acerca de una idea o tema (Kitchenham & Charters, 2007). En una revisión sistemática es importante la definición de un protocolo para reducir la posibilidad de sesgo en la búsqueda, ya que el protocolo define los procedimientos utilizados para guiar una revisión sistemática (Selleri et al., 2015). Para la investigación se siguió el protocolo de revisión sistemática propuesto por (Kitchenham & Charters, 2007), el cual cuenta con tres fases principales: planificación, ejecución, y análisis de resultados, las cuales se describen en la Figura 12.

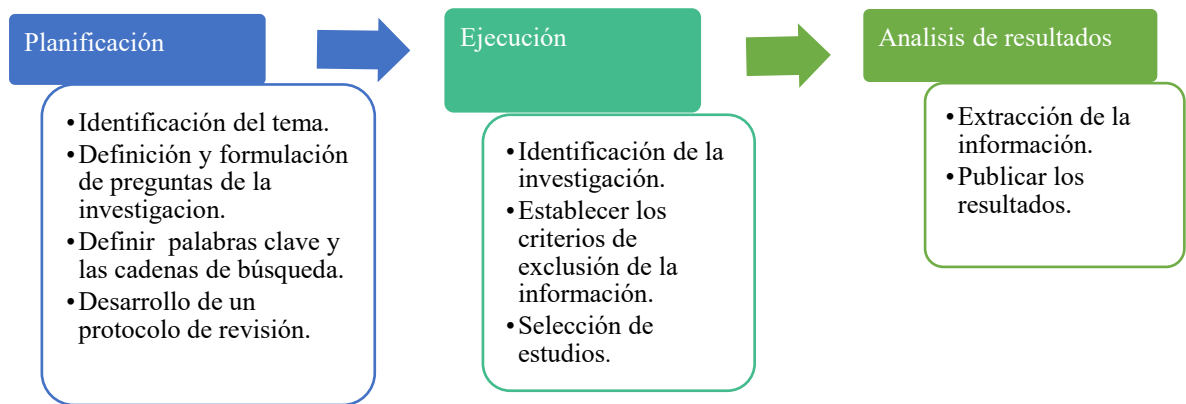


Figura 12. Actividades en el protocolo de revisión sistemática.

A continuación, se definen en detalle las actividades del protocolo de revisión sistemática para cada una de las fases indicadas anteriormente.

### 2.1.1 Planificación

En la fase de planificación se definen las actividades de la revisión sistemática, las cuales se detallarán en las secciones siguientes.

#### 2.1.1.1 Identificación de la necesidad de la RSL

Recoger e identificar datos precisos y actuales con respecto a los marcos, métodos y metodologías, cuyo objetivo es evaluar el uso y la correcta aplicación de métodos ágiles en las PYMES, así como los que se centran en el análisis de la madurez de la organización con respecto a la utilización del método ágil.

### 2.1.1.2 Preguntas de la investigación

Las preguntas planteadas para la obtención de la información son las siguientes:

- RQ1. ¿Qué modelos, metodologías o estándares se utilizan en las PYMEs en Latinoamérica?
- RQ2. ¿Cuál es el porcentaje de PYMEs de Latinoamérica que utilizan métodos ágiles?
- RQ3. ¿Hay marcos de referencia, métodos, metodologías para la evaluación para los métodos ágiles en las PYMEs?

### 2.1.1.3 Fuentes de datos

Las siguientes fuentes de datos se eligieron debido a que son repositorios importantes en la Ingeniería de Software son: IEEE Xplore, Elsevier Science (Science Direct), ScholarGoogle.

### 2.1.1.4 Cadenas de búsqueda

Para generar la cadena de búsqueda, se toman las palabras clave identificadas de las preguntas de investigación y el uso de conectores lógicos AND y OR. Las cadenas de búsqueda resultantes se muestran a continuación:

- CA-01 ((Current State AND (Latam OR Organization Software OR SME)) AND Agile Methodologies).
- CA-02 (Latam (Software Process AND Methodologies Agile) AND SME).



### 2.1.1.5 Criterios de selección

Los criterios de selección son reglas que aseguren que los estudios obtenidos contienen información pertinente relativa a las investigaciones que se llevaron a cabo. A continuación, se presentan los criterios definidos para esta investigación, como se muestra en la Tabla 5.

Table 5. Criterios de selección

Criterios de Inclusión	Criterios de Exclusión
IC-1. Estudios del 2010-2020.	EC-1. Estudios que no cuenten con resultados claros obtenidos en la aplicación de algún método ágil.
IC-2. Estudios en inglés y español.	EC-2. Estudios que no se realicen sobre algún método ágil y las PYMES.
IC-3. Estudios que al menos contengan tres palabras clave en el título y/o en el resumen.	EC-3. Estudios que no cumplan los criterios de inclusión.
IC-4. Estudios que muestran el estado de implementación de métodos ágiles en las pequeñas y medianas empresas.	
IC-5. Estudios que contengan técnicas y herramientas enfocadas en la evaluación de los métodos ágiles en la industria.	
IC-6. Estudios que evalúan la implementación o el uso de métodos ágiles en las PYMES.	
IC-7. Estudios que contienen análisis, evaluación y aplicación de métodos ágiles.	

### 2.1.2 Ejecución

A continuación, se muestran las estrategias para la extracción de los datos, así como los resultados.

#### 2.1.2.1 Estrategia de selección de estudios primarios

Como se muestra en la Figura 13, la implementación de la estrategia para la investigación está compuesta por 8 pasos, como sigue: 1) introducir la cadena de acuerdo a los requisitos de los

motores de búsqueda; 2) revisar los datos del estudio; 3) revisar el título del estudio; 4) revisar el resumen; 5) aplicar el resto de los criterios de selección; y 6) elegir el estudio.

La Figura 14, muestra la estrategia de búsqueda aplicada a esta investigación, con la cual, de un total de 477.163 estudios encontrados, después de aplicar los criterios de selección fueron seleccionados 55 estudios como estudios primarios, ya que cumplieron con los criterios. Estos estudios fueron analizados y utilizados para establecer los resultados de este trabajo de investigación. Esta estrategia para incluir estudios se realiza antes de aplicar los criterios de selección, si el estudio cumple con la estrategia se continua con los criterios de inclusión en la investigación.

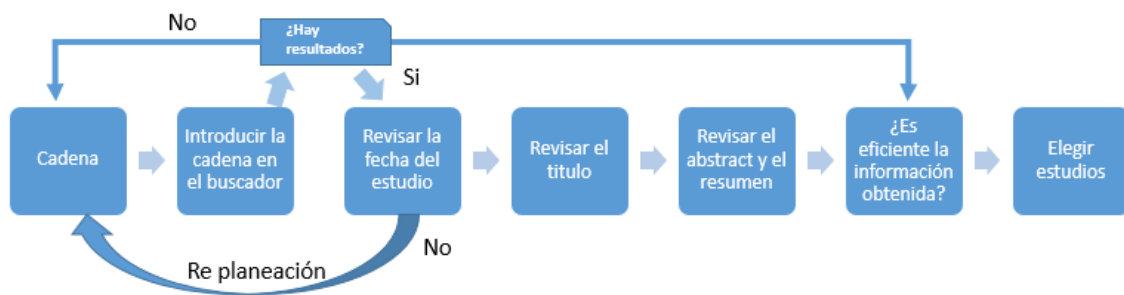


Figura 13. Estrategia de selección de estudios.

A continuación, se muestra la cantidad de estudios encontrados durante el proceso de selección.

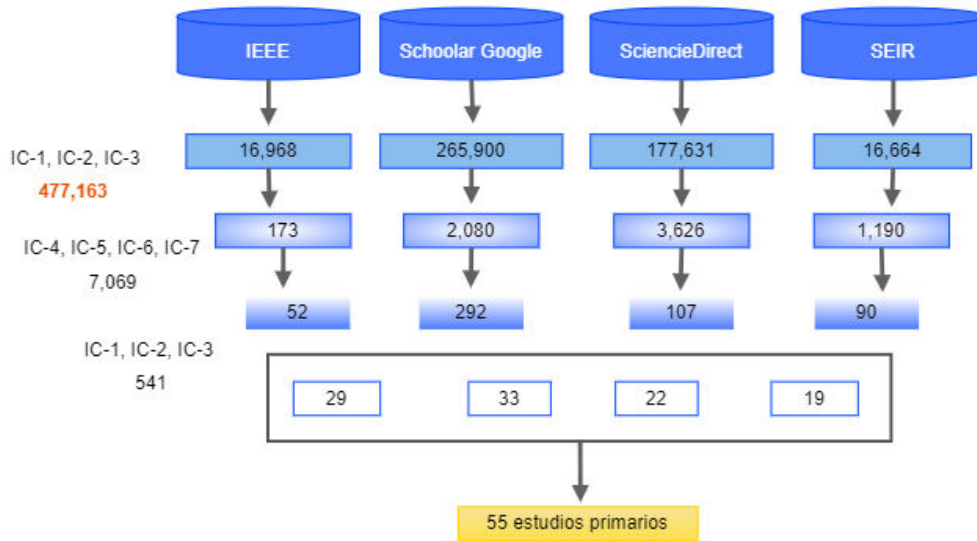


Figura 14. Cantidad de resultados durante la selección de estudios.

### 2.1.2.2 Extracción de datos

Los datos extraídos se registraron en una hoja de cálculo Excel. El uso de una hoja de cálculo permitió una mejor gestión de la clasificación estudios.

La hoja de cálculo contiene la siguiente información: título, autor, año, palabras clave, la fuente, la principal idea, abstracta y comentarios. Después de aplicar la estrategia de selección, se revisaron los artículos en su totalidad. Esta clasificación fue añadida a la hoja de cálculo para mejorar los resultados.

### 2.1.3 Resultados

En las siguientes secciones, los principales resultados obtenidos de la ejecución de la revisión sistemática se detallan.

### 2.1.3.1 Modelos, metodologías o estándares utilizados por las PYMES

En esta sección se muestra los resultados con respecto de los estudios que hacen referencia a los modelos, metodologías o normas utilizadas. De los 55 estudios se excluyeron 26 artículos, después de un análisis a profundidad debido a que la información no tenía el detalle requerido. Por lo tanto, los artículos primarios seleccionados finalmente son 33, la clasificación de los artículos se muestra en la Tabla 6.

Tabla 6. Clasificación de artículos.

<i>Modelos, Metodologías o estándares usados</i>	<i>No. De artículos</i>
Tradicional/ágil Híbridos(CMMI, XP)	4
Metodologías tradicionales/Modelos (CMM, CMMI, SPICE)	7
Métodos ágiles (XP, Rapid, Scrum, Kanban, SDM)	15
Estándares tales como la familia ISO	7

La **Tabla 6**, refleja la descripción de los modelos, metodologías o estándares utilizados por las PYMES, y que se encontraron en los estudios seleccionados.

### 2.1.3.2 Marcos de trabajo, métodos y metodologías para la evaluación de métodos ágiles

Acercas de la identificación de los marcos, métodos o metodologías para evaluar el proceso o el uso y la aplicación de método ágiles, se identificaron alrededor de 35 estudios enfocados en el uso, ejecución, o la aplicación de varios tipos de evaluación. Tabla 7 muestra una clasificación por tipo de evaluación.

Tabla 7. Tipo de propuesta desarrollada para evaluar *los métodos ágiles*.

<i>Tipo Evaluación</i>	<i>Cantidad de estudios</i>
Encuesta	3
Checklist	7
Metodología de Evaluación	16
Framework	8
Modelo de Evaluación	1
Herramienta	44*

\* Encontrado en una investigación de un estudio [31].

## 2.2 Encuesta

Se desarrolló una encuesta para conocer el estado de las PYME debido a la falta de información sobre el uso de métodos ágiles en las PYME en México. La Tabla 8 muestra la encuesta desarrollada.

Tabla 8. Encuesta para conocer el estado de las PYMEs de México.

<b>Encuesta para conocer los procesos de calidad en el desarrollo de software</b>
<b>DATOS GENERALES</b>
1.- ¿Nombre de la empresa en la que labora?
2.- ¿Dónde se encuentra ubicada la empresa?
3.- ¿Puesto en el que se desempeña?
4.- ¿Cuál es el tipo de empresa donde labora?
<b>USO DE METODOS, METODOLOGÍAS Y ESTÁNDARES</b>
5.- ¿La empresa cuenta con un departamento de calidad?
6.- ¿Qué metodologías y/o modelos manejas para asegurar la calidad del software?
7.- ¿Cuál es el proceso de calidad que utilizan en la empresa para el desarrollo de software?
8.- ¿Conoces y has utilizado las metodologías ágiles?
9.- ¿La metodología que conoces la has aplicado en los proyectos de la empresa donde laboras?
Si tu respuesta anterior fue NO, Favor de Responder lo siguiente. 10.- ¿En qué tipo de proyectos has aplicado la metodología que conoces?
11.- ¿Cuál es la metodología que has utilizado?
12.- ¿Cómo ha mejorado la aplicación de la metodología ágil en el proceso de calidad en la empresa?

### 2.2.1 Desarrollo de la Encuesta

Este cuestionario fue elaborado como parte del trabajo de tesis para complementar los resultados de la revisión sistemática, y que se pretende analizar que metodologías, procesos y estándares utilizan las organizaciones para el desarrollo de software con calidad.

Al analizar los resultados de la encuesta se muestra que el mayor porcentaje de los encuestados se centran en los programadores con el 33%, lo cual refleja que el personal de desarrollo se encuentra

más actualizado en el ámbito de métodos ágiles. El resto de los encuestados se centran en los siguientes puestos: Jefe del área de sistemas 15%, Gerente de calidad 15%, Ingenieros de software 12%, Asesor 6%, Líder del Desarrollo 9%.

Para visualizar mejor la distribución en la aplicación de esta encuesta, la cual se centra en el estado de Guanajuato con un 29%, de organizaciones que respondieron la solicitud de la encuesta, como se observa en la Figura 15.

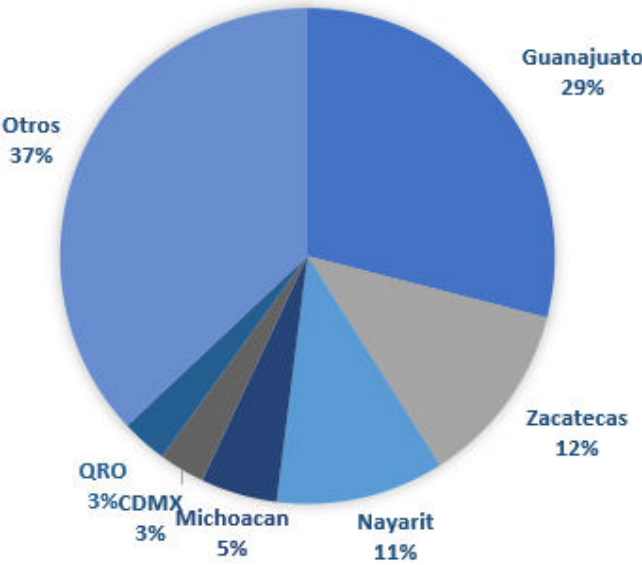


Figura 15. Ubicación de los resultados de la encuesta.

De acuerdo al tamaño de la empresa de desarrollo de software, como se muestra en la Figura 16, existen menos empresas medianas que conocen el método ágil.

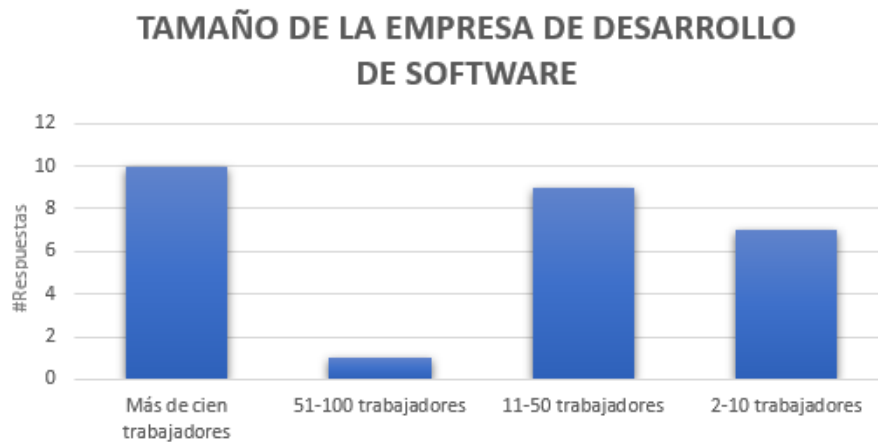


Figura 16. Tamaño de la empresa de desarrollo de software.

De acuerdo a la encuesta las metodologías y/o modelos que se usan para asegurar la calidad del software, las empresas que contestaron utilizan Scrum 37%, seguido de Kanban 14%, o no se utiliza nada para asegurarla. Como se muestra en la Figura 17.

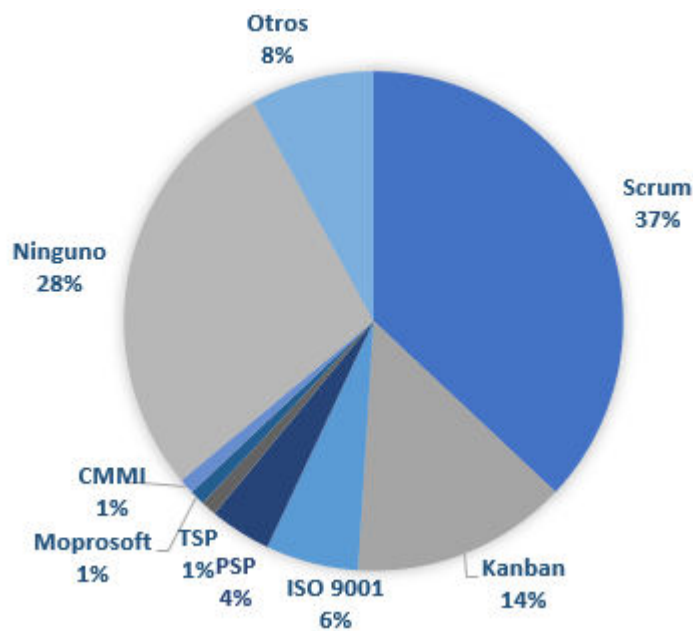


Figura 17. Métodos utilizados.

Sin embargo, los procesos que conocen y que han utilizado en la empresa, referente a la calidad en el desarrollo de software, la respuesta fue que el más utilizado son las pruebas unitarias, aunque se



menciona que aplicar una combinación de Scrum y pruebas unitarias sería lo ideal. Los datos proporcionados son los siguientes: Pruebas por pares con 3%, automatización con 1%, Pruebas unitarias con 8%, Pruebas de integración y despliegue con 3%, Estándares de codificación con 1%, ISO 9001 con 2% y Scrum con 3%, el resto no contestó.

Los resultados de la encuesta reflejan el conocimiento de los métodos ágiles, sin embargo, en México el uso e implementación de métodos ágiles es escaso y que se ha mantenido la aplicación de metodologías tradicionales y sus combinaciones.

## 2.3 Análisis de Herramientas

A continuación, se muestra un análisis de 44 herramientas encontradas; como resultado de la ejecución de una revisión sistemática. Este análisis se realizó para definir el estado actual de herramientas enfocadas en medir si una organización adopta y usa los métodos ágiles de manera adecuada.

### 2.3.1 Procedimiento de Análisis

En esta sección se muestra el procedimiento establecido para realizar el análisis de herramientas el cual incluye: a) descripción del tipo de análisis llevado a cabo y b) una serie de criterios que deben aplicarse.

### 2.3.2 Tipos de análisis

A) **Tipo de herramienta:** este análisis se centra en la clasificación de las herramientas de acuerdo con el tipo de herramienta; que podría ser: lista de verificación, encuestas, cuestionarios, test, juego de herramientas y otros. El tipo de herramientas se describe brevemente a continuación:

- *Lista de comprobación*: la herramienta enfocada en la evaluación del método ágil proporcionando "una lista de cosas que deben ser controlador o hechos". Ayuda a asegurar si un conjunto de tareas se ha llevado a cabo de una manera completa para reducir los fracasos asociados a la memoria humana y la atención.

- *Encuestas*: herramienta enfocada en la evaluación del método ágil que proporciona un conjunto de preguntas cortas que se debe responder y proporcionar un conjunto de opciones de respuestas cortas. Ayuda a preguntar a muchas personas una serie de preguntas con el fin de reunir información acerca de lo que mucha gente piensa o por algo.

- *Cuestionarios*: herramienta enfocada en la evaluación del método ágil, proporcionando un conjunto de preguntas para ser contestadas y proporciona espacios para la recopilación de respuestas abiertas y largas. Ayuda a recoger hechos u opiniones acerca de algo. Un experto deberá analizar las respuestas.

- *Test*: herramienta enfocada en la evaluación del método ágil proporcionando un conjunto de preguntas para medir las habilidades o conocimientos de una persona. Las preguntas deben ser respondidas a través de la selección de una respuesta de un conjunto de frases cortas o palabras con un valor asociado.

- *Kit de herramientas*: herramientas enfocadas en la evaluación del método ágiles, proporcionando software diseñado para evaluar la agilidad a través de Excel, Word o aplicaciones en línea que permiten la recogida de datos relativos a los métodos ágiles.

**B) Conjunto de criterios:** para la realización del análisis se aplican un conjunto de análisis a cada tipo de herramientas.

### 2.3.3 Establecimiento de Criterios

Con el fin de lograr un análisis imparcial de las 44 herramientas, se establecieron un conjunto de criterios. La lista de los criterios establecidos se muestra en la Tabla 9.

Tabla 9. Criterios establecidos.

Criterios	Descripción
<b>Metodología evaluada</b>	La metodología ágil evaluada por la herramienta como XP, Scrum u otros, que incluyen otras metodologías ágiles (FDD, cristal, Kanban, entre otros)
<b>Enfoque</b>	Si la herramienta se centra en la evaluación de la metodología ágil aplicación, la adopción o ambos <ul style="list-style-type: none"><li>• <i>Adopción</i>: la herramienta se centra en evaluar el nivel de adopción y aceptación de la metodología ágil.</li><li>• <i>Implementación</i>: la herramienta se centra en la evaluación del conjunto de las actividades hacia el uso de la metodología ágil</li></ul>
<b>Objetivo</b>	Cuál es el alcance de la herramienta; Generar una discusión o proporcionar una guía
<b>Apoyo al usuario</b>	Si la herramienta tiene un conjunto de instrucciones para su uso
<b>Principios Ágiles</b>	Si la herramienta toma en cuenta los principios ágiles
<b>Puntuación</b>	Si la herramienta se basa en una puntuación específica para evaluar la metodología ágil

### 2.3.4 Análisis de resultados

En esta sección se muestra el análisis realizado y sus resultados.

#### 2.3.4.1 Tipos de Herramientas

Este análisis muestra la clasificación de las 44 herramientas de acuerdo con su tipo. La Figura 18 muestra un gráfico con la clasificación de las 44 herramientas por tipo, la mayoría de las herramientas se basan en conjuntos de herramientas y listas de control. Este dato es importante con el fin de entender qué tipo de herramienta tiene una mejor aceptación en un entorno ágil.

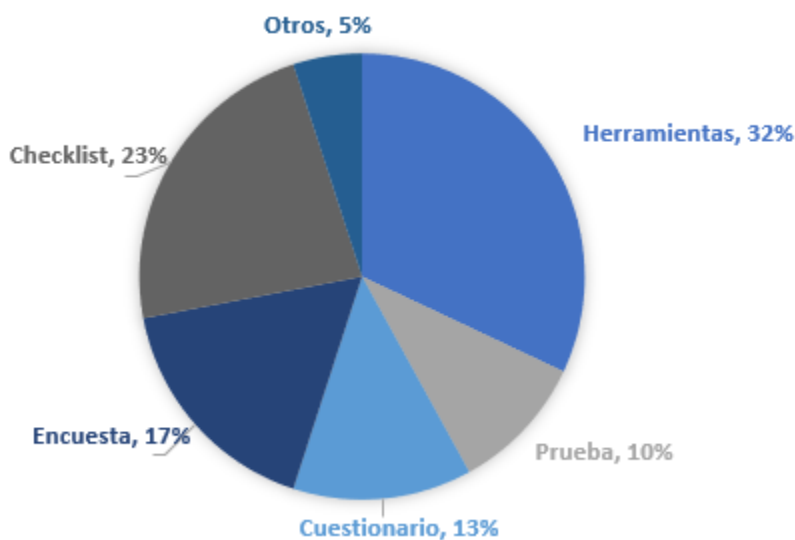


Figura 18. Clasificación de herramientas por tipo.

#### 2.3.5 Aspectos Enfocados

En esta sección se muestran los resultados de la aplicación del conjunto de criterios que se muestran en la Tabla 9 de la sección 3. Este análisis tiene como objetivo identificar qué aspectos son enfocados por las herramientas.

Es importante mencionar que después de recopilar la información de las 44 herramientas, como primer análisis se aplicaron cuatro criterios: (1) herramientas que tienen información pública

disponible en cualquier fuente; (2) es una herramienta software centrado en PYMES; (3) es una herramienta de software para evaluar un método ágil, y (4) explica claramente cómo se aplican los principios ágiles.

Después de aplicar los criterios se seleccionaron aquellas herramientas de software que cumplieran al menos con tres de los criterios. Como resultados, se seleccionaron 34 herramientas para realizar el análisis.

### 2.3.5.1 Herramienta tipo checklist

El análisis de la aplicación de los criterios a las herramientas de software de checklist se muestra en la Tabla 10.

Como muestra en la Tabla 10, después de aplicar los criterios establecidos los resultados obtenidos son los siguientes: (a) *métodos evaluadas*: la mayor parte de las herramientas de evaluar el método Scrum; (b) *enfoque*: todo se centran en la aplicación del método; (c) *Objetivo*: la mayoría se centran en proporcionar una guía para la introducción en un entorno ágil y sólo tres se centran en la evaluación del método ágil; (d) *apoyo a los usuarios*: la mayoría proporcionan instrucciones para el usuario sobre la forma de utilizarlo; (e) *principios ágiles*: la mayoría abordan los principios ágiles; y (f) *Puntuación*: ninguno tiene una puntuación para hacer la evaluación.

Tabla 10. Herramientas de tipo checklist.

Herramienta	Método Evaluación	Enfoque *	Objetivo	Apoyo al Usuario	Principios Ágiles	Puntuación **
<b>The Unofficial Scrum Checklist (Kniberg, 2010)</b>	Scrum	IM	Genera discusión	Instrucciones	Si	NA
<b>Questions for Transitioning to</b>	Otros	IM	Evalúa	Instrucciones	Si	NA

<b>Agile (Rothman, 2013)</b>						
<b>Scrum Checklist framework (Gloger, B, 2012)</b>	Scrum	IM	Evalúa	Instrucciones	Si	NA
<b>Corporate Agile 10-point checklist (Yatzeck, E, 2012)</b>	Otros	IM	Provee una guía	No	No	NA
<b>Joe's Unofficial Scrum Checklist (Little, 2016)</b>	Scrum	IM	Genera discusión	Instrucciones	Si	NA
<b>How to Measure Team Agility (L. Lagestee, 2012)</b>	Otros	IM	Evalúa	Instrucciones	Si	0-4
<b>Open Assessments (Scrum.Org, 2015)</b>	Scrum	IM	Provee una guía	Instrucciones	Si	NA
<b>An Organizational Transformation Checklist (M. Sahota, 2012)</b>	Otros	IM	Provee una guía	Instrucciones	Si	NA
<b>The InfoQ Minibook Scrum Hard Facts: Roles. Artifacts. All meetings (Info Q, 2012)</b>	Otros	IM	Provee una guía	No	Si	NA

<b>Skills Self-Assessments (BPMInstitute,2020)</b>	Agile	AD	Provee una guía	Instrucciones	Si	0-5
--	-------	----	-----------------	---------------	----	-----

\*Los valores de la columna b son: (IM) Implementación o (AD) Adopción.

\*\*NA significa que la herramienta no está disponible.

### 2.3.5.2 Herramienta de software tipo Encuestas

El análisis de la aplicación de los criterios a las herramientas de software de encuestas se muestra la Tabla 11, después de aplicar los criterios establecidos los resultados obtenidos son los siguientes: (a) *métodos evaluadas*: la mayor parte de las herramientas de investigación de otros métodos ágiles como Kanban, Sistemas de Desarrollo Método Dinámico (DSDM) y Feature Driven Development (FDD); (b) *enfoque*: todo se centran en la aplicación de la metodología; (c) *Objetivo*: la mayoría se centran en la evaluación de la metodología ágil; (d) *apoyo a los usuarios*: todos proporcionan instrucciones para apoyar al usuario sobre cómo utilizarlo; (e) *principios ágiles*: ninguno se ocupa de los principios ágiles; y; (f) *anotar*: la mayoría tienen una puntuación para hacer la evaluación.

Tabla 11. Herramienta tipo encuesta.

Herramienta	Metodología Evaluación	Enfoque *	Objetivo	Apoyo al Usuario	Principios Ágiles	Puntuación **
<b>42 point test: How Agile are You (K, Waters, 2008)</b>	Scrum & Otros	AD	Evalúa	Instrucciones	No	1/ 0

<b>Team Barometer (Jalén J., 2014)</b>	XP	Enfoque en la relación del equipo	Genera una discusión	Instrucciones	No	NA
<b>Agile Maturity Self- Assessment (Iver R. 2010)</b>	Otros	AD	Evalúa	Instrucciones	No	1/ 0
<b>Comparative Agility (Rubin, M. 2008)</b>	Scrum & XP	IM	Evalúa	Instrucciones	Si	1/ 0
<b>Agile Enterprise (Storm- Consulting, 2016)</b>	Otros	IM	Evalúa	Instrucciones	Si	1/ 0
<b>Enterprise Agility Maturity Matrix (DrAgile, 2015)</b>	Scrum, Xp &  Otros	IM	Evalúa	Instrucciones	Si	NA

\*Los valores de la columna b son: (IM) Implementación o (AD) Adopción.

\*\*NA significa que la herramienta no está disponible.



### 2.3.5.3 Herramientas tipo Cuestionario

Para el análisis se utilizan los criterios aplicados en las herramientas de cuestionario que se muestran en la Tabla 12.

Tabla 12. Herramienta tipo cuestionario.

Herramienta	Método Evaluación	Enfoque *	Objetivo	Apoyo al Usuario	Principios Ágiles	Puntuación **
<b>Questionnaire for Assessing your Client's Agility (Business, T.D. 2015)</b>	Otros	AD	Evalúa	Instrucciones	Si	1/0
<b>Depth of Kanban (Development L.A.S., 2015)</b>	Otros	IM	Evalúa	Instrucciones	No	NA
<b>Agile Adoption Interview (PLADcloud, 2020)</b>	Otros	IM	Evalúa	Instrucciones	Si	0-100%

\*Los valores de la columna b son: (IM) Implementación o (AD) Adopción.

\*\*NA significa que la herramienta no está disponible.

Como se muestra en la Tabla 12, después de aplicar los criterios establecidos los resultados obtenidos son los siguientes: (a) *Método de evaluación*: las dos herramientas evalúan otros métodos ágiles como

Kanban, Método de Desarrollo de Sistemas Dinámicos (DSDM) y Feature Driven Development (FDD); (b) *enfoque*: se centra en la aplicación y en la adopción; (c) *Objetivo*: ambas herramientas se centran en la evaluación del método ágil; (d) *apoyo de usuario*: ambas herramientas proporcionan instrucciones para apoyar al usuario en forma de utilizarlos; (e) *principios ágiles*: sólo uno aborda los principios ágiles y; (f) *Puntuación*: sólo una tiene una puntuación para hacer la evaluación.

#### 2.3.5.4 Herramienta de software tipo Prueba

El análisis de la aplicación de los criterios a las herramientas de software de lista de verificación se muestra en la Tabla 13.

Tabla 13. Herramienta de tipo prueba.

Herramienta	Métodos Evaluación	Enfoque *	Objetivo	Apoyo al Usuario	Principios Ágiles	Puntuación **
<b>ScrumButt Test aka the Nokia Test (Scruminc, 2015)</b>	Scrum	IM	Provee una guía	Instrucciones	Si	1/0
<b>Borland Agile Assessment (Trenches, F. 2015)</b>	Scrum	IM	Evalúa	Instrucciones	Si	1/0
<b>Agility Assessment</b>	Otros	IM	Evalúa	Instrucciones	Si	0/10

<b>(Refine M, 2020)</b>						
-----------------------------	--	--	--	--	--	--

\*Los valores de la columna b son: (IM) Implementación o (AD) Adopción.

\*\*NA significa que la herramienta no está disponible.

Como muestra la Tabla 13, después de aplicar los criterios establecidos los resultados obtenidos son los siguientes: (a) *Método de evaluación*: las dos herramientas de evaluar el método Scrum; (b) *enfoque*: ambas herramientas se centran en la puesta en práctica; (c) *Objetivo*: una se centra en proporcionar una guía para la introducción en un entorno ágil y la otra se centra en la evaluación del método ágil; (d) *Apoyo de usuario*: ambas herramientas proporcionan instrucciones para apoyar al usuario en forma de utilizarlos; (e) *Principios ágiles*: ambas herramientas abordan los principios ágiles; y (f) *Puntuación*: ambos proporcionan una puntuación para hacer la evaluación.

### 2.3.5.5 HERRAMIENTAS DE TIPO: Kit de Herramientas de software

El análisis de la aplicación de los criterios a las herramientas de software kit de herramientas se muestra en la Tabla 14.

Como muestra la Tabla 14, después de aplicar los criterios establecidos los resultados obtenidos son los siguientes: (a) *Método de evaluación*: la mayor parte de las herramientas de investigación se centran en evaluar otros métodos ágiles como Kanban, Sistemas Dinámicos Método de Desarrollo (DSDM) y Feature Driven Development (FDD); (b) *Enfoque*: la mayor parte de ellos se centran tanto en la aplicación como en la adopción; (c) *Objetivo*: la mayoría se centran en tanto que proporciona una guía para introducir dentro de un entorno ágil y evaluar el método ágil; (d) *Apoyo a los usuarios*: todos proporcionar instrucciones para apoyar al usuario en forma de utilizarlos; (e) *Principios ágiles*: todos abordan los principios ágiles; y (f) *Puntuación*: la mayoría de ellos tienen una puntuación para hacer la evaluación.

Tabla 14. Herramienta de tipo software

Herramienta	Método Evaluación	Enfoque *	Objetivo	Apoyo al Usuario	Principios Ágiles	Puntuación **
<b>Enterprise Agile Practice Assessment Tool (Miller, s. 2012)</b>	Otros	AD	Evalúa	Instrucciones	Si	NA
<b>Readiness &amp; Fit Analysis (LLC, A. D. Assess, 2012)</b>	Otros	AD	Provee una guía	Instrucciones	Si	NA
<b>Agile Journey Index (Schenck, 2015)</b>	Otros	AD & IM	Provee una guía	Instrucciones	Si	1/0
<b>Agile Maturity Matrix in JIRA (IBM, 2015)</b>	Scrum	AD & IM	Evalúa	Instrucciones	Si	1/0
<b>IBM DevOps Practices Self-Assessment</b>	Otros	AD & IM	Evalúa	Instrucciones	Si	NA

<b>(Sidky, S. E. 2012)</b>						
<b>Ready for Agile Part 1 and Part 2 (Agility health, 2015)</b>	Scrum & Otros	AD & IM	NA	Instrucciones	Si	Ágil Parte 1 y Parte 2 [33]
<b>Agile Health Radar (Ivar Jacobson, 2015)</b>	Otros	AD & IM	Provee una guía	Si	Si	Gráfico de Radar ágil [34]
<b>Agile Essentials (card game) (Scheck, M. 2013)</b>	Otros	AD	Provee una guía	Instrucciones	Si	NA

\*Los valores de la columna b son: (IM) Implementación o (AD) Adopción.

\*\*NA significa que la herramienta no está disponible.

### 2.3.5.6 Mejores herramientas

Después de analizar las herramientas se seleccionaron aquellas herramientas que se consideraron “*las mejores herramientas*” debido a que cubren los siguientes aspectos: (a) proporcionan instrucciones para apoyar al usuario en forma de utilizarlos; (b) proporcionan un punto de inicio y final de cada sección que se evalúan y proporcionan monitoreo para el progreso a través

de sus niveles; (c) tienen una puntuación para evaluar el método ágil; y (d) proporcionan resultados, pero no obligan al usuario a pagar la consultoría para ver sus resultados.

La Tabla 15, muestra el análisis realizado de las mejores herramientas para identificar los aspectos tratados en relación con un método ágil. Además, teniendo en cuenta que seis de nueve herramientas se basan en la evaluación del método Scrum, se tomaron como base de llevar a cabo el análisis los roles y artefactos de Scrum.

Tabla 15. Mejores herramientas.

Herramientas	Enfoque de roles ágiles por herramienta	Artefactos del método
<b>scrum checklist framework</b>	dueño del producto, scrum master and scrum equipo.	product backlog, reunión de planificación de Sprint; sprint backlog; reunión scrum diaria; la ejecución de velocidad; Revisión del Sprint; retrospectiva del sprint y Burndown.
<b>open assessment</b>	dueño del producto, scrum master and scrum equipo.	reunión de planificación de Sprint; sprint backlog; la ejecución de velocidad; Revisión del Sprint, retrospectiva del sprint.
<b>42 point test: how agile are you</b>	dueño del producto and scrum equipo.	product backlog, reunión de planificación de Sprint; sprint backlog; reunión scrum diaria; Revisión del Sprint y retrospectiva del sprint.
<b>Enterprise Agility Maturity Matrix</b>	dueño del producto, scrum master and scrum equipo.	product backlog; sprint backlog; reunión scrum diaria; la ejecución de velocidad; Revisión del Sprint, retrospectiva del sprint.
<b>Questionnaire for Assessing your Clients Agility</b>	dueño del producto, scrum master and scrum equipo.	reunión de planificación de Sprint; sprint backlog; reunión scrum diaria; Revisión del Sprint y de quema.

<b>Enterprise Depth of Kanban</b>	scrum equipo.	product backlog, reunión de planificación de Sprint; sprint backlog; Revisión del Sprint, Sprint y retrospectiva reunión scrum diaria.
<b>ScrumButt aka the Nokia Test</b>	dueño del producto.	product backlog, reunión de planificación de Sprint; sprint backlog; reunión scrum diaria; la ejecución de velocidad; Revisión del Sprint; retrospectiva del sprint y Burndown.
<b>Agile Journey Index</b>	scrum master and scrum equipo.	product backlog; sprint backlog; reunión scrum diaria; opinión de sprint, sprint retrospectivo y burndown.

### 2.3.5.7 Hallazgos

Con los hallazgos encontrados dentro del análisis de las herramientas se han identificado las estructuras basadas en equipos, así como la planificación de los procesos a nivel desarrollador, sin embargo existe deficiencia en el control y seguimiento de cada etapa, ya que resulta un poco complejo el uso del instrumentó y además es relativamente poco flexible, pues las evaluaciones se encuentran realizadas la gran mayoría en documentos con macros, y no cuentan con una plataforma sólida para el almacenamiento de la información y seguimiento.

### 2.3.5.8 Necesidades

A pesar de que todas las herramientas utilizan y aplican preguntas para evaluar el método ágil, tienen puntos de mejora, los cuales se identifican como necesidades. A continuación, se mencionan las necesidades identificadas:

- No se menciona si la evaluación se apega a los principios ágiles.

- Los resultados que se muestran son porcentajes sobre las actividades del método que se aplican en las preguntas.
- Las actividades son de implementación del método ágil (Scrum).
- Las actividades son acuerdo al rol del método Scrum.
- No se visualiza a simple vista la aplicación de los principios ágiles.
- No permite dar un seguimiento y control de la evaluación sobre las actividades de acuerdo al rol. No se muestra resultados claros para la toma de decisiones rápidamente.
- La herramienta misma no permite conocer el nivel.
- Se permite acceder a la evaluación.
- Los resultados de la evaluación obligan a contactar con la empresa proveedora para generar consultoría personalizada.

## **2.4 Trabajos Relacionados**

En esta sección, se muestra los trabajos relacionados a la investigación y que sirvieron como base en esta investigación.

### **2.4.1 Metodología para la evaluación de software ágil en enfoques de desarrollo**

Los métodos ágiles proporcionan a un equipo de una organización la capacidad para adoptar un subconjunto seleccionado de fundamentos y prácticas de su cultura, sus valores, y de acuerdo a los tipos de sistemas que desarrollan. Los enfoques de evaluación ágiles existentes se centran en un



análisis comparativo, o están limitados en su alcance y aplicación, donde este método propone un enfoque estructurado, sistemático y exhaustivo para determinar la "bondad" de los métodos ágiles. Examinamos un método ágil basado en (1) su adecuación, (2) la capacidad de la organización para apoyar los principios y prácticas adoptadas especificados por el método, y (3) la eficacia del método, además identifica los objetivos ágiles, los principios que sustentan los objetivos, las practicas ágiles y su relación, además incorpora indicadores para evaluar la eficacia y el grado de implementación de métodos ágiles (James D. et. Al 2011, 54).

#### 2.4.2 Un Framework para Evaluación de Métodos Ágiles

Los autores muestran un marco de trabajo para la evaluación para los métodos ágiles de desarrollo, en específico se puede aplicar a Scrum y eXtreme Programming (XP), este framework evalúa como los métodos ágiles satisfacen los principios básicos definidos en el Manifiesto Ágil. Este framework mide la manera de cómo son cumplidos los postulados del manifiesto ágil, además utiliza indicadores de medidas para los postulados, los cuales son: a) valorar las necesidades del equipo de desarrollo más que el proceso y las herramientas, b) valorar el funcionamiento del desarrollo de software más que la documentación exhaustiva, c) valorar la colaboración del cliente por sobre lo acordado en el contrato, y d) valorar las soluciones en respuesta al cambio más que el seguimiento de un plan. Para la aplicación de este framework se recomienda utilizarlo cuando se requiera elegir un método ágil (K. Mendes, C. et. Al, 2009).

#### 2.4.3 Scrum Checklist

Scrum checklist es una guía que permite adoptar el método ágil Scrum en un ambiente organizacional para equipos con unas simples reglas puede ayudar a recordar las reglas y seguirlas en

el estrés del trabajo diario. Además, muestra cómo crear un divertido y productivo trabajo en un ambiente favorable para los equipos de trabajo, cuando no se tiene ningún conocimiento; cuando se tiene, pero aún se desconoce su uso y finalmente con experiencia en proyectos ágiles (B. Glober, 2012).

#### 2.4.4 Modelo de Madurez Ágil (AMM)

El modelo de madurez ágil (AMM) permite que los métodos ágiles de desarrollo de software permitan adoptar y vigilar a aquellas prácticas de forma continua para maximizar sus beneficios. Este modelo presenta un proceso de evaluación de la capacidad de adaptación, la evaluación de la idoneidad y la mejora en el marco para la evaluación y mejora ágil, así como las mejores prácticas, en cinco niveles definidos, además para comprobar el modelo propuesto, se desarrolló una herramienta web (C. Patel et, al. 2009).

La Tabla 16. Muestra un resumen del análisis de los trabajos relacionados.

Table 16. Trabajos Relacionados.

Trabajos Relacionados	Características				
	<b>Sistema de configuración</b>	<b>Evaluación por Selección</b>	<b>Validación de madurez</b>	<b>Gráficos de resultados</b>	<b>Orientación grafica</b>
<b>Metodología para la evaluación de software ágil enfoques de desarrollo</b>	NO	NO	NO	SI	SI
<b>Un Framework para Evaluación de Métodos Ágiles</b>	NO	SI	SI	SI	SI

<b>Scrum Checklist</b>	NO	NO	NO	SI	SI
<b>Modelo de Madurez Ágil (AMM)</b>	NO	SI	SI	SI	SI

## 2.5 Análisis de tecnologías

Como soporte al método de evaluación en el uso de métodos ágiles, se desarrolla una herramienta que permita interactuar al usuario una forma de evaluación interactiva, para lograr esto se generó una herramienta interactiva de evaluación, que le dé al usuario una experiencia rápida con resultados precisos.

Para realizar el desarrollo de la herramienta se realizó un análisis de las tecnologías que permita alcanzar la meta, para esto existen varios lenguajes de programación y tecnologías de gráficos disponibles para su aplicación. A continuación, se describen las tecnologías existentes que permiten desarrollo de software interactivo. La siguiente clasificación hace referencia a tecnologías que se aplican hoy en día (Think Big, A.S,2013).

### 2.5.1 Arquitectura en Capas

La programación por capas utiliza una arquitectura cliente-servidor cuyo objetivo principal es separar la lógica de negocios de la lógica de diseño (ej. separar la capa de datos de la capa de presentación). Su ventaja principal es que el desarrollo puede ser llevado a cabo usando varios niveles, y en caso de que realizarse un cambio, sólo se impacta en el nivel requerido sin tener que revisar en todo el código (E. López, 2016). Véase Figura 19.

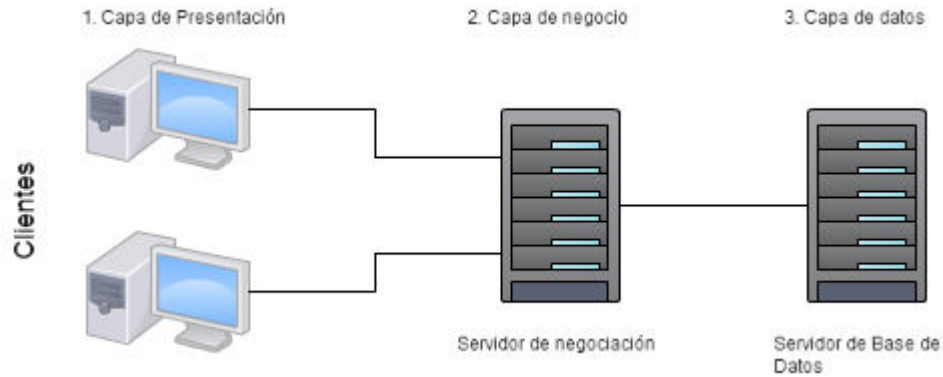


Figura 19. Modelo en capas.

### 2.5.2 Arquitectura tres capas

Modalidad en el desarrollo de software que permite separar la lógica de negocio, de presentación (diseño) y de datos (H. Flórez, 2010).

- Capa Presentación, muestra el diseño de la interfaz gráfica de la información.
- Capa de Lógica, contiene la lógica abstraída del negocio y que es plasmada en algoritmos de la aplicación.
- Capa Datos, se realizan las operaciones sobre los datos.

### 2.5.3 Base de Datos

Para generar una programación adecuada es necesario mantener un esquema de almacenamiento, A continuación, se describen la base de datos:

### 2.5.3.1 ORM (object-relational mapping)

Una base de datos relacional está formada por un mapeo objeto-relacional, generando en la aplicación estructuras definidas como clases orientadas a objetos, las cuales contienen en su definición la configuración del mapeo relacional, es decir, en la aplicación se tiene acceso a la base de datos relacional a través de objetos virtuales. Esta característica permite aplicar herencia y polimorfismo directamente sobre los datos almacenados. (H. Flórez, 2010).

### 2.5.3.2 Doctrine

Doctrine es el ORM utilizado por Symfony por defecto, aunque además se puede utilizar a través de Composer en cualquier proyecto, ya que es totalmente independiente (N. Blanco, 2018).

Este framework permite trabajar con objetos y clases de la aplicación, accediendo a los datos almacenados, permitiendo la integración de las herramientas y de Bootstrap en la aplicación web.

## 2.5.4 LENGUAJES

A continuación, se realiza una comparación de lenguajes para su selección y aplicación en la herramienta desarrollada.

### 2.5.4.1 Lenguaje HTML 5

Hyper Text Markup Language (HTML), Lenguaje de marcado que permite definir y estructurar el funcionamiento del contenido de un sitio web (L. Designer & Hypertextual, 2013).

Véase Tabla 17.

Tabla 17. Características HTML5.

Características	Ventajas	Tipo de aplicaciones
Soportado por cualquier navegador web	permite que los diseñadores de páginas web “rendericen” las imágenes.	Aplicaciones web.
Soporte de app cache	Permite que los navegadores web ejecuten video.	Aplicaciones móviles
Soporte de eventos de arrastrar y soltar	Permite trabajar de forma Off-Line.	Aplicaciones Cliente/servidor
Soporta programación asíncrona	Permite visualizar rápidamente el contenido de una pagina	
Uso de HTML, CSS, Document Object Model (DOM) y JavaScript	Mejora la búsqueda semántica de internet	
Reduce el uso de plugins externos		
Manejo de errores		
Dispositivos independientes		
Proceso de desarrollo público		
Contenido multimedia		

## 2.5.4.2 Lenguaje PHP

Hypertext Preprocessor (PHP), lenguaje libre de alto nivel utilizado para realizar las acciones necesarias en un servidor web, concentrado en páginas HTML. (Ibrugor, 2014 & Christian V., 2001).

Véase Tabla 18.

Tabla 18. Características PHP.

Características	Ventajas	Tipo de aplicaciones
Sencilla de usar.	Lenguaje multiplataforma.	Aplicaciones web.
Alto rendimiento.	Fácil de aprender.	Aplicaciones móviles.
Código abierto.	Orientado a aplicaciones web.	Aplicaciones Cliente/servidor.
Documentación.	Integración con conectores de base de datos.	
Orientado a objetos.	Tiene mucha documentación.	
Biblioteca nativa de funciones.	Lenguaje de código libre y gratuito.	
No requiere definición de tipo de variables.	Funciones nativas.	
Manejo de excepciones.	Múltiples Frameworks.	
Utiliza modelo vista controlador.		

## 2.5.4.3 Lenguaje ASP.NET

Lenguaje de programación diseñado para desarrollo web empresariales. Forma parte de .NET Framework; compatible con el lenguaje común en tiempo de ejecución (CLR), es sencillo, seguridad de tipos, herencia y orientado a objetos (Violeta, M. Microsoft, I, P, 2013). Véase Tabla 19.

Tabla 19. Características ASP.NET.

Características	Ventajas	Tipo de aplicaciones
Sencilla de usar.	Implementación de una interfaz de usuario Web compleja.	Aplicaciones web.
Elimina elementos que no son utilizados.	Mantiene Cache.	Aplicaciones móviles.
Lenguaje de última generación.	Carpetas especializadas.	Aplicaciones Cliente/servidor.
Orientado a objetos.	Configuración en ficheros.	
Orientado a componentes.	No utiliza frames.	
Recolector de basura.	compatibilidad con XML y servicios Web.	
Seguridad.	Facilidad de adaptación con complementos.	
Multilinguaje.	Compatibilidad con multitud de lenguajes.	
Escalable.		

A continuación, se muestra la comparación de las tecnologías analizadas, en la Tabla 20.

Tabla 20. Comparación de Tecnologías.

Características para mi herramienta	HTML5	PHP	ASP
Sencilla de usar	X	X	X
Control de Versiones		X	X
Soporta plugins	X	X	X



Soportado por cualquier navegador web	X	X	
Permite trabajar de forma Off-Line.	X	X	X
Programación asíncrona	X	X	X
Multiplataforma	X	X	X
Herramienta Gratuita	X	X	
Aplicaciones web	X	X	X
Aplicaciones móviles	X	X	X
Aplicaciones Cliente/Servidor	X	X	X
Protección de archivos		X	X
Administración remota		X	X

#### 2.5.4.4 Justificación de la elección de tecnología

Debido a los resultados mostrados, se puede discernir que la tecnología que es potencialmente más adecuada a los objetivos de la tesis, se desarrollará la herramienta con el lenguaje PHP.

Para la selección de la metodología de desarrollo se consideró la tecnología de base de datos y de lenguaje, que permiten la incorporación de herramientas que mejoran la experiencia de usuario y facilitan su mantenimiento, además de incorporar el método Scrum y la arquitectura en capas, que juntas mejoran la organización de los módulos de desarrollo y la interacción de diferentes herramientas que permiten mejorar su operación.

# Capítulo 3. Desarrollo de la propuesta del método

En este capítulo se describe la metodología que se estableció para el desarrollo de la Tesis. Además, se presenta un método de evaluación y evolución de métodos ágiles propuestos, como solución para resolver la problemática planteada en el Capítulo 1, que comprende la evaluación en el uso de métodos ágiles y proporcionar una guía para la evolución en el uso del método.

## 3.1 Metodología para desarrollo de la tesis

El objetivo de esta investigación es desarrollar un método de evaluación y evolución en el uso de métodos ágiles en pequeñas y medianas empresas de desarrollo de software. Para lograrlo se estableció una metodología que consta de seis pasos en la Figura 20 y que se describen brevemente a continuación.

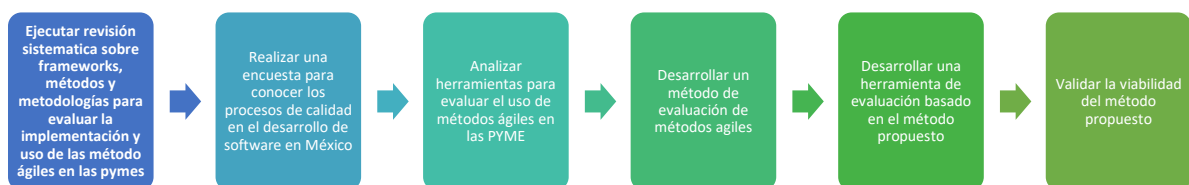


Figura 20. Método utilizado en la investigación.

1. *Ejecutar revisión sistemática sobre marcos de trabajo, métodos y metodologías para evaluar la implementación y uso de los métodos ágiles en las PYMEs:*

Esta actividad se realizó para obtener un panorama actual acerca del tema, obteniendo los estudios relacionados con las formas de evaluación existentes de métodos ágiles. Para lograrlo, se llevó a cabo una SRL enfocada en identificar el estado de arte centrado en los marcos de trabajo, métodos y metodologías diseñadas para la evaluación de la implementación y uso de los métodos ágiles en las pequeñas y medianas empresas (PYMEs). Esta actividad se encuentra detallada en el Capítulo 2 sección 2.1.

*2. Realizar una encuesta para conocer los procesos de calidad en el desarrollo de software en México*

En esta actividad se realizó una encuesta a un conjunto de empresas de desarrollo de software en México para conocer el uso y aplicación de métodos ágiles de los procesos de desarrollo en empresas mexicanas. Esta actividad se encuentra detallada en el Capítulo 2 sección 2.2.

*3. Analizar herramientas para evaluar el uso de métodos ágiles en las PYMES*

A partir de los resultados de la revisión sistemática se identificaron herramientas existentes de evaluación de métodos ágiles, por lo que se desarrolló una búsqueda y análisis de información, y posteriormente se identificaron las herramientas existentes que cubren los aspectos necesarios para la implementación o adopción de un método ágil pero no para ambos. Esta actividad se encuentra detallada en el Capítulo 2 sección 2.3.

*4. Desarrollar un método de evaluación de métodos ágiles*

Se define un método de evaluación de métodos ágiles que permita evaluar a una organización, para definir el grado de utilización de métodos ágiles, el método desarrollado se explica en este Capítulo 3 en la sección 3.4.

*5. Desarrollar una herramienta de evaluación basado en el método propuesto*

Para implementar el método de evaluación de métodos ágiles se desarrolló un prototipo de herramienta el cual permitirá a las organizaciones medir el grado de uso de métodos ágiles (evolución) de forma automatizada y obtener gráficos y/o recomendaciones que le permitan mejorar en el uso del método ágil. La herramienta se explica en el Capítulo 3 sección 3.5.

#### *6. Validar la viabilidad del método propuesto*

Se desarrolló un estudio de caso en empresas para validar la viabilidad del método y su herramienta de implementación. La validación realizada se muestra en el Capítulo 4 de este documento de Tesis.

## **3.2 Desarrollo del método**

En esta sección se detalla el método de desarrollo propuesto para la evaluación del uso de métodos ágiles.

### **3.2.1 Introducción**

De acuerdo a Highsmith en (C. Patel et al., 2009), “la agilidad es la habilidad tanto para crear como para responder a cambios con la finalidad de obtener ganancias en entornos turbulentos”. Por lo tanto, los métodos ágiles definen cómo debe ser el desarrollo del software siguiendo los valores y principios ágiles (M. Llana et al., 2013), así como, los principios definidos en el manifiesto ágil (C. Patel et al., 2009) respondiendo así al desarrollo de software rápido y siendo capaces de enfrentar problemas que se presentan a lo largo de las etapas del desarrollo de software.

El método toma como base dos fuentes como referencia que son:

- Modelo de Madurez Ágil (AMM) (C. Patel et al., 2009), para el desarrollo de software ágil. En este modelo se muestra como las prácticas de desarrollo de software ágil maduran desde un nivel inicial basado en los principios y prácticas ágiles para la mejora continua, en el cual cada nivel tiene objetivos predefinidos para ayudar al enfoque de la organización de sus actividades de mejora.
- Marco de trabajo Scrum Checklist (G. Boris, 2012), para guiar y ayudar al usuario en las reglas para incorporar y adaptar el método ágil en un ambiente productivo con el equipo de trabajo.

### 3.2.2 Fundamentos del método

Para realizar el método se ha identificado que para una correcta evaluación de un método ágil se debe de tener en cuenta tanto las habilidades formales como el comportamiento respecto al uso del método ágil utilizada en la organización. Véase Figura 21.

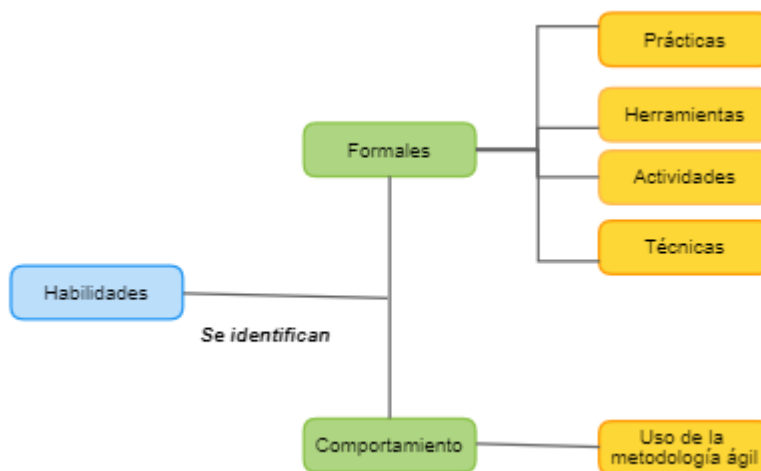


Figura 21. Habilidades que se identifican.

Por lo tanto, el método propuesto está formado por dos etapas principales:

- Evaluación: El objetivo es evaluar el estado actual del uso de métodos ágiles.
- Evolución: El objetivo es guiar en el uso y la implementación de métodos ágiles para mejorar las habilidades en el uso de herramientas y en la forma de aplicarlas en el ambiente organizacional.

Para la evaluación de cada una de las etapas se han definido cinco niveles, los cuales permitirán conocer el nivel de madurez del conocimiento en la aplicación e implementación de métodos ágiles.

El método propuesto tiene como propósito evaluar el uso de métodos ágiles utilizadas en el proceso de desarrollo de software. Véase Figura 22.

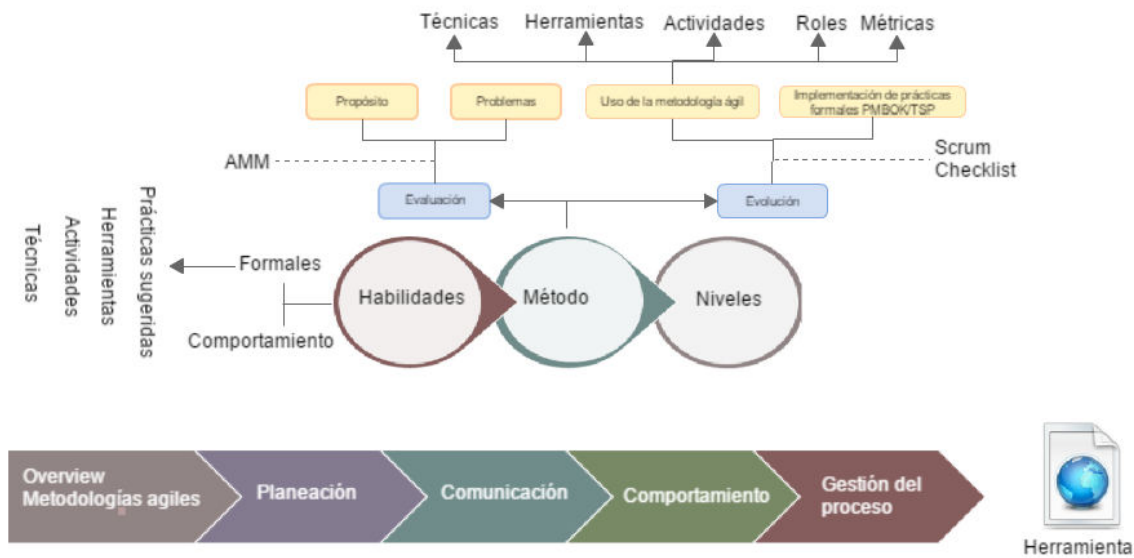


Figura 22. Esquema de la metodología propuesta.

Como se observa en la Figura 22, el método está compuesto por 5 niveles los cuales representan el grado de madurez de la organización. A continuación, se explican los niveles propuestos:

**Nivel 1:** El propósito de este nivel es reconocer y/o guiar al usuario en el “buen” camino de la agilidad. Los usuarios que se encuentran en este nivel desconocen las prácticas ágiles y las actividades se realizan de forma empírica.

**Nivel 2:** El propósito de este nivel es establecer la comunicación de todos los miembros involucrados en el desarrollo de software, así como establecer la planeación total del proyecto. Los usuarios que se encuentran en este nivel tienen problemas relacionados con la comunicación, código, integración, retención del personal, estándares, excesos de tiempos y cliente insatisfechos.

**Nivel 3:** El propósito de este nivel es ayudar a identificar y mejorar los problemas relacionados con el cliente y reforzar la relación que se tiene con el cliente. Los usuarios que se encuentran en este nivel tienen problemas relacionados con la reducción de control y la falta de estándares y prácticas establecidas.

**Nivel 4:** El propósito de este nivel es ayudar al equipo de desarrollo o administradores a identificar y mejorar los problemas relacionados con el equipo y su organización, a través de una evaluación. Los usuarios que se encuentran en este nivel pueden tener problemas relacionados con el registro y uso de datos históricos, la gestión del proyecto, la evaluación de riesgos y la generación de métricas.

**Nivel 5:** El propósito de este nivel es Mejorar el contexto, administrar la incertidumbre, revisiones de los resultados de los proyectos sincronizados y la prevención de defectos. Los usuarios que se encuentran en este nivel pueden tener problemas relacionados con la gestión de información histórica.

A continuación, se define a detalle el método propuesto.

### 3.2.3 Fase de Evaluación

En esta etapa se evalúa el estado actual sobre el uso y aplicación los métodos ágiles, enfocándose en el propósito de cada nivel y los problemas presentados por la organización. Véase Figura 23.

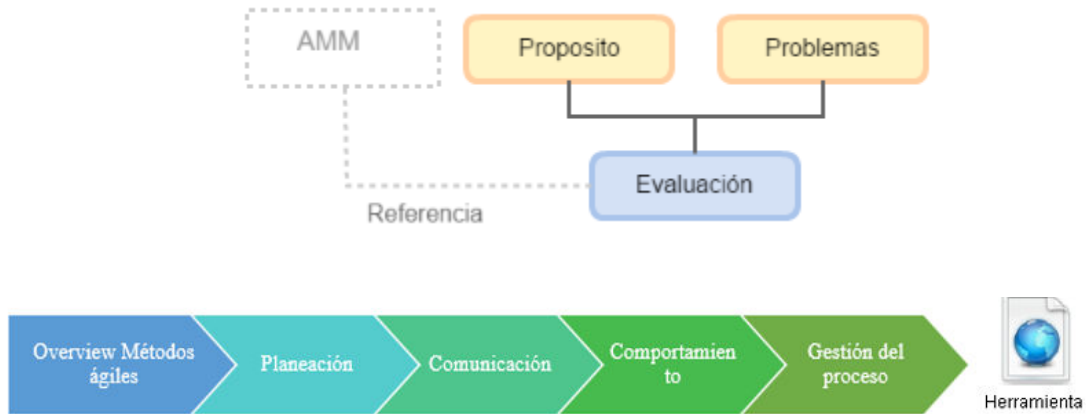


Figura 23. Evaluación del uso de métodos ágiles.

### 3.2.4 Fase de Evolución

Basado en los resultados de la evaluación, en esta etapa se refuerzan las habilidades para la aplicación de métodos ágiles, con lo cual se evoluciona en el su uso del método ágil a través de la sugerencia de técnicas y/o herramientas, actividades y roles. Esto se realiza proporcionando una guía de la correcta incorporación de métodos ágiles en la organización. Véase Figura 24.

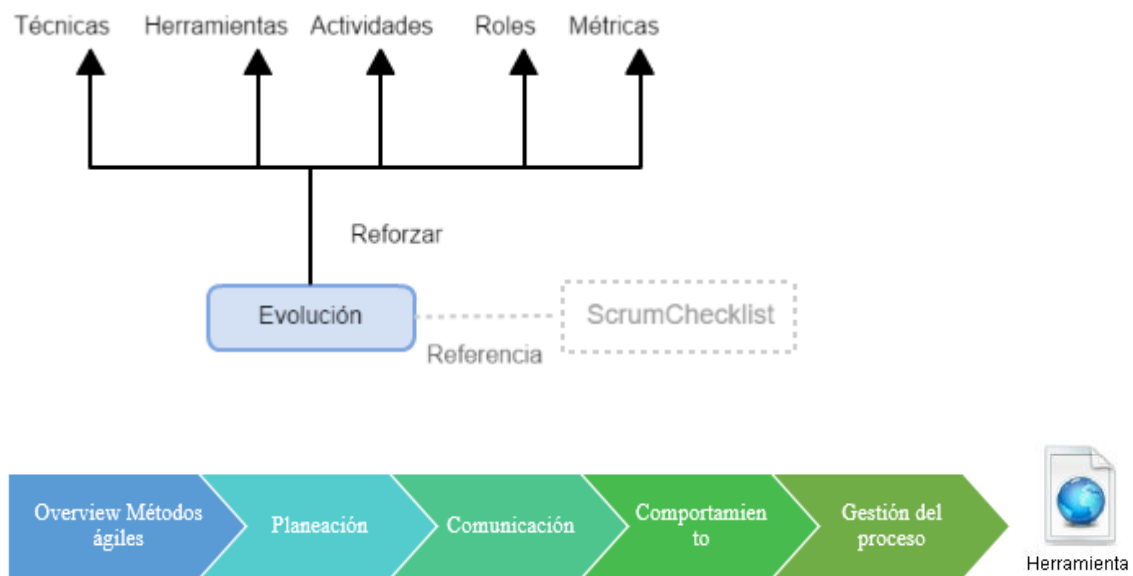


Figura 24. Evolución en el uso de métodos ágiles.



### 3.3 Descripción detallada del método

En esta sección se definen a detalle los niveles del método.

#### 3.3.1 Estructura para descripción general de los niveles del método

Los niveles del método se han definido siguiendo la estructura propuesta en el trabajo de investigación [Muñoz, et al. 2010]. Los elementos seleccionados para la descripción del método son:

- **Propósito:** sentencia clara y concisa que describe el objetivo de la fase.
- **Diagnóstico:** proporciona apoyo para diagnosticar el nivel en el que se encuentra una empresa, describiendo un conjunto de propósitos, problemas y los objetivos a evaluar de cada nivel.
- **Actividades:** etapas intermedias del proceso organizadas ordenadamente.
- **Roles:** describe la función de las personas que intervienen en las actividades de la fase.
- **Técnicas:** material que describe protocolos, normas o un conjunto de reglas para la realización de las actividades de la fase.
- **Herramientas:** material de apoyo para la realización de las actividades de la fase.
- **Productos de trabajo:** son los entregables que van a ser generados por una fase, las cuales están determinadas por dos tipos: a) Salidas: Productos o servicios que resultan de una fase, y b) Salidas intermedias, entradas de otras fases del proceso.

- **Métricas:** medidas definidas para medir los resultados y permitir un avance entre niveles.
- **Contexto:** es la manera de reconocer cada nivel del uso del método en la práctica.
- **Actividades del usuario en la herramienta:** describe las actividades que el usuario debe realizar como parte de la evaluación en la herramienta.

### 3.3.2 Elementos comunes

Los elementos comunes son aquellos que se repiten en todos los niveles propuestos, a continuación, se muestran:

- **Roles**

Los niveles van dirigidos a usuarios que tengan un rol Scrum Master o líder de proyecto, sin embargo, no es obligatorio tener un rol establecido (tener un rol establecido no es obligatorio).

- **Herramientas y técnicas**

- **Encuesta:** es un instrumento para la recopilación de información diseñado para poder cuantificar información. El nivel propone una encuesta que permitan extraer datos para conocer el estado del usuario con los métodos ágiles. Estas preguntas de la encuesta se encontrarán expresadas en tarjetas interactivas que permitan agilizar la respuesta del usuario, con una selección de la respuesta de la tarjeta con la pregunta que considere correcta.

- **Productos de trabajo:** es el nivel en el que el usuario se encuentra.

- **Métricas:** cada nivel equivale al 20 de 100 teniendo cinco niveles a evaluar, el cien por ciento de cada nivel está dado por respuestas correctas establecidas, a continuación, se muestra la fórmula:

1.  $\%TotalObtenidoNivel=$

$$(Número\ de\ respuestas\ correctas / Total\ de\ preguntas) * 100$$

2.  $\%ValorNivel=(TotalObtenidoNivel/100) * ValorNivel$

*ValorNivel: Es el valor del nivel, el cual es 20.*

Los porcentajes ideales para avanzar de un nivel a otro es de:  $\geq 70\%$ , ya que se necesita conocer las bases del método ágil para continuar subiendo a los siguientes niveles.

### 3.3.3 Nivel 1: Overview

El nivel uno se centra en el conocimiento del usuario sobre los métodos ágiles, con este nivel se pretende reconocer y fijar el conocimiento del usuario en uno de los niveles sugeridos, o en su defecto inicializarlo en la implementación de métodos ágiles.

- **Contexto**

En este nivel las prácticas sobre algún método ágil son casi nulas y no se logra reconocer las actividades que ya se tienen establecidas (fortalezas y debilidades).

- **Propósito**

El propósito de este nivel es descubrir, si conocen los métodos ágiles y como las utilizan/aplican en su ambiente de trabajo.

- **Diagnóstico**

La Tabla 21 muestra el propósito, problemas y objetivos a evaluar para el nivel 1.

Tabla 21. Elementos para el diagnóstico del Nivel 1.

Propósito	Problemas	Objetivos a evaluar
<p><b>Reconocer y/o guiar al usuario en el “buen” camino de la agilidad.</b></p>	Falta de tiempo	1. Verificar que siga las reuniones.
	Desplazamientos de calendario	2. Verificar que exista un product backlog.
	Comunicación	3. Verificar que se tengan roles.
	Calidad	4. Verificar que los roles estén asignados.
	Costos de desarrollo	5. Existe Sprints (en actividades, habilidades, técnicas) 6. Qué resultados han obtenido en su esquema de desarrollo.

- **Actividades:**

- 1) Cuestionario de conocimiento base de métodos ágiles.
- 2) Cuestionario sobre cómo se aplica conocimiento de métodos ágiles.

### 3.3.4 Nivel 2: Planeación

El nivel dos se centra en el conocimiento del usuario sobre las técnicas o herramientas del equipo, con este nivel se pretende reconocer las técnicas que se aplican para comunicar al equipo las actividades y las herramientas que el equipo utiliza en el proceso de desarrollo de software en un proyecto, reconocer que técnicas utilizadas se aplican en los métodos ágiles.

- **Contexto:**

En este nivel las prácticas sobre algún método ágil son casi nulas, y no se logra reconocer las actividades que ya se tienen establecidas (fortalezas y debilidades).

- **Propósito:**

El objetivo de este nivel es descubrir cómo realizan las prácticas y que herramientas se utilizan en el equipo de desarrollo, así como los estándares que se utilizaron en los proyectos.

- **Diagnóstico:**

En este apartado se muestra la definición de los elementos base (columna problemas y propósito) y los elementos incorporados para la evaluación del avance por nivel. Véase la Tabla 22.

Tabla 22. Elementos para el diagnóstico del Nivel 2.

Propósito	Problemas	Se evalúa el uso de métodos ágiles
Comunicación de todos los miembros involucrados en el desarrollo de proyectos y la planeación total de proyectos.	Comunicación.	1. Verificar las reglas
	Integración.	de comunicación y los sprint.
	Retención del personal.	2. Verificar como realizan la
	Estándares de codificación.	planeación.
	Excesos de tiempo.	3. Verificar cuales si tienen reuniones.
	Clientes insatisfechos.	4. Verificar si se realiza sprint diario. 5. Existe Graficas de tareas. 6. Verificar si tienen un tablero de actividades.

- **Actividades:**

- 1) Evaluar elementos base que se enfocan en desarrollo y planeación del proyecto.
- 2) Evaluar elementos incorporados para la ejecución del método como: reglas de comunicación, sprint, reuniones, y el uso de artefactos como gráficas y tableros.

### 3.3.5 Nivel 3: Comunicación

El nivel tres se centra en la comunicación de los miembros del proyecto, llámese equipo y cliente, con este nivel se pretende reconocer y fijar el conocimiento del usuario para avanzar en los niveles sugeridos y apoyar en las mejores prácticas para la comunicación de los miembros.

- **Contexto:** En este nivel la organización reconoce que lo importante para mejorar es la comunicación de todas las actividades entre los miembros.
- **Objetivo:** El objetivo de este nivel es descubrir, como se lleva a cabo la comunicación entre el equipo y como las aplican en su ambiente de trabajo, identificando las formas de comunicación que son mantenidas e idénticas en un ambiente ágil.
- **Diagnóstico:** En este apartado se muestra la definición de los elementos base y los elementos incorporados para la evaluación del avance por nivel. Véase Tabla 23.

Tabla 23. Elementos para el diagnóstico del Nivel 3.

Propósito	Problemas	Se evalúa el uso de métodos ágiles
Ayudar a identificar y mejorar los problemas relacionados con el cliente y reforzar la relación que se tiene con él.	Relación cliente.	1. Verificar si se han tenido problemas con el cliente.
	Testeo.	2. Verificar si se han clasificado los problemas con el cliente.
	Liberaciones Frecuentes.	3. Verificar como han mantenido la

	Estándares de codificación.	<p>comunicación con el cliente.</p> <p>4. Verificar si el cliente mantiene una relación directa con el equipo.</p>
--	-----------------------------	--

- **Actividades:** Evaluar e identificar problemas relacionados con el cliente.

### 3.3.6 Nivel 4: Comportamiento

El nivel cuatro se centra en el conocimiento del usuario sobre los métodos ágiles, con este nivel se pretende reconocer el conocimiento del usuario con los métodos ágiles.

- **Contexto:** En este nivel la organización ya tiene establecidas las reglas, las actividades y su asignación, así como un canal de comunicación y de administración de proyectos, medición de riesgos y calendarios exactos.
- **Propósito:** El objetivo de este nivel es descubrir, como realizan las actividades de administración de proyecto con respecto a los métodos ágiles, así como su aplicación en su ambiente de trabajo.
- **Diagnóstico:** En este apartado se muestra la definición de los elementos base y los elementos incorporados para la evaluación del avance por nivel. Véase Tabla 24.



Tabla 24. Elementos para el diagnóstico del Nivel 4.

Propósito	Problemas	Se evalúa el uso de métodos ágiles
<p>Ayudar al equipo de desarrollo y administradores a identificar, mejorar los problemas relacionados con la administración del proyecto.</p>	<p>Gestión del proyecto.</p>	<p>1. Verificar como es la convivencia del equipo</p>
	<p>Auto evaluación del equipo.</p>	<p>2. Verificar como se realiza el seguimiento</p>
	<p>Gestión de riesgos.</p>	<p>3. Verificar si se cuenta con evaluación de riesgos.</p> <p>4. Verificar se realizan acciones correctivas.</p> <p>5. Verificar si hay reglas de comportamiento.</p> <p>6. Verificar si hay reglas de carga de trabajo.</p>

- **Actividades:** Evaluar cómo se lleva a cabo la administración del Proyecto ágil.

### 3.3.7 Nivel 5: Comportamiento Extendido

El quinto nivel se centra en el conocimiento del usuario sobre cómo almacenar y revisar los resultados de la información de los proyectos de desarrollo los métodos ágiles.

- **Contexto:** Con este nivel se pretende reconocer la forma en que se almacena la información de gestión de proyectos para que pueda ser consultada y mitigar riesgos con bases acerca de la experiencia ocurrida. En este nivel la organización mantiene y utiliza la información histórica.
- **Propósito:** El objetivo de este nivel es descubrir cómo almacenan la información de los proyectos de desarrollo tales como lecciones aprendidas, repositorios y los resultados.
- **Diagnóstico:** En este apartado se muestra la definición de los elementos base y los elementos incorporados para la evaluación del avance por nivel. Véase tabla 25.

Tabla 25. Elementos para el diagnóstico del Nivel 5.

Propósito	Problemas	Se evalúa el uso de métodos ágiles
Mejorar el contexto, administrar la incertidumbre, revisiones de resultados del	Gestión del proyecto.	1. Verificar si se realiza un almacenamiento histórico. 2. Verificar si todos tienen acceso al histórico.
	Auto evaluación del equipo.	
	Gestión de riesgos.	

proyecto sincronizados y la prevención de defectos.	Repositorios.	<ol style="list-style-type: none"> <li>3. Verificar si se registrar los errores históricamente.</li> <li>4. Verificar si se tienen registros de lecciones aprendidas.</li> <li>5. Verificar si se tienen repositorios.</li> <li>6. Verificar como realizan el almacenamiento.</li> <li>7. Verificar cuales y como son los resultados.</li> </ol>
---	---------------	--

- **Actividades:** Evaluar la administración de los resultados y registros históricos.

### 3.4 Desarrollo de la herramienta

En esta sección se describe el proceso de desarrollo de la herramienta, la cual sirve como soporte para la aplicación del método de evaluación propuesto. A continuación, se describe cada uno de los artefactos creados para el modelado y desarrollo de la herramienta.

### 3.4.1 Requisitos Funcionales

Para el desarrollo de software se definen requerimientos funcionales los cuales son requisitos que el usuario define como necesarios y que se espera que el sistema realice. Los requisitos se generaron de acuerdo a la experiencia que se quiere entregar al usuario y se definieron los roles principales para el sistema, los cuales definen los permisos que se tendrán sobre la herramienta. A continuación, se listan los roles principales. Véase Tabla 26.

Tabla 26. Tipos de usuario.

Tipo Usuario	Descripción
Usuario	Rol que permite entrar al sistema y jugar con la herramienta para evaluar su grado de agilidad por medio de los cuestionarios presentados en forma de juego.

Para determinar estos roles serían los principales, se pensó en una administración de la herramienta que permita adecuarse a los cambios presentados en cada organización o para modificación del contenido para mejora de los indicadores.

### 3.4.2 Diagramas de flujo: Interacción sistema-usuarios

A continuación, se presentan los diagramas de diseño que definen la estructura y la interacción del usuario con el sistema.

### 3.4.2.1 Interacción Administrador-herramienta

A continuación, se muestra la figura 25, donde se describe la gestión de la herramienta, el cual administra el cuestionario y los usuarios del sistema.

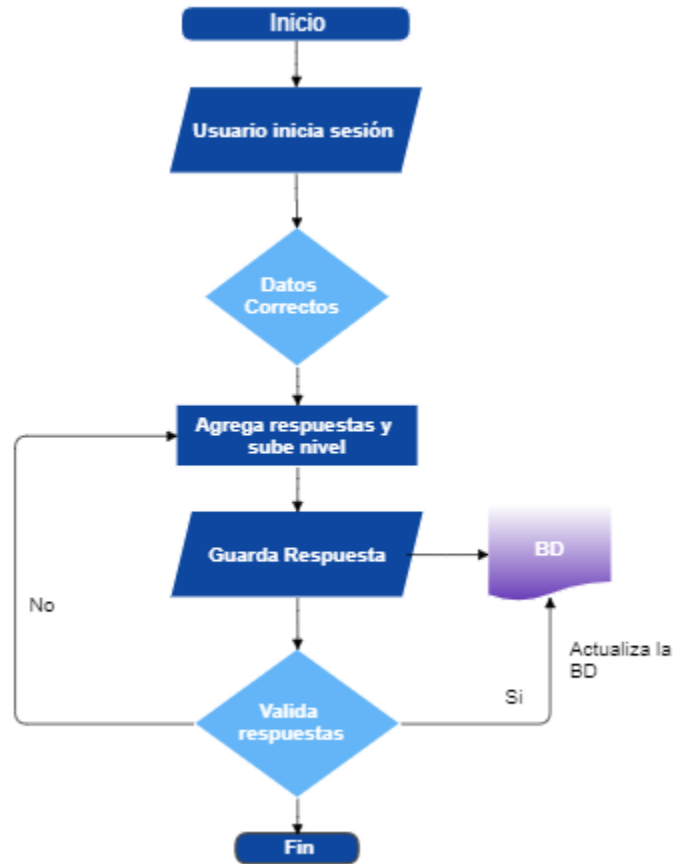


Figura 25. Interacción Administrador-Sistema.

### 3.4.2.2 Interacción Usuario-Sistema

El usuario ingresa al sistema para valorar su grado de agilidad. A continuación, se muestra el flujo del usuario con el sistema.

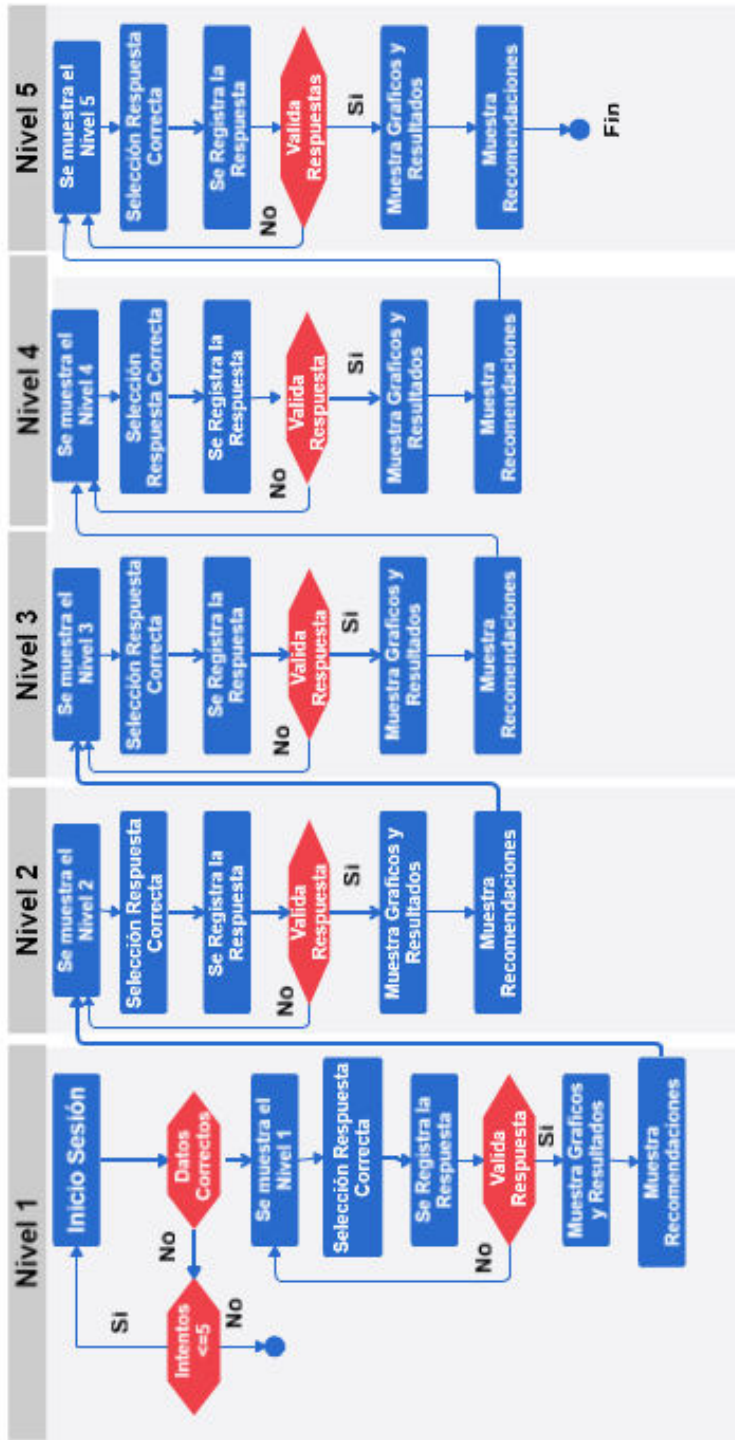


Figura 26. Interacción Usuario-Sistema-Nivel.

### 3.4.3 Diseño y Arquitectura de la herramienta

La herramienta está desarrollada en el lenguaje de programación PHP utilizado para desarrollo web de páginas dinámicas y que puede interactuar con otros lenguajes de programación web. PHP es el lenguaje más aplicado en páginas y portales web. Lenguaje de código abierto, lo que significa que cualquier usuario pueda aplicarlo en diversas aplicaciones.

Para la vista de usuario de la herramienta se utilizó latte, motor de plantillas basado en código PHP que facilita la interpretación de plantillas de diseño, así, como en la protección de datos al interactuar con la herramienta. Se muestra el diseño de la arquitectura de la aplicación. Figura 27.

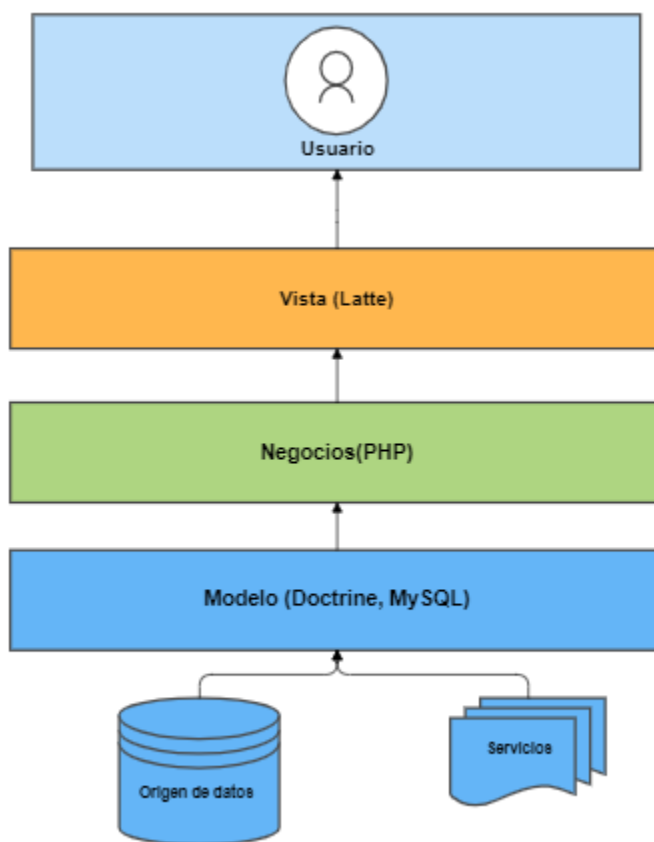


Figura 27. Arquitectura de la aplicación.

Para gestionar los datos de la aplicación se utilizó el lenguaje de base de datos MySQL con el servidor de base de datos PHP MyAdmin.

PHP gestiona la información por medio de una conexión única para el administrador MySQL y que interactúa por medio de una llamada de Entity framework la cual define un modelo conceptual, un modelo de almacenamiento y la asignación a través del ORM Doctrine que permite trabajar con objetos y clases de la aplicación. A continuación, se muestra el esquema de base de datos de la herramienta. Figura 28.

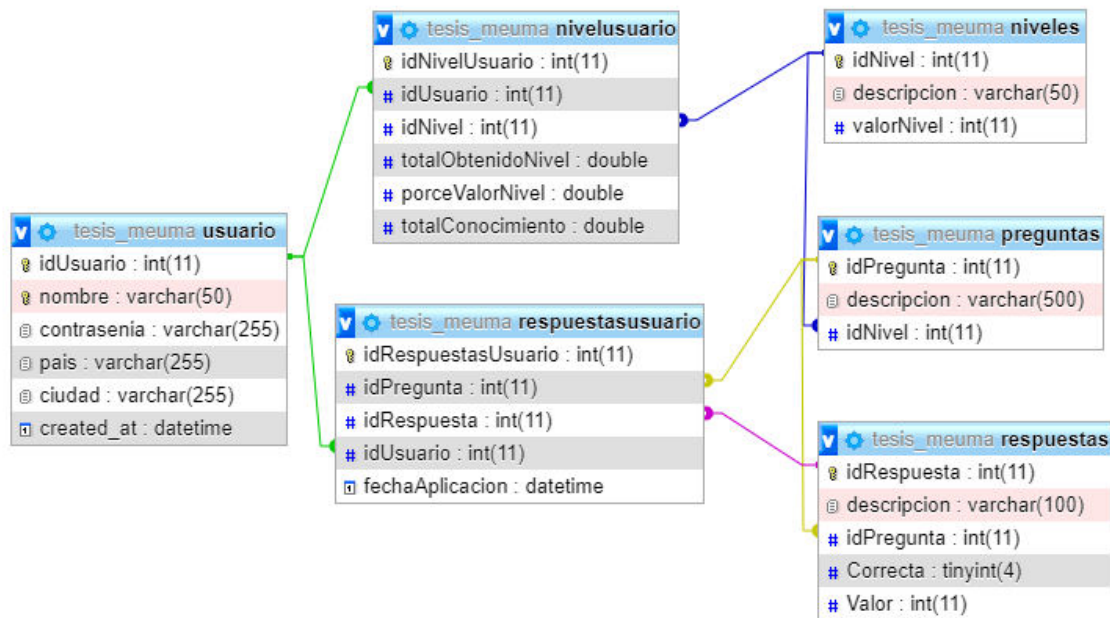


Figura 28. Esquema de BD.



### 3.4.4 Diseño de prototipo

A continuación, se presenta el prototipo del sistema como prospectos de los modelos de diseño, donde se visualizará la interacción con los elementos de la pantalla.

A continuación se muestra la página principal de la herramienta. Figura 29.

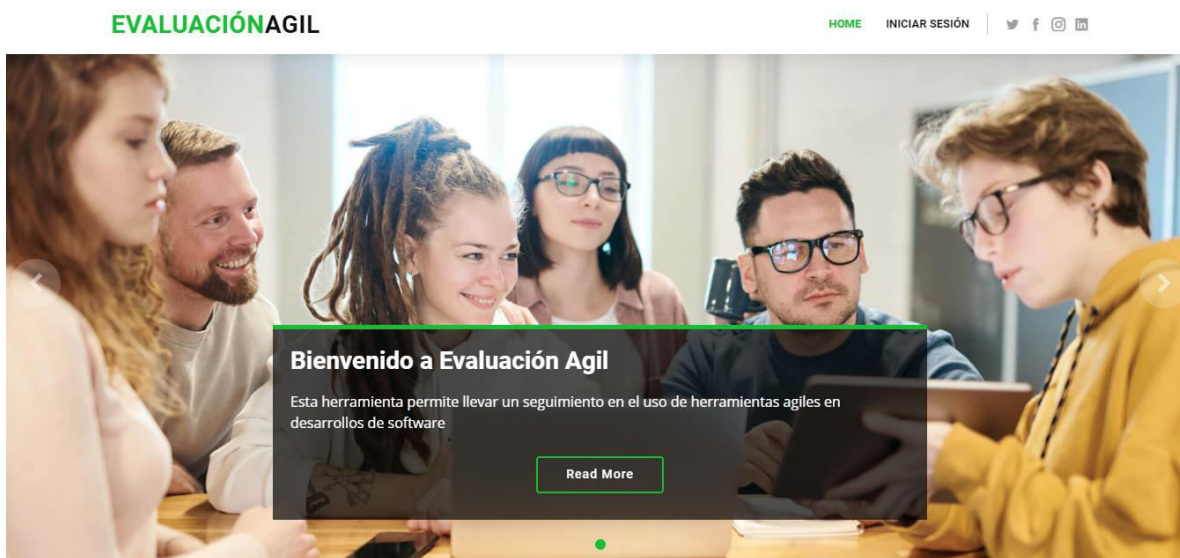


Figura 29. Página principal.

#### 3.4.4.1 Diseño de la Herramienta

En la siguiente sección se detallan las pantallas finales de diseño de la herramienta propuesta.

Tabla 27. Actividades de usuario.

Actividades de Usuario	Descripción de la Actividad
Registro	El usuario deberá haberse registrado en el sistema antes de realizar cualquier nivel. Este registro se realiza en el módulo de registro e inicio de sesión.
Encuesta	El usuario deberá seleccionar la respuesta que crea correcta dentro de las tarjetas que se presentan en el sistema.
Validación	El usuario debe superar nivel a nivel para acceder a cada uno de ellos.

Producto de las Actividades	Descripción de las actividades del producto
Resultados	el sistema muestra los resultados con el nivel marcado de acuerdo a las respuestas recolectadas en la encuesta.
Recomendaciones	<b>Reglas:</b> (1) El usuario deberá seleccionar por lo menos de una respuesta, si no es terminada la encuesta, no se considerará como terminada y no podrá continuar al siguiente nivel; (2) Para avanzar entre los niveles el usuario deberá superar un nivel a la vez; y (3) Los niveles se muestran conforme se avanza entre los niveles.

### 3.4.4.2 Pantalla Inicio de Sesión

A continuación, se muestra la pantalla de inicio de sesión del sistema para ingresar a los niveles. Véase Figura 30.



Usuario

Contraseña

Recuérdame [Olvidaste la contraseña?](#)

INICIAR SESIÓN

Figura 30. Pantalla inicio de sesión.

#### 3.4.4.3 Pantalla registro de un nuevo usuario

A continuación, se muestra la pantalla de registro de un usuario nuevo para acceder al sistema. Véase Figura 31.

# Registro de usuario

Nombre

Contraseña

Ciudad

Pais

No soy un robot

reCAPTCHA  
Privacidad · Términos

Regresar Guardar

Figura 31. Pantalla de registro de nuevo usuario.

## 3.4.4.4 Pantalla para los niveles

Las pantallas que se muestran a continuación, son las pantallas que representan en cada uno de los niveles. Véase Figura 32.



Figura 32. Pantalla de niveles.

### 3.4.4.5 Vista de Resultados

A continuación, se muestra la pantalla de la vista de los resultados de calificación de los niveles. Véase Figura 33.



Figura 33. Pantalla gráfica de resultados.

### 3.4.4.6 Videos de recomendaciones

A continuación, se muestra la pantalla de la vista de las recomendaciones de calificación de los niveles. Véase Figura 34.

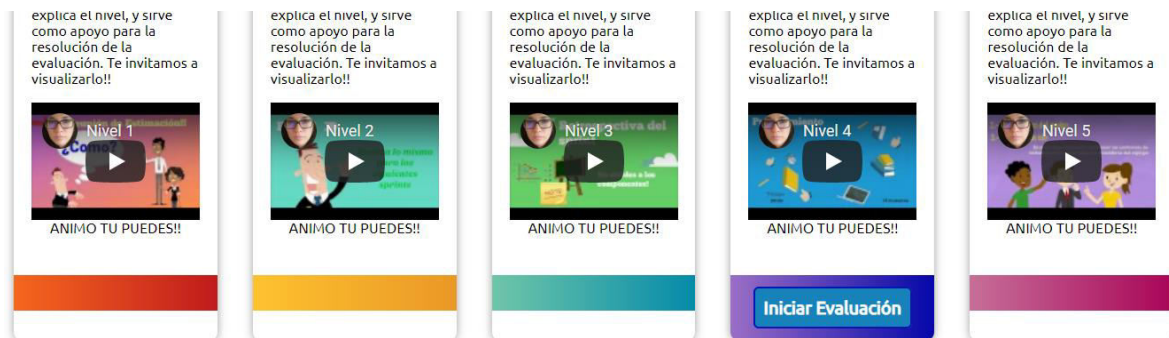


Figura 34. Pantalla recomendaciones.

# Capítulo 4. Desarrollo del estudio de caso

De acuerdo a (Runeson & Host, 2008), un estudio de caso es un método empírico que se basa, en que las características importantes tienen que ser flexibles para acoplarse a la dinamicidad de las propiedades cambiantes del mundo real, y que permiten concluir claramente las pruebas de manera cualitativa o cuantitativa, las cuales deben de ser planificadas basadas en teorías establecidas.

Para el desarrollo del estudio de caso se utilizaron las actividades establecidas según (Runeson & Host), A continuación, se muestran dichas actividades. Véase Figura 35.



Figura 35. Pasos del proceso de un estudio de caso.

## 4.1 Diseño del estudio de caso

El estudio de caso para este análisis se planifican objetivos, se establece un conocimiento actual, preguntas de investigación y métodos de recolección de datos.

### 4.1.1 Objetivo del estudio de caso

Aplicar método propuesto en organizaciones o a individuos que deseen conocer su madurez en la implementación y uso de los métodos ágiles, para conocer la viabilidad del método aplicado a través de la herramienta.

### 4.1.2 Conocimiento Actual

Debido a la creciente importancia del software para la resolución de problemas de distintos dominios, se observa un gran interés de las PYMEs relacionado a la adopción de métodos ágiles, con la finalidad reducir los tiempos para desarrollar el software. Sin embargo, la mayoría de las veces, se continúa con los problemas en el desarrollo de software, reflejado en la baja calidad de los productos.

Esto se debe a la carencia de conocimiento de las PYMES para la correcta implementación de estos métodos. Normalmente, la madurez de una empresa en el uso de un método ágil, únicamente se mide con base en: los años que lleva la empresa usando ese método, el nivel de los individuos que lo usan y, el tamaño de los proyectos a los cuales se aplica.

Acerca del conocimiento actual sobre las herramientas y/o investigaciones realizadas actualmente véase el capítulo 2.

### 4.1.3 Marco conceptual

El método propuesto está enfocado en evaluar el cómo se usan y aplican los métodos ágiles en una organización dedicada al desarrollo de software y donde los miembros del equipo se involucran en todos los aspectos del trabajo de desarrollo como lo marca la agilidad. Los individuos a los que está enfocada la herramienta son todos aquellos individuos que tienen perfil de líder y que ha dirigido equipos de desarrollo de software y que tiene experiencia acerca de los métodos ágiles.

Cuando se evalúa la herramienta por los individuos de prueba se espera que conforme avancen el nivel adquiera conocimientos sobre los métodos ágiles, a continuación, se muestra que se espera adquirir conforme se avance en los niveles. Véase Figura 36.



Figura 36. Aspecto evaluado en la herramienta.

- Principios ágiles y conocimiento: en este aspecto se evalúa la comunicación constante que se debe mantener con todos los involucrados del equipo de desarrollo, así como la realización de los planes con el equipo, las reuniones y los cambios sugeridos por el cliente.
- Adaptabilidad: en este aspecto se evalúa las actividades realizadas por el equipo, los cambios constantes o repentinos del cliente, así como la adaptación del equipo a las reuniones que se necesiten para lograr la agilidad.
- Incorporación: en este aspecto se evalúa las actividades adecuadas con técnicas, herramientas que apoyen a llevar a cabo el desarrollo de software dentro del marco de la agilidad, además de diseñar las pruebas pertinentes para el desarrollo.
- Administrable: en este aspecto se evalúa las actividades, las métricas y los roles que son necesarios para el desarrollo de software y que son definidas en el marco de la agilidad,



además de generar el plan necesario para las liberaciones de software y los planes para la ejecución de pruebas.

- Optimización: en este aspecto se evalúa las revisiones constantes y la retroalimentación acerca de las entregas con el cliente, además de generar planes de proyección sobre el proyecto acerca de los cambios solicitados, generar un almacenamiento de la información.

#### 4.1.4 Objeto de Estudio

A continuación, se muestra la información correspondiente al estudio de caso.

##### 4.1.4.1 Empresas pilotos

El estudio de caso fue aplicado a un grupo de profesionistas que tienen relación con tecnologías y que se encuentran involucrados en metodologías de software. A continuación, se muestra en la Tabla 28 la información del grupo de profesionistas que validaron la encuesta realizada.

Tabla 28. Empresas pilotos.

<i>Usuario</i>	<i>Tipo de puesto</i>	<i>Experiencia con Ágiles</i>
1	Empleado.	Ninguna.
2	Coordinador de desarrollo.	Ninguna.
3	Software engineer.	Scrum.
4	Director de proyectos.	Scrum.
5	Consultor.	Kanban.
6	Outsourcing.	Scrum.
7	Gerente.	Scrum.

8	Arquitecto JR.	Scrum.
9	Programador.	Lean Kanban.
10	Programador.	Kanban.
11	Software engineer.	Kanban.
12	Director de proyectos.	Cascada.
13	Programador.	Scrum.
14	Outsourcing.	Kanban.
15	Programador.	Lean Kanban.

Cabe resaltar que, para aplicar el estudio de caso se realizó una presentación de la herramienta acerca de su uso y los resultados que se muestran en cada nivel.

#### 4.1.4.1 Construcción de la muestra

El método propuesto está enfocado en individuos que tengan un perfil de líder, desarrollador o Scrum master, ya que se consideran parte primordial en el desarrollo de software y la conexión con el equipo y todos los involucrados, por lo cual se propone algunas inferencias en la aplicación del estudio de caso y mejorar la comprensión de la herramienta se realizaron ejemplos de casos ideales y casos extremos para alcanzar los niveles ideales en el método propuesto.

Inferencias de ayuda: (1) si el individuo no tiene conocimientos acerca de los métodos ágiles, puede realizar todas las pruebas de los niveles que necesite para alcanzar el porcentaje ideal y avanzar al siguiente nivel; (2) si necesita ver los videos introductorios podrá realizarlo cuando lo desee mientras descubre cada nivel; y (3) al finalizar el nivel se muestra automáticamente el siguiente nivel a evaluar.

#### 4.1.4.2 Preguntas de investigación

El objetivo del estudio del caso determina las preguntas de investigación, la hipótesis y problemática planteada en la sección 1.1.9 del capítulo 1. A continuación, se muestran las preguntas de investigación.

P1. ¿Es adecuado el diagnóstico de la herramienta propuesta acerca del conocimiento en el uso y aplicación de métodos ágiles?

P2. ¿Cuál es el grado de conocimiento adquirido sobre métodos ágiles?

P3. ¿Se adquiere la habilidad en el uso de métodos ágiles con mayor facilidad con el uso del método propuesto?

## 4.2 Preparar la recolección de datos

Para recolectar los datos se utilizó una encuesta desarrollada en la herramienta de Google Docs, la cual será enviada al usuario al momento de terminar la evaluación en la herramienta propuesta la cual se muestra en la tabla 29. Se definieron una serie de pasos para la ejecución del estudio de caso, los cuales se describen a continuación.

- 1) Sección-guía introductoria de la herramienta.
- 2) El individuo prueba utiliza la herramienta.
- 3) El individuo prueba contesta una encuesta para validar la herramienta.
- 4) Se analizan los resultados de la herramienta.
- 5) Se analizan los resultados de las encuestas.

6) Se establecen las conclusiones.

#### 4.2.1 Objetivos de mejora

Para cumplir el objetivo propuesto, se realizó una demostración previa, antes de ejecutar el estudio de caso con los individuos prueba, esto para evitar los mínimos problemas posibles al momento de la realización. Durante la demostración se ejecutó lo siguiente:

- Realizar una reunión.
- Presentación de la herramienta.
- Presentación de la aplicación de método propuesto.
- Recopilación de información para el ajuste de la herramienta.

Se recopilaron los detalles de la demostración y se analizaron para evaluar los efectos que producen los cambios sugeridos, se efectuaron los cambios y las adecuaciones necesarias para el cumplimiento del objetivo de la herramienta.

#### 4.2.2 Encuesta de validación

Se ejecutó un conjunto de preguntas con el fin de validar la efectividad de la herramienta, en la tabla 29 se encuentra las preguntas formuladas.

Tabla 29. Encuesta para recolección de información.

Preguntas de evaluación	Preguntas de investigación
<b>Aspectos Generales</b>	
1.- ¿Dónde te encuentras ubicado?	N/A
2.- ¿Puesto en el que se desempeña?	N/A
3.- ¿Cuál es el tipo de empresa donde labora?	N/A
4.- ¿Qué metodologías y/o modelos manejas para asegurar la calidad del software? (ejemplo: Cascada, scrum, Kanban, cmmi etc)	N/A
<b>Preguntas relacionadas con el método: Evaluación</b>	
5.- ¿Hasta qué nivel llegaste en la herramienta?	P1
6.- ¿Consideras que te ayudo a conocer más las metodologías ágiles?	P2
7.- ¿Has adquirido conocimiento valioso acerca de las metodologías ágiles?	P3
<b>Preguntas relacionadas con el método: Evolución</b>	
8.- ¿Considera que los videos son fáciles de comprender?	P1, P2
9.- ¿Los videos aportan información relevante para mejorar tu habilidad en el uso y la aplicación ágil?	P1, P2
10.- ¿Qué te pareció la herramienta?	P3
11.- ¿Éstas de acuerdo con el diagnostico que se te presenta?	P3, P2
12.- ¿Éstas de acuerdo con los resultados del diagnóstico? (Si/No, ¿Por qué?	P3, P2

### 4.3 Recolección de datos

Para recopilar la información se utilizaron una serie de métricas, una vez que se ejecutó la evaluación con el usuario, las cuales fueron registradas posteriormente.

### 4.3.1 Métricas

Para la generación del análisis de resultados se probaron medidas, las cuales están basadas en dos de los principales objetivos: la evaluación acerca del conocimiento del usuario y la evolución que el usuario realice cuando obtenga nuevas habilidades o mejore sus conocimientos en métodos ágiles.

**Evaluación:** Métrica establecida a través de una herramienta de encuesta basada en el método propuesto, la cual permite conocer el porcentaje de conocimiento por cada nivel. A través de cinco gráficas que muestran dichos porcentajes basados en los cinco aspectos, lo que permite analizar la información de la evaluación.

**Evolución:** Métrica establecida por medio de dos variables:

Grado de conocimiento adquirido: el cual se evalúa con las preguntas cinco, seis y siete que se presentan en la evaluación; e hipótesis (la evaluación de la hipótesis se compone con las preguntas relacionadas tanto a la evaluación como a la evolución del cuestionario, que dice: que te ha parecido la herramienta).

A continuación, se muestra las gráficas obtenidas de la información ingresada por los participantes en la herramienta.

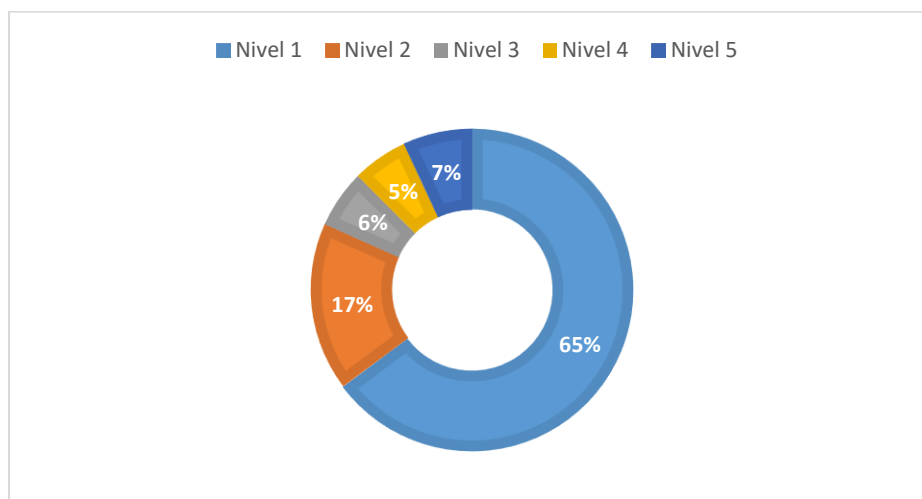


Figura 37. Total, de conocimiento por nivel.

El gráfico anterior muestra el promedio total de conocimientos obtenidos por los usuarios en cada nivel, este dato es obtenido de acuerdo al valor de la respuesta dada, de acuerdo a los niveles:

donde se muestra que el 65% de los participantes cuentan con conocimientos básicos en métodos ágiles, así como se demuestra que han realizado practicas sobre las áreas claves para la correcta implementación y ejecución de los principios ágiles. Y que el 7% de los participantes han realizado actividades ágiles que sirvan a la implementación optima del proceso ágil.

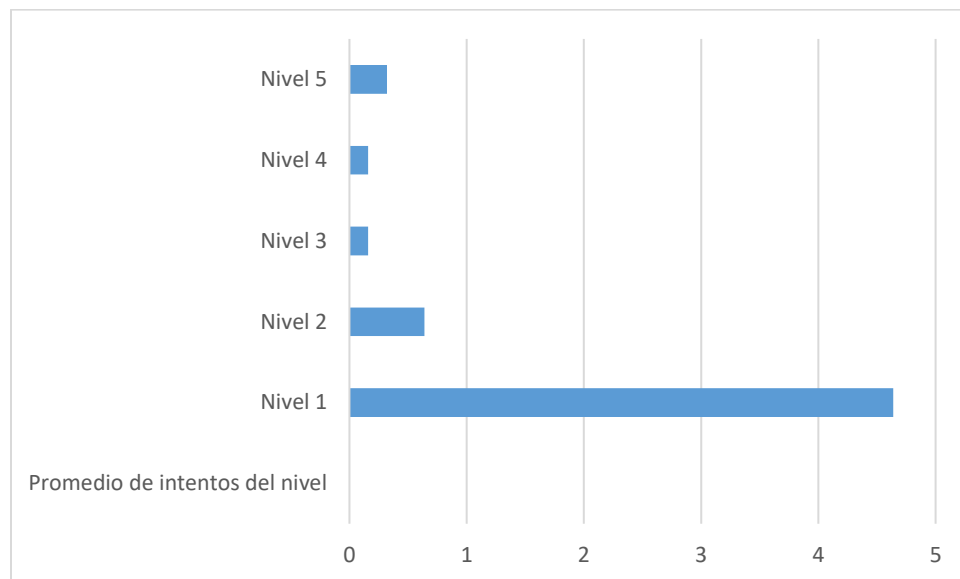


Figura 38. Promedio de intentos por nivel.

El gráfico anterior muestra el promedio total de intentos realizados por los participantes en cada nivel, este dato se obtiene al contabilizar el número de veces que el usuario realizo el nivel. En promedio los participantes realizaron más de 4 intentos en su mayoría en el nivel 1.

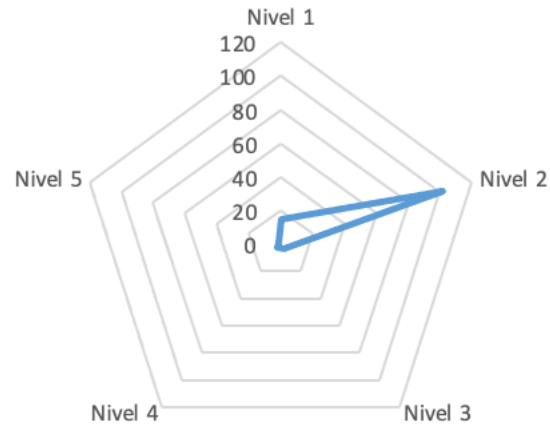


Figura 39. Total, obtenido por niveles.

En el gráfico anterior se muestra el grado de habilidad adquirido: el cual se evalúa con las preguntas ocho, nueve y diez que se presentan en la evaluación, se muestran los gráficos acerca de la evaluación. En la gráfica anterior se muestra que los conocimientos adquiridos acerca de métodos ágiles se da entre los participantes en el nivel 1 y 2, lo cual representa que en los niveles consecuentes se cuenta con poca experiencia en la aplicación de métodos ágiles.

## 4.4 Análisis de datos

Una vez que se ejecutó el estudio de caso y se analizó la información, se graficaron los datos de los participantes, los resultados obtenidos por nivel.

### 4.4.1 Descripción de las inferencias de diseño

A través de un gráfico radial presentado en la recolección de datos, analizando los cinco aspectos principales en cada nivel, donde se muestra el porcentaje calculado por las métricas descritas en el capítulo 3 sección 3.4.



- **Apoyo para la preparación de datos:** se generan los datos de los gráficos con la información recopilada del participante.
- **Interpretación de los datos:** la herramienta muestra el porcentaje obtenido de las respuestas, así como el total del nivel.
- **Repetible:** Opción para repetir el nivel alcanzado.

#### 4.4.2 Adoptar las inferencias

Las inferencias explican las posibles adopciones y cuáles son los datos necesarios para las validaciones internas de las métricas generadas por cada nivel.

- **Inferencias Casuales:** influencia de muestreo, es decir de los resultados de la población total; Individuos pruebas. Esta directamente influenciada por la característica de que los participantes sean ingenieros en sistemas o cuenten con amplio conocimiento en tecnologías y/o administración de proyectos en TI.
- **Inferencias de arquitectura:** 1) análisis; de la arquitectura de la base de datos permitirá conocer la herramienta por completo, así como la estructura de los datos almacenados en la misma; 2) variación; existe variación en el conjunto de preguntas presentadas por nivel; y si se agrega más preguntas o modificarlas la variación está dada por la respuesta colocada; 3) abstracción; comparar los datos obtenidos con los datos que se tienen presentados en la encuesta realizada en el capítulo 2, para obtener un porcentaje final del conocimiento del uso e implementación de métodos ágiles.
- **Inferencia relacional:** 1) objetivos: una vez recopilada y analizada la información de la herramienta se conocerá si los objetivos fueron alcanzados; 2) Motivación: con la

información obtenida en la ejecución de la herramienta permitirá conocer el estado de las organizaciones y los individuos, lo cual permitirá dar a conocer el estado de gestión de proyectos ágiles de desarrollo de software y de las empresas en este ramo; de manera que se impulsen nuevas estrategias para que las organizaciones de este sector crezcan en México.

## 4.5 Resultado del análisis

En el análisis de datos se generaron gráficas que apoyan al entendimiento de percepción de los individuos prueba con respecto a la herramienta, se muestran los resultados más relevantes de la encuesta de satisfacción:

En la siguiente figura 40, Se muestra con un porcentaje del 27% de validación de la herramienta, esto quiere decir que los individuos prueban se centran en la ciudad de México.

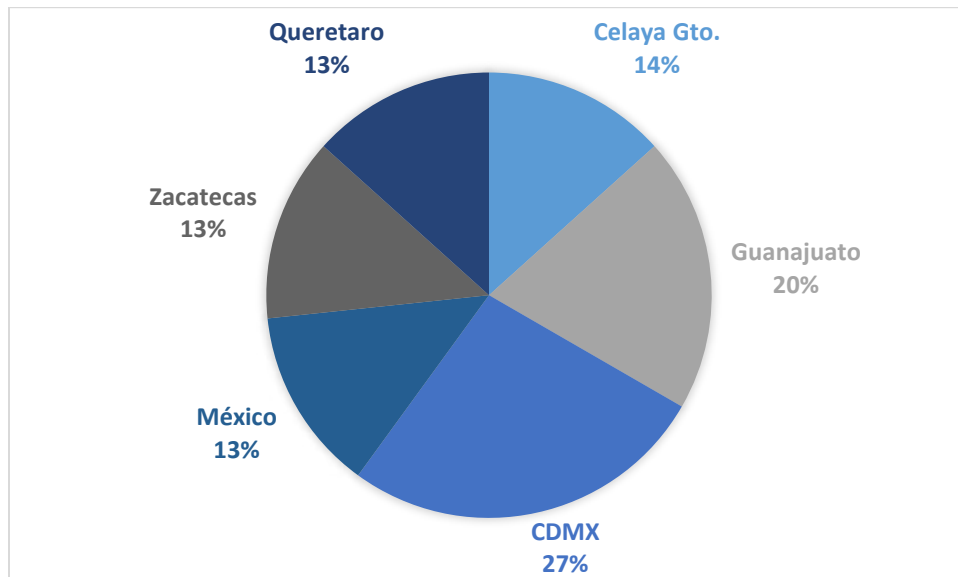


Figura 40. Ubicación de los participantes.

1.- *¿Consideras que te ayudo a conocer más los métodos ágiles?*

Respecto al conocimiento adquirido en la interacción con la herramienta es favorable el 85% de los individuos pruebas, reconocen que ayudó a confirmar su conocimiento, el 15% restante considera que necesitan más de una aplicación de cada nivel para realizar un aprendizaje.

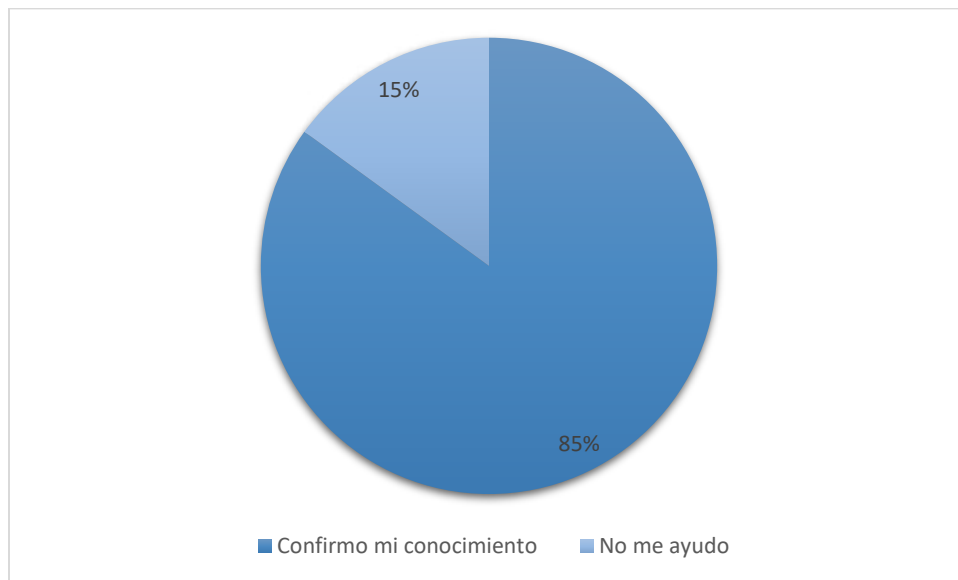


Figura 41. Resultados de la pregunta 1.

2.- *¿Has adquirido conocimiento valioso acerca de los métodos ágiles?*

De acuerdo a las respuestas sobre la interacción con la herramienta, la percepción sobre si es de ayuda y genera un conocimiento valioso, donde el 87% de los individuos prueba, reconocen que ayudó a confirmar su conocimiento, el 8% y el porcentaje restante considera que las preguntas son básicas y que no necesariamente permitirían alcanzar una evaluación acertada ya que no existe una respuesta correcta ante lo cuestionado, más bien todo ocurre en la aplicación y el cómo se afronta el problema a resolver.

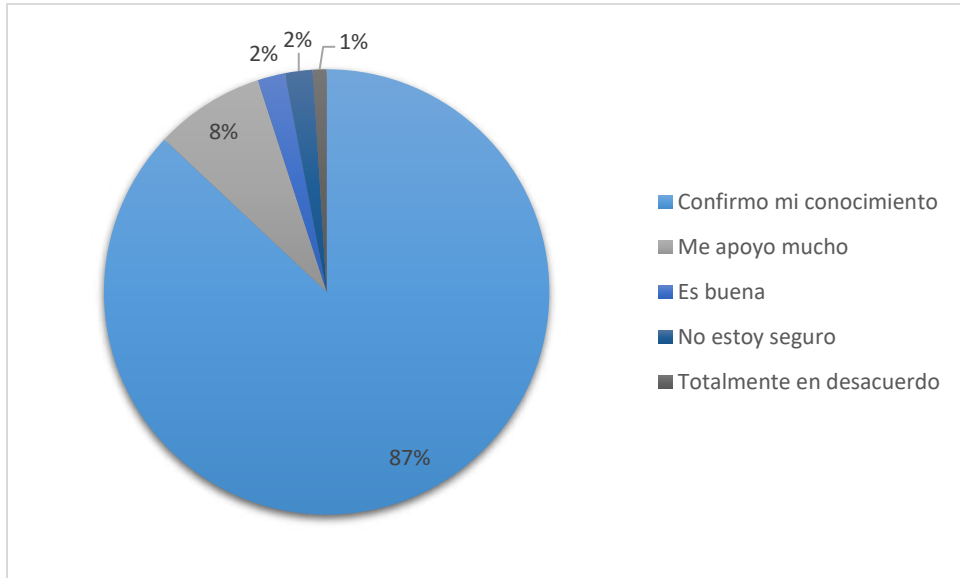


Figura 42. Resultados de la pregunta 2.

3.- *¿Los videos aportan información relevante para mejorar tu habilidad en el uso y la aplicación ágil?*

De acuerdo a las respuestas sobre la interacción con la herramienta, la percepción sobre si es de ayuda los recursos audiovisuales que permiten generar conocimiento de métodos ágiles, donde el 74% de los individuos prueba, reconocen que ayudó a confirmar su conocimiento, el 23% restante considera que con apoyo audiovisual o sin él no tiene relevancia o simplemente no ayuda.

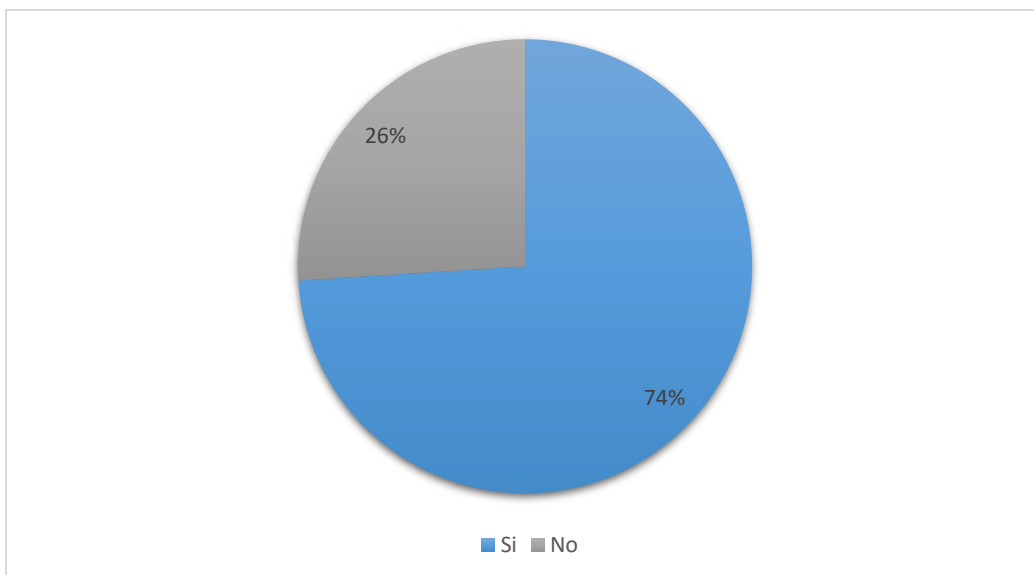


Figura 43. Resultados de la pregunta 3.

Además, los individuos de la muestra realizaron las siguientes observaciones con respecto a la herramienta. Las observaciones se muestran en la siguiente tabla.

Tabla 30. Observaciones de individuos de la muestra.

Elemento	Observaciones
Refuerzo Conocimiento	<ul style="list-style-type: none"> <li>• Recurso audiovisual aporta información relevante y sirvió para comprender los métodos ágiles.</li> <li>• Recursos audiovisuales apoyan al entendimiento del uso y aplicación de métodos ágiles.</li> </ul>
Mejora	<ul style="list-style-type: none"> <li>• Generar una introducción interactiva que indique cual es la finalidad de la herramienta.</li> <li>• Se reconoce que es una herramienta de capacitación.</li> <li>• Las respuestas a las preguntas deberían ser variadas.</li> </ul>

En la encuesta de satisfacción se señalan las siguientes fortalezas de la herramienta:

- ✓ Evaluación sencilla para que cualquier persona en cualquier nivel de tecnología realice.
- ✓ Es una herramienta que fácilmente puede utilizarse para capacitar equipos de trabajo.
- ✓ EL apoyo audiovisual mejora la capacitación.

#### 4.5.1 Resultados del estudio del caso

Para valorar el método y la herramienta desarrollados, se ejecutó un estudio de caso, del cual se obtuvieron los siguientes resultados:

Cada nivel es evaluado por cinco aspectos definidos en el modelo de madurez ágil AMM, como son conocimientos principios ágiles y conocimiento, como se adaptan los procesos conocidos de agilidad, como se incorporan los procesos ágiles, como administra los procesos ágiles, como se busca optimizar los procesos ágiles.

Esto permitirá conocer la debilidad o fortaleza del usuario en la aplicación o uso de los métodos ágiles. En el estudio del caso participaron 15 profesionistas con características de TI, a continuación, se exponen los resultados de la herramienta y de la encuesta realizada.

**Nivel 1 (Overview)**, Visión general acerca del conocimiento de cada individuo de la muestra acerca de métodos ágiles.

A continuación, se muestra el gráfico de las respuestas constantes aplicadas al grupo de individuos prueba para el nivel 1:

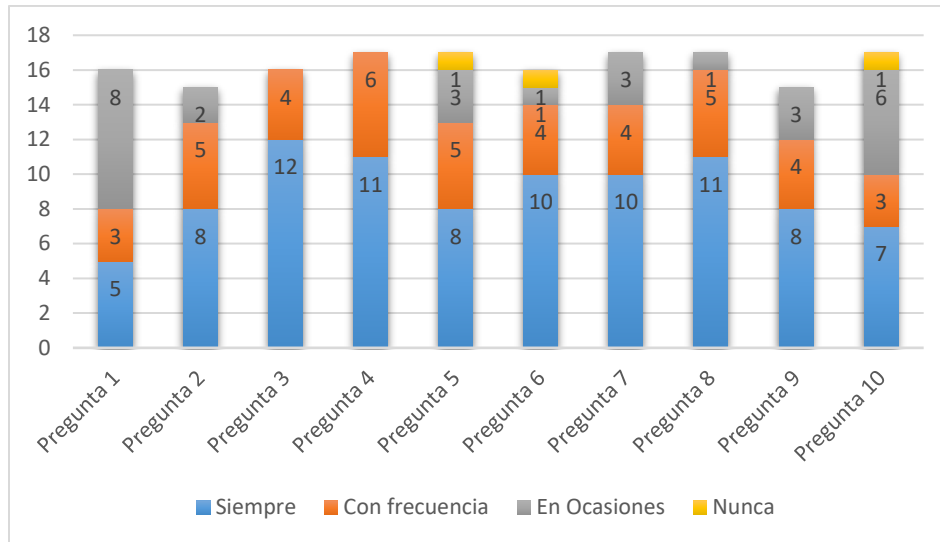


Figura 44. Total, de respuestas de participantes de la muestra del nivel 1.

Como conclusión de la Figura 44, los individuos de la muestra tienen o han tenido un acercamiento a métodos ágiles y conocen o aplican de una forma transversal los métodos en su ambiente tecnológico, la mayoría supero el nivel.

**Nivel 2 (Planeación)**, Visión general acerca de la planeación del equipo con métodos ágiles del individuo de la muestra.

A continuación, se muestra el gráfico de las respuestas constantes aplicadas al grupo de individuos de la muestra para el nivel 2:

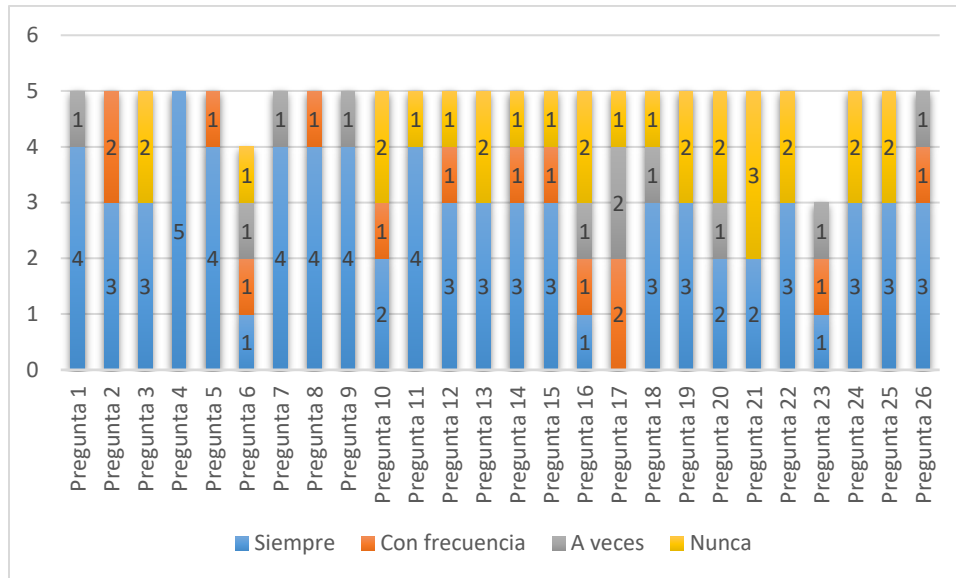


Figura 45. Total, de respuestas de individuos de la muestra del nivel 2.

Como conclusión de la Figura 45, los individuos de la muestra han realizado planeación sobre proyectos de software, así como la gestión y control de incidentes ocurridos durante el desarrollo utilizando métodos ágiles. Además, el nivel de aprobación de los participantes del nivel se redujo a 4.

**Nivel 3 (Comunicación)**, Visión general acerca del conocimiento sobre comunicación e incorporación del individuo prueba acerca de métodos ágiles.

A continuación, se muestra el gráfico de las respuestas constantes aplicadas al grupo de individuos de la muestra para el nivel 3:



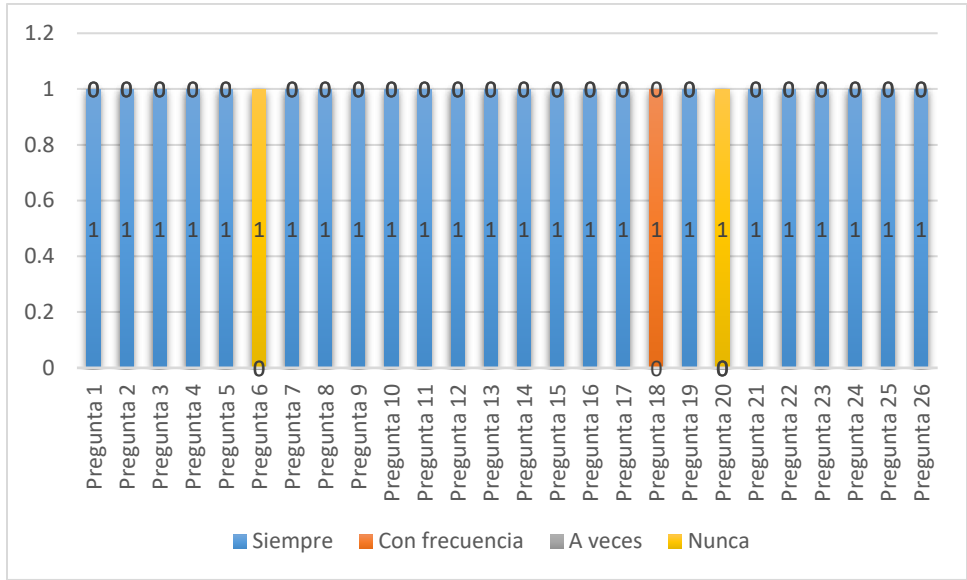


Figura 46. Total, de respuestas de individuos de la muestra para el nivel 3.

Como conclusión del Figura 46, los individuos de la muestra utilizan o aplican métodos como priorización, gestión en la organización, comunicación constante en el equipo y ambientes de pruebas, durante el desarrollo con métodos ágiles. Además, el nivel de aprobación de esta prueba de este nivel por los individuos de la muestra se redujo a 1.

**Nivel 4 (Comportamiento)**, Visión general acerca del conocimiento sobre el comportamiento y administración del individuo prueba acerca de métodos ágiles.

A continuación, se muestra el gráfico de las respuestas constantes aplicadas al grupo de individuos de la muestra para el nivel 4:

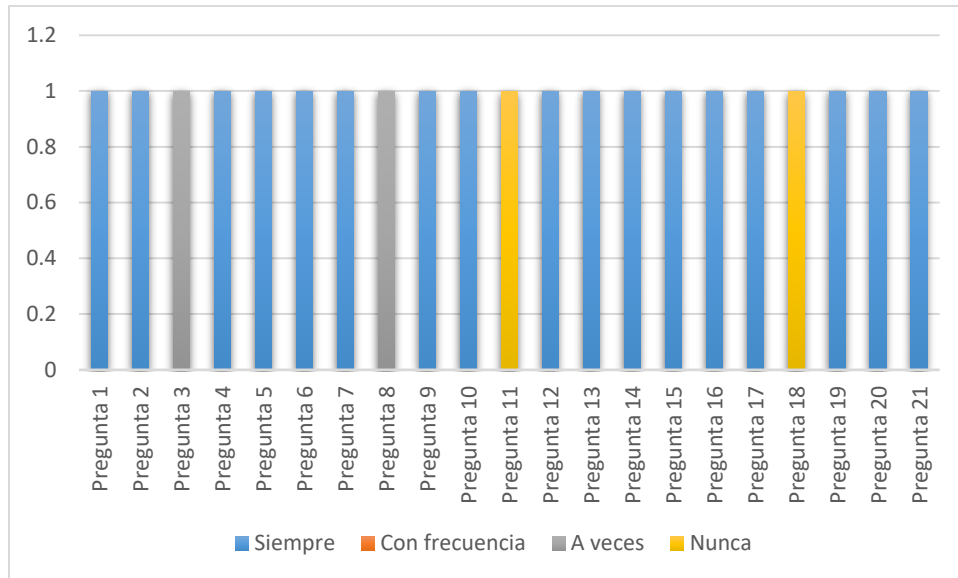


Figura 47. Total, de respuestas de individuos de la muestra para el nivel 4.

Como conclusión de la Figura 47, los individuos de la muestra no conocen como aplicar los métodos ágiles con el uso de herramientas que permitan un mejor control y gestión de los recursos en un proyecto de software. Además, el nivel de aprobación de esta prueba de este nivel por los individuos de la muestra se redujo a 1.

**Nivel 5 (Gestión del proceso)**, Visión general acerca del conocimiento sobre incorporación a actividades de desarrollo del individuo prueba acerca de métodos ágiles.

A continuación, se muestra el gráfico de las respuestas constantes aplicadas al grupo de individuos de la muestra para el nivel 5:

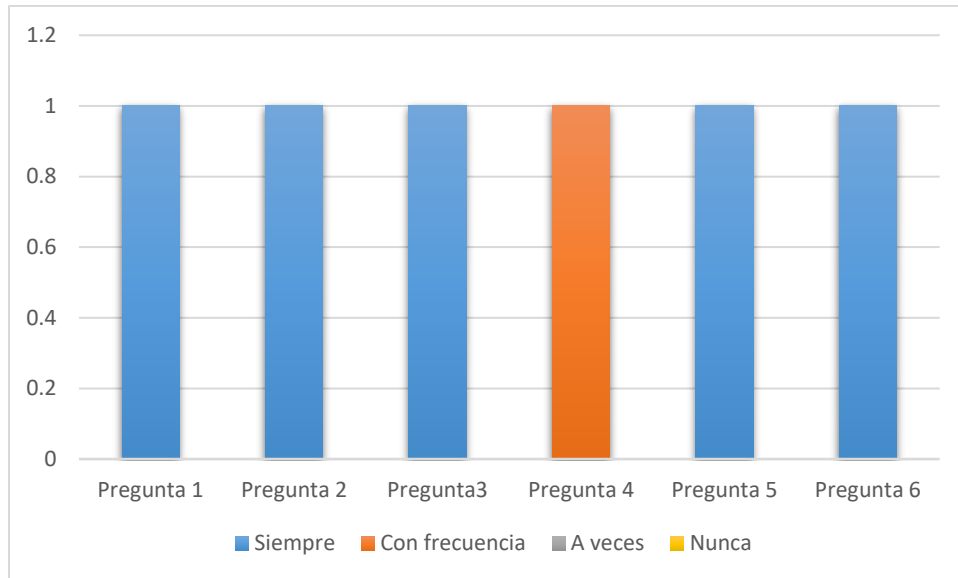


Figura 48. Total, de respuestas de individuos de la muestra para el nivel 5.

Como conclusión de la Figura 48, los individuos de la muestra no conocen o no aplican constantemente métodos ágiles para la comunicación, gestión y control de cómo aplicar los métodos ágiles en un proyecto de software.

## Capítulo 5. Conclusiones

En este capítulo se presentan los resultados obtenidos de este trabajo de tesis. Además, se presenta el trabajo futuro y los productos académicos derivados de la investigación.

## 5.1 Conclusiones

Cada vez más organizaciones dedicadas a desarrollo de software desean incrementar y mejorar la forma de desarrollo de software para obtener productos de mayor calidad, en un esfuerzo por lograr tal fin y con la introducción de los métodos ágiles, diversas organizaciones han incorporado a sus procesos estos métodos.

Sin embargo, aún con la implementación de los métodos ágiles las PYMEs, siguen teniendo problemas al desarrollar software, ya que existe una falta de conocimiento de cómo usar o implementar un método ágil correctamente.

Esta investigación analizó los resultados donde los métodos ágiles tienen realce, incluyendo además, la aplicación de los métodos en PYMEs, mencionando algunos países como Estados Unidos y Europa. Cabe resaltar que, esta investigación se centró en las PYMEs de México para proveer una herramienta que apoye en adoptar y/o mejora su conocimiento en la implementación de los métodos ágiles y en específico de Scrum. Para lograrlo, esta investigación se centra en la definición de un método que evalúa el uso e implementación del método Scrum, proporcionando un medio para mejorar su conocimiento y habilidades en este método ágil.

Para verificar si el conocimiento acerca de los métodos ágiles en la organización se aplica correctamente y si se cuenta con el conocimiento adecuado para el uso y la implementación, el método desarrollado incorpora como base para la evaluación a el modelo de madurez ágil (AMM) y Scrum Checklist , definiendo cinco niveles de evaluación para obtener un diagnóstico y obtención del conocimiento del método ágil enfocando en herramientas, técnicas, actividades, roles y métricas, con el objetivo que al final se obtenga un grado de madurez en el uso y aplicación de método ágil Scrum.

Para lograr el desarrollo del método diversas actividades se han llevado a cabo como a continuación se describen y que están enfocadas en el cumplimiento de los objetivos propuestos en esta tesis:

Se realizó una revisión sistemática de la literatura con el fin de establecer el estado de arte, que permitió encontrar registros de estudios previos en el marco de interés de la investigación, resaltado como resultado que cada vez más se utilizan y aplican los métodos ágiles, y que existen propuestas desarrolladas para generar una evaluación del conocimiento o aplicación de éstas.

A raíz de los resultados obtenidos en la revisión sistemática, se observó que no se cuenta con información relevante sobre el tema en México, por lo que se llevó a cabo un estado de la práctica generando una encuesta como instrumento que se aplicó a nivel nacional, lo cual permitió conocer si en México utilizan, conocen o aplican los métodos ágiles. El resultado de la encuesta resaltó que en México las personas con mayor conocimiento de métodos ágiles son los profesionales con un puesto en desarrollo de software en las PYMEs, además, se identificó que el uso de los métodos ágiles ha mejorado el proceso de desarrollo. sin embargo, se siguen manteniendo los métodos establecidos en la organización se prueba el método ágil, pero terminan regresando a sus métodos anteriores.

Se definió un método de evaluación del uso e implementación de métodos ágiles que permite conocer a las organizaciones el conocimiento que tienen acerca de los métodos ágiles. Además, de mejorar su conocimiento y sugerir la forma correcta de aplicarlo. El método está compuesto de cinco cuestionarios que se presentan en cada nivel y, los cuales permitirán avanzar dependiendo de la selección de la respuesta, así mismo permite conocer el grado de madurez con respecto a la agilidad, ya sea organizacionalmente o individualmente.

Para el uso del método propuesto, se generó una herramienta que incorpora los cinco niveles de evaluación donde cada uno está compuesto por una encuesta que permite valorar el conocimiento del usuario de acuerdo a las métricas propuestas, permitiendo la comparación del nivel inicial y final. Además, muestra un video explicativo acerca de los niveles en el método Scrum y como se aplican. Cabe destacar que la herramienta implementa una evaluación, que permite avanzar dependiendo del conocimiento evaluado, así como si se responde de una manera errónea en una pregunta importante se le restara puntos.

Contando con una herramienta que disponga un método que permita evaluar y mejorar, así como medir el conocimiento de una persona de un método ágil aplicado. Entonces, se logra facilitar el diagnóstico de la adquisición de conocimiento y las habilidades en el uso del método ágil. y de esta forma comprobando la hipótesis antes planteada.

La tendencia de las PYMEs en México sigue siendo la producción de software de forma empírica con algunas incorporaciones con técnicas como pruebas unitarias, uso de proceso de software personal para la administración personal y de proyectos, además, existen empresas que incorporan el método ágil a su proceso. Sin embargo, que a pesar de que incorporaron un método ágil no identifican ninguna mejoría en la producción o en el proceso lo cual, por lo que regresan a los procedimientos que ya tenían implantados o adoptan otro nuevo, aunque consideran que el proceso existente para el desarrollo de software es robusto y tedioso, continúan llevándolo a cabo.

## **5.2 Trabajo Futuro**

Como trabajo futuro se establecieron las siguientes propuestas:

- Generar diversas encuestas por cada nivel.
- Incluir sugerencias acerca de certificación en métodos ágiles dependiendo del avance.
- Incluir un simulador de proyectos con método ágil para evaluación del conocimiento y habilidad.

## **5.3 Logros**

A continuación, se muestran los productos que se han generado como resultado de la investigación:

### 5.3.1 Publicaciones

- 2017. Muñoz, Mirna; Mejia, Jezreel; Calvo-Manzano, Jose A.; San Feliu, Tomas; Corona, Brisia; Miramontes, Juan. Diagnostic Assessment Tools for Assessing the Implementation and/or Use of Agile Methodologies in SMEs: An Analysis of Covered Aspects. *Software Quality Professional*, 19(2), ISSN: 15220542, pp 16-27
- Corona Brisia, Muñoz Mirna, Miramontes Juan, Calvo-Manzano Jose A., San Feliu Tomas (2016), *Revisión Sistemática para Establecimiento del Estado de Arte sobre Frameworks, Métodos y Metodologías enfocados en Evaluar la Implementación y Uso de las Métodos Ágiles en las Pymes*, Software, SICC 2015, Seminario Internacional en Ciencias de la Computación.
- Corona Brisia, Muñoz Mirna, Miramontes Juan, Calvo-Manzano Jose A., San Feliu Tomas (2016) *Establishing the state of art of frameworks, methods and methodologies to assess the implementation and use of agile methodologies in SMEs: A systematic review*, ReCIBE Año 5 No.1, Abril 2016, Revista electrónica de Computación, Informática, Biomédica y Electrónica
- Mirna Muñoz, Jezreel Mejia, Brisia Corona (2016), *Hacia la evaluación de la implementación y uso de métodos ágiles en las PYMEs: Un análisis de herramientas de evaluación de Métodos ágiles-Toward assessing the implementation and use of agile methodologies in SMEs: An analysis of the actual agile methodologies assessment tools*, Software, CISTI2016- 11ª Conferencia Ibérica de Sistemas y Tecnologías de Información

- Muñoz Mirna, Mejia Jezreel, Corona Brisia, Miramontes Juan, Calvo-Manzano Jose A., San Feliu Tomas (2016), *Analysis of tools for assessing the implementation and use of agile methodologies in SMEs*, SPICE 2016, 16th International Conference on Process improvement and Capability Determination in Software.

### 5.3.2 Estancia

Se realizó una estancia en la Universidad Politécnica de Madrid en España, con la finalidad de realizar la revisión sistemática de la literatura, así como desarrollar y concretar el estado del arte de la investigación.

### 5.3.3 Presentación CIMPS

Como resultado de la investigación de estas 41 herramientas se generó un artículo para la “*4a Conferencia Internacional en Mejora de Procesos Software*” (CIMPS) en donde se presentaron los resultados obtenidos.



# Referencias

- AgilityHealth. (2015). Accelerate Your Maturity and Agility by Measuring What Matters. Agosto 11, 2016, de agilityhealthradar Sitio web: <https://agilityhealthradar.com/>
- BUSINESS, T. D. (2015). Assessing your Client's Agility - An Agility Questionnaire. Agosto 11, 2016, de thedigitalbusinessanalyst Sitio web: <https://thedigitalbusinessanalyst.co.uk/>
- BPM Institute.org. (2020). Skills Self-Assessments. Octubre 14, 2020, de BPMInstitute.org Sitio web: <https://www.bpmiunstitute.org/skills-self-assessments>
- Chetankumar, P. and Ramachandran, M. (2009). A Software Process Improvement framework for Agile Software Development Practices. En Agile Maturity Model (AMM) (pp.1-25). Researchgate: [https://www.researchgate.net/publication/45227382\\_Agile\\_Maturity\\_Model\\_AMM\\_A\\_Software\\_Process\\_Improvement\\_framework\\_for\\_Agile\\_Software\\_Development\\_Practices](https://www.researchgate.net/publication/45227382_Agile_Maturity_Model_AMM_A_Software_Process_Improvement_framework_for_Agile_Software_Development_Practices)
- Cossio, M., Giesen, L., Araya, G., Pérez-Cotapos, M., VERGARA, R, Manca, M., and Héritier, F. (2012). Teoría-MÉTODOS DE DESARROLLO DE SOFTWARE: EL DESAFÍO PENDIENTE DE LA ESTANDARIZACIÓN. Una ética Para Cuántos?, XXXIII (2), 23–42. <http://doi.org/10.1007/s13398-014-0173-7.2>
- Carnegie Mellon University. (2015). the Software Engineering Institute (SEI). Septiembre 15, 2015, de SEI Sitio web: <https://www.sei.cmu.edu/about/>
- Codd. E. (19-03-2008). herramientas utilizadas por las empresas para las aplicaciones OLAP (OLAP significa 'On-Line Analytical Processing'). Octubre 22, 2015, de Conocimiento con todos y para todos (ecured) Sitio web: [http://www.ecured.cu/index.php/Cubos\\_OLAP](http://www.ecured.cu/index.php/Cubos_OLAP).
- Durón Brenda, Muñoz Mirna, Mejía Jezreel Centro de Investigación en Matemáticas Av. Universidad no 222, 98068 Zacatecas, México {brenda.duron, mirna.munoz, jmejia}@cimat.mx. (2013). Estado actual de la implementación de mejoras de procesos en las organizaciones software. IEEE, 1, 7. 10/06/2015, De IEEE Base de datos.
- De La Villa, M., Ruiz, M., and Ramos, I. (2004). Modelos de Evaluación y Mejora de Procesos: Análisis Comparativo. CEURWorkshop Proceedings, Vol 120 Paper 4, 1–18. 2015, Agosto 16, De Researchgate Base de datos.
- Designer, L. (n.d.). Algunas Características de HTML5 que usted debe saber. Retrieved from <http://latindesigner.net/algunas-caracteristicas-de-html5-que-usted-debe-saber/>
- El semanario. (n.d.). 1 PYMEs en México, fuente principal de empleos. Economista, 10. Retrieved from <http://elsemanario.com/34293/analisis-mexico-vs-los-brics-china/>
- EcuRed. (2015). PHP. Septiembre 23, 2016. de Red Ecuatoriana EcuRed Sitio web: <https://www.ecured.cu/PHP>
- Estrada, S, Cano, K. and Aguirre J. (2018). ¿Cómo se gestiona la tecnología en las pymes? Diferencias y similitudes entre micro, pequeñas y medianas empresas. Contaduría y Administración 64 (1) Especial Innovación, 2019, Especial Innovación, 2019, 1-21. octubre 30, 2020, De <http://www.scielo.org.mx/>.
- Flórez, H. (2010). Arquitectura multicapa mediante ajax and orm Multilayer architecture driven ajax and orm. Enero 10, 2021, de <http://www.konradlorenz.edu.co/> Sitio web: [http://www.konradlorenz.edu.co/images/investigaciones/matematicas/crear%20sitios\\_web.pdf](http://www.konradlorenz.edu.co/images/investigaciones/matematicas/crear%20sitios_web.pdf)
- Gloger Boris. (2012). Scrum Checklist 2012-Your Scrum Checklist. Alemania: Realisation Kraftjungs: scrumlies Jo Legat (Independiente), p.5 o pp.5-22.
- Gómez M. (2011). Material Didáctico Notas del Curso - Análisis de Requerimientos. Abril 15, 2015, de Universidad Autónoma Metropolitana Unidad Cuajimalpa Sitio web: [http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas\\_Analisis\\_Requerimiento.pdf](http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas_Analisis_Requerimiento.pdf)
- Gómez-Gil2, C. E. R. P. (2012). Análisis empírico sobre la adopción de las metodologías ágiles en los equipos de desarrollo de software en empresas mexicanas. Tópicos Selectos de Tenologías de La Información Y Las Comunicaciones, 8. Retrieved from <http://ccc.inaoep.mx/~pgomez/publications/congress/ERCANI12.pdf>
- Gasca-Hurtado G.P., Giraldo L.M., Calvo-Manzano J.A., Echeverri Arias J.A. (2011). Análisis estadístico de la implementación de buenas prácticas en organizaciones desarrolladoras de software. IEEE, 1, 8. junio 10, 2015, De IEEE Base de datos.
- Gómez G.E., Aguilar A., Ancona G.B., Gómez O. (2014). Avances en las Mejoras de Procesos Software en las MiPyMEs Desarrolladoras de Software: Una Revisión Sistemática. Revista Latinoamericana de Ingeniería de Software, 2(4): 262-268, ISSN 2314-2642, 1, 7. Julio 01, 2015, De IEEE Base de datos
- Hypertextual. (n.d.). Entendiendo HTML5: guía para principiantes. Retrieved from <http://hipertextual.com/archivo/2013/05/entendiendo-html5-guia-para-principiantes/>
- Heredia Ruiz J, Álvarez Almanza L, Linares Pons N. Comparación y tendencias entre metodologías ágiles y formales. Metodología utilizada en el Centro de Informatización para la Gestión de Entidades. Serie Científica de la Universidad de las Ciencias Informáticas <http://publicaciones.uci.cu/index.php/SC|seriecienfifica@uci.cu>, 1, 17. Julio 15, 2015, De Scholar Google Base de datos.
- ISO WEB Espaciales, I.-I. C. de D. (2015). Estándares. [Http://www.icde.org.co/web/guest/wiki/-/wiki/Wiki](http://www.icde.org.co/web/guest/wiki/-/wiki/Wiki) de la ICDE/Est%C3%AIndares

- ISO Historia de las métricas de calidad de software 2.1. (n.d).  
[http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/moreno\\_a\\_jl/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/moreno_a_jl/capitulo2.pdf)
- ISO (International Organization for Standardization). (2015). About ISO. Julio 15, 2015, de ISO Sitio web: <http://www.iso.org/iso/home/about.html>
- ISO, 2004a. ISO/IEC 15504-2:2003/Cor.1:2004(E). Information technology –Process assessment – Part 2. Performing an assessment. International Organization for Standardization. Available from [www.iso.org](http://www.iso.org).
- ISO, 2004b. ISO/IEC 15504-4:2004 Information technology – Process assessment– Part 4. Guidance on use for process improvement and process capability determination. International Organization for Standardization. Available from [www.iso.org](http://www.iso.org).
- ISO, 2008. ISO/IEC TR 15504-7:2008. Information technology – Process assessment– Part 7. Assessment of organizational maturity. International Organization for Standardization.
- Iver, R. Mac. (2010). Agile Maturity Self-Assessment. Octubre, 10, 2015, 1–2. <http://www.robbiemaciver.com/documents/presentations/A2010-Agile Maturity Self-Assessment.pdf>
- IBM Corporation. (2012). Herramientas y software DevOps. octubre 10, 2015, de IBM Corporation Sitio web: [https://www.ibm.com/mx-es/cloud/devops?p1=Search&p4=43700052828197561&p5=b&gclid=CjwKCAjwv\\_iEBhASEiwARoemvNEPZA\\_G9CLgUq6CpcuuffrpHJvV-IPbPIOcH8aik5XiwZai36\\_HfJRoCuk0QAvD\\_BwE&gclsrc=aw.ds](https://www.ibm.com/mx-es/cloud/devops?p1=Search&p4=43700052828197561&p5=b&gclid=CjwKCAjwv_iEBhASEiwARoemvNEPZA_G9CLgUq6CpcuuffrpHJvV-IPbPIOcH8aik5XiwZai36_HfJRoCuk0QAvD_BwE&gclsrc=aw.ds)
- International Organization for Standardization (ISO). (---). ISO 9001:1994. Abril 14, 2016, de ISO Sitio web: [http://www.iso.org/iso/catalogue\\_detail.htm?csnumber=16534](http://www.iso.org/iso/catalogue_detail.htm?csnumber=16534)
- ibrugor.com. (21 octubre, 2014). ¿Qué es PHP? ¿Para qué sirve? Noviembre 23, 2015, de ibrugor-Consultoria-Marketing-online Sitio web: <http://www.ibrugor.com/blog/que-es-php-para-que-sirve/>
- Ivan Ramirez. (2018). Ciclo de vida de desarrollo de Software. Octubre 13, 2020, de efectodigital. online Sitio web: <https://www.efectodigital.online/post/2018/04/23/ciclo-de-vida-de-desarrollo-de-software>
- Instituto Nacional de Tecnologías de la Comunicación (INTECO). (marzo 2009). INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA. ScholarGoogle, 1, 83. Julio 15, 2015.
- Jacobson, I. (2015). AGILE ESSENTIALS STARTER PACK | AGILE PRACTICES. Retrieved from <https://goo.gl/BFHDtN>
- Janlén, J. Team Barometer. Retrieved from <http://goo.gl/2zL8V> (2014)
- José H. Canós, Patricio Letelier y M<sup>a</sup> Carmen Penadés DSIC -Universidad Politécnica de Valencia Camino de Vera s/n, 46022 Valencia {jhcans | letelier | mpenades }@dsic.upv.es. (2011). Metodologías Ágiles en el Desarrollo de Software. IEEE, 1, 8. Julio 15, 2015, De IEEE Base de datos.
- Kent Beck Mike Beedle Arie van Bennekum Alistair Cockburn Ward Cunningham Martin Fowler James Grenning Jim Highsmith Andrew Hunt Ron Jeffries Jon Kern Brian Marick Robert C. Martin Steve Mellor Ken Schwaber Jeff Sutherland Dave Thomas. (2001). Manifiesto por el Desarrollo Ágil de Software. Julio 06, 2015, de Alianza ágil Sitio web: <http://agilemanifesto.org/iso/es/>
- Kitchenham, B., and Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering (Vol.2 ed.)
- Kniberg, H. The Unofficial Scrum Checklist. Retrieved from <https://goo.gl/D6whaJ>. (2010).
- Kent Beck Mike Beedle Arie van Bennekum Alistair Cockburn Ward Cunningham Martin Fowler James Grenning Jim Highsmith Andrew Hunt Ron Jeffries Jon Kern Brian Marick Robert C. Martin Steve Mellor Ken Schwaber Jeff Sutherland Dave Thomas. Manifiesto for agile software development. Available at: [www.agilemanifesto.org/](http://www.agilemanifesto.org/). (2001).
- Kouzina, O. (n.d.). 1 Edge of Chaos Agile Development Blog. SCRUM, LEAN, KANBAN, VISUALIZATION, USER EXPERIENCE, 10. Retrieved from <http://www.targetprocess.com/blog/2013/08/back-to-the-future-of-agile-software-development.html>
- La Secretaría de Hacienda y Crédito Público. (miércoles, 11 de junio, 2014). VISIÓN Y MISIÓN. miércoles, 11 de junio, 2014, de LA SHCP Sitio web: <http://www.hacienda.gob.mx/LASHCP/Paginas/visionMision.aspx>
- Lopez, E. (2016). Arquitectura de ncapas. Retrieved from [http://www.academia.edu/10102692/Arquitectura\\_de\\_ncapas](http://www.academia.edu/10102692/Arquitectura_de_ncapas)
- Laboratorio Nacional de Calidad del Software Marzo; Instituto Nacional de Tecnologías de la comunicación (INTECO). (marzo de 2009). INGENIERÍA DEL SOFTWARE: METODOLOGÍAS Y CICLOS DE VIDA Laboratorio. Esta guía ha sido desarrollada por el Laboratorio Nacional de Calidad del Software de INTECO., 1, 83. Julio 15, 2015, De ScholarGoogle.
- Lalsing, V., Kishnah, S., and Pudaruth, S. (2012). P Eople F Actors in a Gile S Oftware, 3(1), 117–137. (GS\_ Metodología para evaluacion)
- LLC, A. D. ASSESS – AGILE JOURNEY INDEX. Retrieved from <http://goo.gl/hL7VJs> (2012).
- Little, J. Joe’s Unofficial Scrum Checklist. Retrieved from <http://goo.gl/z83cyY> Unofficial Scrum Checklist. (2016).
- Lagestee, L. How to Measure Team Agility. Retrieved from <http://goo.gl/fM3h9s> (2012).
- Llaneza M., Dapozo G., Greiner C., Estayno M. (2013). Análisis comparativo de modelos de calidad orientado al desarrollo de software en PYMES. XV Trabajo de Investigadores de la Computación, 1, 5. Julio 01, 2015, De Scholar Google Base de datos.
- Mahnic, V., and Drnovscek, S. (2005). Agile software project management with scrum. EUNIS 2005 Conference-Session Papers and Tutorial Abstracts, January 2005, 6. [fri.uni-lj.si](http://fri.uni-lj.si)

- Moreno, L. (2012). Historia de las métricas de calidad de software 2.1. noviembre 23, 2015, de Universidad de las Américas de Puebla (UDLAP) Sitio web: [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lis/moreno\\_a\\_jl/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/moreno_a_jl/capitulo2.pdf)
- Mejhem Yousef al-Tarawneha, Mohd Syazwan Abdullah, Abdul Bashah Mat Alic. (2010). A Proposed Methodology for Establishing Software Process Development Improvement for Small Software Development Firms. ScienceDirect, 3, 5. abril 15, 2016, De ScienceDirect Base de datos.
- Microsoft. (2016). Microsoft SQL Server. Retrieved from <https://msdn.microsoft.com/es-mx/library/bb545450.aspx>
- Microsoft, M. (n.d.). Información general acerca de .NET Framework. Retrieved from [https://msdn.microsoft.com/es-mx/library/zw4w595w\(v=vs.110\).aspx](https://msdn.microsoft.com/es-mx/library/zw4w595w(v=vs.110).aspx)
- MILLER, S. Is Your Organization Ready for Agile? - Part 1. Retrieved from <https://goo.gl/gK54Pc> (2012)
- Mendez Calo, K. (2009). Un framework para evaluación de metodologías Ágiles, (1). Retrieved from [http://sedici.unlp.edu.ar/bitstream/handle/10915/21086/Documento\\_completo.pdf?sequence=1](http://sedici.unlp.edu.ar/bitstream/handle/10915/21086/Documento_completo.pdf?sequence=1).(es el 55, un framework par evaluación de metodologías ágiles)
- Microsoft, MSDN LIBRARY. (actualizado en octubre del 2012). Microsoft. 2015, de Microsoft Cooperation Sitio web: <http://msdn.microsoft.com>
- MSDN LIBRARY. (actualizado en octubre del 2012). Visual Studio 2010. Enero 05, 2013, de Microsoft Cooperation Sitio web: <http://msdn.microsoft.com/en-us/library/VisualStudio>
- MSDN LIBRARY. (actualizado en octubre del 2012). SQL Server 2012. Noviembre 22, 2015, de Microsoft Cooperation Sitio web: <http://msdn.microsoft.com/en-us/library/SQL>
- Microsoft, M. (n.d.). Lenguaje Visual C#. Retrieved from [https://msdn.microsoft.com/es-MX/library/aa287558\(v=vs.71\).aspx](https://msdn.microsoft.com/es-MX/library/aa287558(v=vs.71).aspx)
- Mejhem Yousef al-Tarawneha, Mohd Syazwan Abdullah, Abdul Bashah Mat Alic. (2010). A Proposed Methodology for Establishing Software Process Development Improvement for Small Software Development Firms. ScienceDirect, 3, 5. Abril 15, 2016, De ScienceDirect Base de datos.
- Nordin, J. (2013). Depth of Kanban - A Good Coaching Tool. Agosto 23, 2015, de Lean-Agile Software Development Sitio web: <http://leanagileprojects.blogspot.com/2013/03/depth-of-kanban-good-coaching-tool.html>
- NormasISO. (actualizado2020). ISO/IEC 15504 SPICE. Octubre 13, 2020, de NormasISO Sitio web: <https://www.normas-iso.com/iso-iec-15504-spice/>
- N. Blanco. (2018). Qué es Doctrine y por qué utilizarlo. Enero 10, 2021, de <https://openwebinars.net/> Sitio web: <https://openwebinars.net/blog/que-es-doctrine-y-como-funciona>
- Ortíz García M. (2014). “Desarrollo de una aplicación para la unificación de metodologías ágiles para la gestión de proyectos en entornos outsourcing para PYMES”. Orizaba, Veracruz, México: Instituto Tecnológico de Orizaba.
- OCDE/CA. (2019). Índice de Políticas PYME: América Latina y el Caribe 2019 Políticas para PYMEs competitivas en la Alianza del Pacífico y países participantes de América del Sur. AMÉRICA LATINA Y EL CARIBE 2019, 1, 40. octubre 30, 2020, De [www.oecd.org](http://www.oecd.org).
- PLADcloud. (2020). Agile Adoption Interview. Octubre 14, 2020, de PLADcloud Sitio web: <https://pladcloud.typeform.com/to/HjcVKG>
- Pinoa, J., Pedreira, O., García, F., Rodríguez, M. & Piattini, M. (2010). Using Scrum to guide the execution of software process improvement in small organizations. Elsevier, 83, 16. Abril 15, 2016, De The Journal of Systems and Software Base de datos
- Pérez M.A. (2014). Guía Comparativa de Metodologías Ágiles. 2015, de Universidad de Valladolid E. U. de Informática (SEGOVIA) Sitio web: <https://uvadoc.uva.es/bitstream/10324/1495/1/TFG-B.117.pdf>
- Proyectos ágiles (2012). Qué es SCRUM. Noviembre 23, 2015, de [proyectosagiles.org](http://proyectosagiles.org) Sitio web: <https://proyectosagiles.org/que-es-scrum/>
- Project Management Institute (PMI). (11/02/2020). Pulse of the Profession (2020). (PMI), N, 1. noviembre 4, 2020, De <https://www.pmi.org/learning/library/forging-future-focused-culture-11908>
- Ruiz, J., Almanza, L., and Pons, N. (2011). Comparación y tendencias entre metodologías ágiles y formales. Metodología utilizada en el Centro de Informatización para la Gestión de Entidades. Serie Científica, (10). Retrieved from <https://publicaciones.uci.cu/index.php/SC/article/view/484/0>
- Rothman, J. Questions for Transitioning to Agile. Retrieved from <http://goo.gl/cR3meH> (2013).
- Rubin, M. C. and K. Comparative Agility. Retrieved from <https://goo.gl/I039oe> (2008).
- Rina Marissa Aguilera Hintelholzer. (28 México ene./abr. 2013). Identidad y diferenciación entre Método y Metodología. Abril 15, 2016, de SCIELO Sitio web: [http://www.scielo.org.mx/scielo.php?script=sci\\_arttext&pid=S0185-16162013000100005](http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S0185-16162013000100005)
- RefineM. (2020). Agility Assessment. Octubre 14, 2020, de by RefineM Sitio web: <https://agility.refinem.com/>
- SIDKY, A. (2011). Enterprise Agile Practice Assessment Tool. Agosto 23, 2015, de [ahmedsidky](http://ahmedsidky.com/), Educating for agile mindsets Sitio web: <https://www.ahmedsidky.com/>
- Sahota, M. (2012). An Agile Adoption and Transformation Survival Guide. Octubre 16, 2015, de [infoq.com](http://infoq.com) Sitio web: <https://www.infoq.com/minibooks/agile-adoption-transformation/>
- SG Buzz. (2004). Procesos de Software. 2015, de SG Buzz Sitio web: <http://sg.com.mx/revista/1/procesos-software#.V2qvFLgrIdU>
- Scrum.org. Open Assessments. Retrieved from <https://www.scrum.org/Assessments/Open-Assessments> (2015).
- Sahota, M. An Organizational Transformation Checklist. Retrieved from <http://goo.gl/CPRfyU> (2012).

- Storm-Consulting. Agile Enterprise Survey. Retrieved from <http://goo.gl/uBTdlw> (2016).
- Scruminc. ScrumButt Test aka the Nokia Test. Retrieved from <https://goo.gl/2VNsku> (2015).
- Schenck, M. Agile maturity – How agile is your organization? Retrieved from <http://goo.gl/MjO8SD> (2015).
- Sidky, S. E. based on research by A. Ready for Agile Demo Part 2-part1. Retrieved from <http://safeshare.tv/w/HWCJuBVueB>. (2012).
- Schenck, M. Agile Maturity Matrix in JIRA. Retrieved from <http://goo.gl/weQC4t> (2013).
- Subgurim.net 2016. (lunes, 06 de febrero de 2006). ¿Por qué ASP.NET? noviembre 23, 2015, de subgurim.net Sitio web: <http://www.subgurim.net/Articulos/asp-net-general/3/por-que-asp-net.aspx>
- Sutherland, K., & SchwaberJeff. (n.d.). La Guía de Scrum. Retrieved from <http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf>
- Silva, F. S., Soares, F. S. F., Peres, A. L., Azevedo, I. M. de, Pinto, P. P., & Meira, S. R. de L. (2014). A Reference Model for Agile Quality Assurance: Combining Agile Methodologies and Maturity Models. 2014 9th International Conference on the Quality of Information and Communications Technology, 139–144. doi:10.1109/QUATIC.2014.25
- Shvetha Soundararajan. (March 17, 2011). A METHODOLOGY FOR ASSESSING AGILE SOFTWARE DEVELOPMENT APPROACHES. Computer Science and Applications, 1, 68. Abril 15, 2016, De IEEE Base de datos.
- software Engineering Institute, "CMMI for Development, Version 1.3," ed, 2010.
- Sven Overhage, Sebastian Schlauderer, Dominik Birkmeier, Jonas Miller. (2011). What Makes IT Personnel Adopt Scrum? A Framework of Drivers and Inhibitors to Developer Acceptance. IEEE, 44, 10. abril 15, 2016, De IEEE Base de datos.
- Sofia Riesco. (2018). ¿Cuáles son las normas ISO más importantes? Octubre 13, 2020, de formazion.com Sitio web: [https://www.formazion.com/noticias\\_formacion/cuales-son-las-normas-iso-mas-importantes-org-5273.html](https://www.formazion.com/noticias_formacion/cuales-son-las-normas-iso-mas-importantes-org-5273.html)
- StateofAgile. (Diciembre, 2019). the 14-annual state of agile survey. the 14 annual State of Agile Report, 14, 18.
- Tomohiro Hayata Computer Science Department California State University Dominguez hills Carson, California, USA [thayata@toromail.csudh.edu](mailto:thayata@toromail.csudh.edu); Jianchao Han Computer Science Department California State University Dominguez hills Carson, California, USA [jhan@csudh.edu](mailto:jhan@csudh.edu). (2011). A Hybrid Model for IT Project with Scrum. IEEE, 1, 6. Junio 10, 2015, De IEEE Base de datos.
- Trenches, F. the A. T. Assessment, Borland Agile. Retrieved from <http://goo.gl/k71DXm> (2015).
- Think Big, A. S. (2019). 2 CHAOS MANIFESTO 2013. CHAOS MANIFESTO, 2, 52. Retrieved from <http://www.versionone.com/assets/img/files/ChaosManifesto2019.pdf>
- The Standish Group. (2019). 9-chaos-report-decision-latency-theory-2019. Abril 2021, de The Standish Group Sitio web: <https://www.standishgroup.com/news/45>
- Vargas, I. (2013). Pymes generan 81% del empleo en México. Noviembre 23, 2015, de CNNEXPANSION Sitio web: <https://expansion.mx/mi-carrera/2013/01/14/pymes-generan-81-del-empleo-en-mexico>
- Van Der, C., Henst S. (2001). ¿Qué es PHP? Noviembre 23, 2015, de maestrosdelweb.com Sitio web: <http://www.maestrosdelweb.com/phpintro/>
- Vanessa Rossello Villán. (2019). Las metodologías ágiles más utilizadas y sus ventajas dentro de la empresa. Octubre 13, 2020, de ibsschool.com Sitio web: <https://www.iebschool.com/blog/que-son-metodologias-agiles-agile-scrum/>
- Violeta 1879354. (n.d.). Características principales de C#. Retrieved from <http://es.scribd.com/doc/7411856/Caracteristicas-de-C#scribd>
- VersionOne. (2002). Pioneers in Agile Project Planning & Management. Julio 15, 2015, de © 2015, VersionOne, Inc. All Rights Reserved. TeamRoom™ is a trademark of VersionOne Inc. Sitio web: <http://www.versionone.com/about-us/>
- VersionOne. (2015). state-of-agile-development-survey-ninth. VersionOne, 9, 16. Julio 12, 2015, De © 2015, VersionOne, Inc. All Rights Reserved. TeamRoom™ is a trademark of VersionOne Inc. Base de datos.
- VersionOne. (2014). 8vo.Estudio Anual del Estado de las Ágiles. VersionOne, 8, 17. Julio 15, 2015, De © 2015, VersionOne, Inc. All Rights Reserved. TeamRoom™ is a trademark of VersionOne Inc Base de datos.
- VersionOne. (2018). 15th State of Agile Report. VersionOne, 15, 6-10. abril 16. 2021, De © 2015, VersionOne, Inc. All Rights Reserved. TeamRoom™ is a trademark of VersionOne Inc Base de datos. Retrieved from <https://stateofagile.com/>
- Waters, K. 42point test: How Agile are You. Retrieved from <http://goo.gl/VgKF7q> (2008).
- Windows. (2015). Choosing a technology for your Windows 10 game. Retrieved from <https://dev.windows.com/en-us/games/technology>
- Web-gdl. (n.d.). Ventajas del HTML5. Retrieved from <http://web-gdl.com/servicios/ventajas-del-html5/>
- Watts S. Humphrey, Sofi ware Engineering Institute. (Publicado en 1988, Republicado 2012). Characterizing the Software Process: A Maturity Framework. Instituto de Ingeniería Eléctrica y Electrónica (IEEE), 1, 7. Julio 23, 2015, De IEEE Base de datos.
- Xiaofeng Wanga, Kieran Conboyb, Oisin Cawleyc. (2012). “Leagile” software development: An experience report analysis of the application of lean approaches in agile software development. elsevier, 85, 13. Abril 15, 2016, De The Journal of Systems and Software Base de datos.
- Yatzeck, E. Corporate Agile 10-point checklist. Retrieved from <http://goo.gl/bFrjky> (2012).