



Pontificia Universidad
Católica del Ecuador

UNIDAD ACADÉMICA:

OFICINA DE POSTGRADOS

TEMA:

Modelo de Integración del *Project Management Body of Knowledge* con las
Metodologías Ágiles de Desarrollo de Software

**Proyecto de investigación y Desarrollo previo a la obtención del título
de**

Magister en Gerencia Informática

Línea de Investigación, Innovación y Desarrollo principal:

Ingeniería de Software y/o Plataformas Educativas

Caracterización técnica del trabajo:

Desarrollo

Autora:

Luisa Maribel Uribe Campaña

Director:

Ing. Darío Javier Robayo Jácome, Mg.

Ambato - Ecuador

Abril 2018

Modelo de Integración del *Project Management Body of Knowledge* con las Metodologías Ágiles de Desarrollo de Software

Informe de Trabajo de Titulación presentado ante la
Pontificia Universidad Católica del Ecuador Sede Ambato

Por:

Luisa Maribel Uribe Campaña

En cumplimiento parcial de
los requisitos para el Grado de
Magister en Gerencia Informática



Oficina de Postgrados

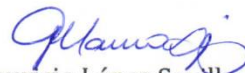
Abril 2018

**Modelo de Integración del *Project Management*
Body of Knowledge con las Metodologías Ágiles de
Desarrollo de Software.**

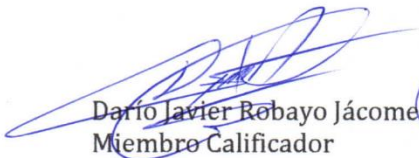
Aprobado por:



Concepción del Carmen Bedon Vaca, Mgt.
Presidente del Comité Calificador
Coordinador de la Oficina de
Postgrados



Galo Mauricio López Sevilla, Mg.
Miembro Calificador



Darío Javier Robayo Jácome, Mg.
Miembro Calificador
Director de Proyecto



Hugo Rogelio Altamirano Villarroel, Dr.
Secretario General



SECRETARÍA GENERAL
PROCURADURÍA



Liliana del Rocio Mena Hernández, Mg.
Miembro Calificador

Fecha de aprobación:
Abril 2018



BIBLIOTECA

Ficha Técnica

Programa: Magister en Gerencia Informática

Tema: Modelo de Integración del *Project Management Body of Knowledge* con las Metodologías Ágiles de Desarrollo de Software

Tipo de trabajo: Proyecto de Investigación.

Clasificación técnica del trabajo: Desarrollo.

Autor: Luisa Maribel Uribe Campaña

Director: Ing. Darío Javier Robayo Jácome, Mg.

Líneas de Investigación, Innovación y Desarrollo

Principal: Ingeniería de Software y/o Plataformas Educativas

Resumen Ejecutivo

En la zona centro del Ecuador han aumentado las pequeñas y medianas empresas de desarrollo de software. No obstante, presentan diversos problemas en la gestión de proyectos, que inciden directamente en el no cumplimiento de requerimientos, tiempo de duración, presupuesto y calidad, a criterio de antecedentes de estudios realizados como es el caso de la Escuela Politécnica del Ejército, Centro de Estudios de Desarrollo de Software, que realizó un estudio donde se comprobó que las empresas no cumplen con la aplicación de metodologías para el desarrollo de proyectos (Palacios, 2014). Se utilizan metodologías ágiles sin estructuración y planificación previa, los encargados de los proyectos no tienen definidos sus roles y funciones ni trabajan con modelos adecuados que posibiliten controlar cambios, modificaciones en el alcance, planificación y seguimiento de la calidad del producto final. No están enfocados en la rentabilidad del proyecto, falta definición en las funciones y roles dentro del equipo de desarrollo de

software, en los procedimientos y métodos de seguimiento de proyectos, insuficiente precisión y definición en los entregables, poca o nula gestión de riesgos (Palacios, 2014).

La presente investigación, establece un enfoque holístico para vincular las mejores técnicas, herramientas, procedimientos y normas del *Project Management Institute* (PMBOK), con las metodologías ágiles de desarrollo de software, para mayor eficiencia en la gestión de proyectos. En este sentido, el aporte científico de la investigación está precisamente, en la propuesta de un modelo para la integración del PMBOK a las metodologías ágiles como apoyo a la dirección de proyectos de desarrollo de software, realizando el análisis de la situación actual, el estudio teórico conceptual de la guía del PMBOK y los métodos ágiles, así como el análisis de los criterios de selección de las metodologías.

La propuesta se comprueba al implementarse en un proceso de gestión de proyectos si se considera que como resultados esperados, la gestión de proyectos debe ser estructurada, planificada en base a la optimización de tiempos y recursos.

DECLARACIÓN Y AUTORIZACIÓN

Yo: **LUISA MARIBEL URIBE CAMPAÑA**, con **CC. 050197231-9**, autora del trabajo de graduación intitulado: "**Modelo de Integración del *Project Management Body of Knowledge* con Metodologías Ágiles de Desarrollo de Software**", previa a la obtención del título profesional de Magister en Gerencia Informática, en la escuela de **Ingeniería en Sistemas**.

- 1.- Declaro tener pleno conocimiento de la obligación que tiene la Pontificia Universidad Católica del Ecuador, de conformidad con el artículo 144 de la Ley Orgánica de Educación Superior, de entregar a la SENESCYT en formato digital una copia del referido trabajo de graduación para que sea integrado al Sistema Nacional de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor.
- 2.- Autorizo a la Pontificia Universidad Católica del Ecuador a difundir a través de sitio web de la Biblioteca de la PUCE Ambato, el referido trabajo de graduación, respetando las políticas de propiedad intelectual de Universidad.

Ambato, abril 2018



LUISA MARIBEL URIBE CAMPAÑA

CC. 050197231-9



Dedicatoria

Este trabajo está dedicado a Dios, por la segunda oportunidad de vida que me ha dado. A mi hija, por ser mi razón de existir, mi fuente de motivación y de lucha constante. A mi esposo y mi madre, quienes me brindaron su amor y apoyo para poder culminar con éxito este proyecto. A mi hermano, quien me impulso a seguir adelante y conquistar este reto.

Reconocimientos

En reconocimiento a la Pontificia Universidad Católica del Ecuador, Sede Ambato, a sus autoridades, personal administrativo y personal docente, que en el transcurso de la carrera me han apoyado de forma personal y técnica para el cumplimiento de esta reto. A la empresa ADS SOFTWARE CIA LTDA, y todo su personal, por la apertura y amabilidad para el desarrollo de la presente investigación. De manera muy especial al Ing. Darío Robayo Mg., tutor del proyecto, por su conocimiento, paciencia y orientación en la investigación, lo que me ha permitido terminar con éxito el mencionado trabajo y así alcanzar mi meta tan deseada.

RESUMEN

El objetivo es validar un *Modelo de Integración del Project Management Body of Knowledge* con las Metodologías Ágiles de Desarrollo de Software que permita a las pequeñas y medianas empresas de la región central del Ecuador, dedicadas al desarrollo de Software, el mejoramiento de la calidad y gestión de los proyectos. Las limitaciones de la gestión de proyectos, que inciden en el no cumplimiento de requerimientos, tiempo de duración, presupuesto y calidad, aplicación de metodologías sin estructuración y planificación previa; falta de definición de roles y funciones, así como trabajar con modelos no adecuados que posibiliten controlar cambios, en el alcance, planificación y seguimiento de la calidad del producto final. Se utilizó la metodología cuantitativa, que permitió la realización del estudio diagnóstico obteniendo como resultados, que el 40 % de las empresas encuestadas, afirman que ocasionalmente la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles, lo que evidencia que aún las empresas no se encuentran familiarizadas con la gestión de proyectos. El modelo de integración propuesto, al implementarse en un proceso de gestión de proyectos permitió la optimización de tiempos y recursos, calidad, estructuración y planificación óptima. No se pretende estandarizar la gestión de proyectos en las empresas que conforman la población y muestra a partir del modelo propuesto, si se considera que no existe asociatividad entre ellas, no se integran experiencias y conocimientos en la región.

Palabras Claves: Guía, Integración, PMBOOK, Metodologías ágiles, Proyectos.

ABSTRACT

The objective of this work is to validate Project Management Body of Knowledge, an integration model with Agile Methodologies for Software Development that allow small and medium companies of the central area of Ecuador to improve projects' management and quality. Projects' management is limited due to the lack of compliance with the requirements, the project's duration, budget and quality; the methodology's application without planning and structure, the lack of roles and duties as well as working with unsuitable models that could enable to monitor any changes in the planning, follow-up and scope of the final's product quality. Quantitative methodology was applied to develop a diagnostic assessment with the results that 40% of the surveyed companies agree that they use PMBOK guide 5th edition occasionally, in addition to manage the product's cycle during the project, which could provide an added value such as documentation improvement and monitoring through agile methods if they are well managed. This is proof that the companies are not familiar yet with project management. The integration model that is proposed which was run by means of a project's management process allowed to optimize time, resources, quality, structuring and better planning. This work is not aimed at standardizing the company's project management that are part of the population and sample sizes based on the proposed model, since there is no relationship between them nor experiences or knowledge of the area.

Keywords: Guide, Integration, PMBOOK, Agile Methodologies, Projects.



Luisa Maribel Uribe Campaña
MGI-136



Pontificia Universidad
Católica del Ecuador

CERTIFICACIÓN

Una vez revisado el "Abstract" del Proyecto de Investigación y Desarrollo con el tema: Modelo de Integración del Project Management Body of Knowledge con Metodologías Ágiles de Desarrollo de Software, presentado por LUISA MARIBEL URIBE CAMPAÑA previo a la obtención del título de Magister en Gerencia Informática, certifico que el mismo guarda concordancia con el contenido del texto en español.

Ingeniero, *MBA.*

Eduardo Hong

Director de la Escuela de Lenguas y Lingüística



Pontificia Universidad
Católica del Ecuador

DIRECCIÓN
ESCUELA DE
LENGUAS Y LINGÜÍSTICA

TABLA DE CONTENIDOS

Ficha Técnica	iii
Declaración y Autorización.....	v
Dedicatoria	vi
Reconocimientos	vii
Resumen	viii
Abstract.....	ix
Tabla de contenidos	xi
Lista de Tablas	xiii
Lista de Figuras	xiii
CAPÍTULOS	
1. Introducción.....	1
1.1. Presentación del trabajo	2
1.2. Descripción del Documento.....	3
2. Planteamiento de la Propuesta de Trabajo	4
2.1. Información técnica básica	4
2.2. Descripción del problema.....	4
2.3. Preguntas Básicas	6
2.4. Formulación de Metas	6
2.5. Formulación de hipótesis	6
2.6 Objetivos	6
2.7. Delimitación funcional	8
3. Marco Teórico	9
3.1. Antecedentes	9
3.2. Programación extrema (XP).....	13
3.3 Scrum.....	21

3.4 Diferencias entre Scrum y Xp.....	36
3.5 Esfuerzo de adaptación de la metodología.....	36
3.6 Elementos de la Guía del PMBOK determinantes para su integración a las metodologías ágiles	39
3.7 Estado del Arte.....	43
4. Metodología	46
4.1. Métodos aplicados.....	46
4.2. Población y muestra	47
4.3. Materiales y Herramientas.....	48
4.4. Procedimientos.....	489
4.5 Diagnóstico	56
5. Resultados	79
5.1. Producto final del proyecto de titulación.....	79
5.2. Evaluación preliminar	91
5.3. Regulaciones y normativas para la puesta en práctica de la guía de integración.....	98
5.4. Análisis de resultados.....	118
6. Conclusiones y Recomendaciones	120
6.2. Conclusiones	123
6.2. Recomendaciones.....	1233
APÉNDICES	
Apéndice A.- Encuesta aplicada a los directores o personal encargado del desarrollo de proyectos de las pequeñas y medianas empresas de desarrollo de software de la región central del Ecuador	124
Referencias.....	131

Lista de Tablas

Comparación de metodologías ágiles	38
Procesos y subprocesos, ciclo de vida del proyecto.....	41
Parámetros de integración.....	50
Denominación, ubicación de las empresa.....	57
Años desde su creación.....	57
Experiencia profesional.....	57
Organización del proyecto	58
Actividades técnicas y de gestión	59
Etapa de gestión	59
Etapa técnica.....	60
Priorización en la empresa	610
Guía del PMBOK	621
Sustitución por actividades de la guía del PMBOK.....	62
Argumentación	63
Ciclo del proyecto	63
EDT	643
Procesos de inicio y planificación.....	64
Actividades propias del producto.....	654
Procesos de cierre del proyecto	65
Captura de requerimientos.....	665
Captura de requerimientos iniciales	66
Planificación del alcance	66
Frecuencias.....	67
Priorización en la empresa	733
Evaluación.....	74
Actividades de gestión	74

Gestión Ciclo de Vida.....	75
Utilización de metodologías ágiles	75
Metodologías más utilizadas	765
Frecuencia de actividades	76
Procesos	76
XP. Consideraciones	77
Requisitos.....	77
Procesos actuales en la empresa.....	91
Entregables solicitados para la evaluación	95
Acta de constitución del proyecto	101
Proceso de identificación de los interesados	102
Proceso de planificación.....	104
Planificación del alcance	105
Priorización de historias de usuarios	105
Proceso para planificar el Sprint.....	106
Desarrollo del cronograma	106
Planificación de costos.....	107
Proceso para planificar la calidad del proyecto	108
Roles y responsabilidades	108
Ejecución.....	109
Seguimiento y control	109
Áreas de estudio	110
Análisis comparativo. Metodologías ágiles vs tradicionales.....	112
Enfoques entre metodologías	113
Análisis comparativo	118

Lista de Figuras

1. Áreas de conocimiento del PMBOK.....	49
2. Grupo de procesos para la integración	80
3. Áreas de conocimiento del PMBOK para la integración.....	80
4. Aspectos fundamentales.....	81
5. Principios a seguir para lograr la integración del PMBOK a las metodologías ágiles. Manual de uso y aplicación.....	85
6. Características que predominan en el proceso de desarrollo ágil.....	86
7. Gestión de Proyectos.....	88
8. Parámetros que se considera para la integración del MODELO.....	88
9. Parámetros de metodologías ágiles para la integración del MODELO.....	89
10. Procesos para el manejo de un proyecto de Software.....	90
11. Problemas identificados.....	92
12. Grupo de procesos de la guía del PMBOK.....	95

Capítulo 1

Introducción

A la hora de diseñar métodos de negocios que requieran una alta efectividad al momento de ser aplicados, el desarrollo de proyectos de software desde el punto de vista de la Ingeniería de Sistemas, es sumamente importante para la organización y optimización de las actividades llevadas a cabo por una empresa, por lo cual es necesaria la aplicación de diferentes procesos ágiles de desarrollo de software y gestión de proyectos.

Existen numerosas propuestas de metodologías para desarrollar software. Tradicionalmente estas metodologías se centran en el control del proceso, definición y establecimiento de las actividades, herramientas y notaciones al respecto, dado estas reglas, estas metodologías se caracterizan por no ser flexibles y en la mayoría de los casos, se ajustan a los archivos que se generan en función del control de actividades. Es por ello que se considera no es adecuado para muchos de los proyectos actuales por los cambios que acontecen en el contexto de desarrollo de software, que exige trabajar en menor tiempo sin perder la integridad de los productos. En este escenario, las metodologías ágiles constituyen una alternativa viable para contrarrestar las dificultades antes mencionadas.

Los objetivos de las metodologías ágiles, entre los cuales se destaca la preferencia de algunos valores por sobre otros, por ejemplo: individuos e interacciones, sobre procesos y herramientas, software operativo, sobre documentación extensiva y colaboración con el cliente, sobre negociación de contratos.

El aplicar metodologías tradicionales obliga a forzar a los clientes, a que tomen la mayoría de las decisiones en el alcance del producto software en la fase inicial, porque no han sido desarrolladas para trabajar con incertidumbre; luego el coste del cambio de las decisiones puede llegar a ser elevado. El control del proceso es mediante una rigurosa definición de roles, actividades, artefactos, incluyendo modelado y documentación detallada.

Para evitar esto, surgieron otras metodologías, que se adecuan más a las nuevas condicionantes del contexto en cuanto al diseño de proyectos de desarrollo de software, llamadas metodologías ágiles. La diferencia inmediata radica en mantener una buena relación con el cliente, es menos orientada a la documentación. Lo importante son los procedimientos, enfocados al desarrollo de las habilidades del equipo.

Las características de los proyectos para los cuales las metodologías ágiles han sido especialmente pensadas se ajustan a un amplio rango de proyectos de desarrollo de software como aplicaciones de gestión, aplicaciones en la web; “aquellos en los cuales los equipos de desarrollo son pequeños, con plazos reducidos, requisitos volátiles, y/o basados en nuevas tecnologías.

Estos avances suponen un cambio tanto en la concepción de los sistemas a desarrollar como en la forma de interactuar con estos por parte de los usuarios finales, por lo que es necesario conocer qué metodología ágil es la más idónea a ser aplicada en el desarrollo de proyectos de software si se considera la metodología para el diseño, seguimiento, sistematización y evaluación de los proyectos en sí mismos, por lo que es necesario concatenar ambas metodologías.

1.1. Presentación del trabajo

La presente investigación, establece un enfoque holístico para vincular las mejores técnicas, herramientas, procedimientos y normas del *Project Management (PMBOK)*, definido por el Project Management Institute (PMI) con las metodologías ágiles de desarrollo de software, para mayor eficiencia en la gestión de proyectos. En este sentido, el aporte científico de la investigación está precisamente, en la propuesta de un modelo para la integración del PMBOK a las metodologías ágiles como apoyo a la dirección de proyectos de desarrollo de software, realizando el análisis de la situación actual, el estudio teórico conceptual de la guía del PMBOK y los métodos ágiles, así como el análisis de los criterios de selección de las metodologías.

La propuesta se comprueba al implementarse en un proceso de gestión de proyectos si se considera que como resultados esperados, la gestión de proyectos debe ser estructurada, planificada en base a la optimización de tiempos y recursos.

1.2. Descripción del Documento

En el Capítulo 1 se establece la introducción del trabajo, la presentación y estructuración capitular del proyecto.

En el Capítulo 2 se plantea la propuesta de trabajo.

El Marco Teórico es abordado en el Capítulo 3; en particular, la Sección 3.1 está dedicada a definiciones y conceptos, en tanto que la Sección 3.2 permite establecer el estado del arte.

En el Capítulo 4 se presenta la Metodología; partiendo de la etapa de Diagnóstico (Sección 4.1), pasando por los Métodos particulares aplicados (Sección 4.2) para llegar a la descripción de Población y Muestra (Sección 4.4).

El Capítulo 5 está dedicado a la Presentación y Análisis de los Resultados del trabajo.

Las Conclusiones y Recomendaciones son materia del Capítulo 6.

El trabajo está complementado por tres Apéndices. El Apéndice A está reservado para Procedimientos Detallados. Por otro lado, algunos criterios adicionales para presentación del IFP se enuncian en el Apéndice B.

Capítulo 2

Planteamiento de la Propuesta de Trabajo

En esta sección se analiza la propuesta de trabajo, información técnica básica, la problematización, además del planteamiento de los objetivos esperados.

2.1. Información técnica básica

Tema: Modelo de Integración del *Project Management Body of Knowledge* con las Metodologías Ágiles de Desarrollo de Software

Tipo de trabajo: Proyecto de Investigación y Desarrollo

Clasificación técnica del trabajo: Desarrollo

Líneas de Investigación, Innovación y Desarrollo

Principal: Ingeniería de Software y/o Plataformas Educativas

2.2. Descripción del problema

En los últimos años, en la zona centro del país, se observa un creciente aumento de pequeñas y medianas empresas (PYMES) de desarrollo de software según registros de la Superintendencia de Compañías y el Servicio de Rentas Internas (Superintendencia de Bancos, Valores y Seguros; Cámara de Comercio Latacunga; Servicio de Rentas Internas, 2015) donde se crean aplicaciones informáticas, así como software personalizados en correspondencia a las necesidades y requerimientos de cada cliente en particular. No obstante, muestran diversos problemas en torno a la gestión de proyectos.

Los problemas que se presentan, a criterio de los gerentes de proyectos de desarrollo de software, que han accedido a brindar información para la presente investigación, radican en que no existen parámetros que muestren qué métodos utilizar, por lo que se utilizan metodologías ágiles sin

estructuración y planificación previa, siendo difícil poder identificar cuál es la que se utiliza (*Scrum*, *Extreme Programming (Xp)*, entre otras); se presentan problemas en el levantamiento de requerimientos: los encargados de los proyectos no tienen definidos sus roles y funciones y generalmente se trabaja sin modelos adecuados que posibiliten controlar cambios, modificaciones en el alcance, planificación y seguimiento de la calidad del producto del proyecto.

Las empresas no trabajan en la mayoría de los casos, enfocadas en la rentabilidad del proyecto, falta definición en las tareas y asignaciones dentro del equipo de desarrollo de software, en los procedimientos y métodos de seguimiento de proyectos. Todo ello ha permitido corroborar las necesidades de mejora de la dirección de proyectos en estas empresas.

Debido a que no se utiliza una metodología y herramientas definidas, para la administración y ejecución de proyectos, su manejo depende más de la habilidad y conocimiento del Gestor que lidera la iniciativa. Tampoco se cuenta con herramientas que permitan reportar el avance y estado del proyecto con soportes objetivos, se comuniquen los riesgos y su forma de mitigarlos, se controlen los cambios, se consolide y unifique el acervo de procesos de la organización, se documente la información del proyecto, así como las lecciones aprendidas para que en proyectos futuros no se vuelvan a cometer los mismos errores o se tomen las buenas experiencias vividas y se implementen en proyectos por desarrollar (Palacios, 2014).

El estudio diagnóstico en las empresas de la región central del Ecuador, evidencia la presencia del problema, las brechas existentes y las necesidades de mejoras en la dirección de proyectos que ocasiona que varios de estos proyectos terminen fuera de parámetros aceptables con respecto al alcance, tiempo, costo, calidad, riesgos y recursos humanos y los que lideran proyectos, reconocen que no existe una definición única que indique a los gerentes de proyectos que estándares o métodos utilizar debido a que cada proyecto tiene sus propias particularidades.

En tal sentido, se plantea como problemática fundamental, cómo mejorar la gestión de los proyectos de desarrollo de software es en las PYMES de la zona centro de Ecuador.

2.3. Preguntas Básicas

¿Por qué se origina?

En el estudio diagnóstico realizado en las empresas de desarrollo de software de la región central del Ecuador, se evidencia las necesidades de mejoras en la dirección de proyectos que ocasiona que varios de éstos terminen fuera de parámetros aceptables con respecto al alcance, tiempo, costo, calidad, riesgos y talento humano y los que lideran proyectos, reconocen que no existe una definición única que indique a los gerentes de proyectos que estándares o métodos utilizar debido a que cada proyecto tiene sus propias particularidades.

¿Dónde se origina? La problemática a investigar se presenta en las pequeñas y medianas empresas de desarrollo de software de la región central del Ecuador y un conjunto de empresas ubicadas entre las ciudades de Quito, Ambato y Latacunga.

2.4. Formulación de metas

Entregar un modelo de integración de metodologías ágiles a la guía del PMBOK, que permita el fortalecimiento de la sistematización del ciclo de vida del proyecto, planificación óptima en la gestión de proyectos y la facilidad para definir los entregables del proyecto.

2.5. Formulación de hipótesis

El diseño de un modelo de integración entre los aspectos fundamentados en la Guía del PMBOK Quinta Edición y las metodologías ágiles, contribuirá a la gestión eficiente de proyectos de desarrollo de software en las pequeñas y medianas empresas de Software de la región central de Ecuador.

2.6 Objetivos

Objetivo General

Validar un *Modelo de Integración del Project Management Body of Knowledge* con las Metodologías Ágiles de Desarrollo de Software que permita a las pequeñas y medianas empresas de la región central del Ecuador, dedicadas al desarrollo de Software, el mejoramiento de la calidad y gestión de los proyectos.

Objetivos Específicos.

1. Fundamentar las teorías y prácticas aplicadas por varios autores relacionados con las áreas de conocimiento del Project Management Body of Knowledge y las metodologías ágiles para la gestión de proyectos de software.
2. Diagnosticar la situación actual de la dirección de proyectos de desarrollo de software para las diez empresas de desarrollo de Software que conforman la muestra del estudio, pertenecientes a la región central del Ecuador.
3. Identificar componentes o factores que influyen en la evaluación de metodologías aplicadas en el desarrollo de software y las áreas de conocimiento del Project Management Body of Knowledge para la gestión de proyectos de software.
4. Determinar los parámetros de integración entre el Project Management Body of Knowledge y los métodos ágiles que permitan la sistematización del ciclo de vida de los proyectos de desarrollo de software y formular el modelo de integración de ambas metodologías.
5. Desarrollar la matriz de evaluación de metodologías ágiles para los proyectos de desarrollo de software y la aplicación del modelo.
6. Evaluar la factibilidad del modelo de integración en un proceso de gestión de proyectos en las empresas que conforman la muestra.

2.6. Delimitación funcional

Pregunta 1. ¿Qué será capaz de hacer el producto final del trabajo de titulación?

Las estrategias que se proponen están orientadas mayormente a:

El Modelo de integración que se propone, permitirá a las organizaciones:

- La gestión eficiente de proyectos
- Planificación adecuada del ciclo de vida de los proyectos
- Determinar roles y funciones
- Medir tiempos
- Mantener un equilibrio entre desarrollo de actividades de gestión y actividades técnicas.

El modelo que se propone se pretende ejecutar en una empresa de las que conforman la muestra, a partir de la selección de un proceso de gestión de proyectos, evaluándose la factibilidad de los parámetros de integración para el desarrollo de software.

De igual manera, el estudio diagnóstico en las empresas permitió hacer un análisis de la problemática a investigar, en torno a la necesidad de que los gerentes de proyectos cuenten con herramientas, guías que les permitan orientar su trabajo y gestionar de forma eficiente el desarrollo de proyectos, bajo nuevas perspectivas metodológicas.

2.6.1. ¿Qué no será capaz de hacer el producto final del proyecto de titulación?

Estandarizar la gestión de proyectos en las empresas que conforman la población y muestra a partir de los modelos propuestos, si se considera que no existe asociatividad entre las empresas, no se integran experiencias y conocimientos en la región.

Capítulo 3

Marco Teórico

3.1. Antecedentes

En febrero de 2001, existen registros de que aparece el término ágil aplicado al desarrollo de software (Moyota, 2014). Su objetivo fue perfilar los valores y los principios que deberían permitir a los equipos de desarrollo, crear productos de software y que estos equipos fueran capaces de asimilar rápidamente modificaciones que se presentaran y posibilitar el manejo de la incertidumbre. Sus propósitos eran precisamente, ofrecer una herramienta distinta a los procesos tradicionales, caracterizados por ser muy estructurados, sin holguras y análisis después de contratado el proyecto sistematizados por la documentación que se genera en cada una de las actividades desarrolladas.

3.1.1 Fundamentos de las Metodologías Ágiles

El “Manifiesto Ágil” (2001), establece los principios sobre los que se basan las metodologías llamadas ágiles:

- El contacto directo con los clientes, el seguimiento y relación más allá de los procesos y las herramientas.
- El funcionamiento óptimo del producto software, sin considerar mayormente la documentación generada.
- La colaboración con el cliente, el contacto directo no solo en el momento de acuerdos y contratos.
- La respuesta al cambio, no solo en función de cumplimientos de esquemas de trabajo (Rico, 2009).

3.1.1.1 Se valora más a los individuos y su interacción que a los procesos y las herramientas.

Este es posiblemente el principio más importante del manifiesto.

Las metodologías ágiles son más adaptables a las situaciones concretas bajo las cuales se desarrollan proyectos de software, las cuales se centran y dan mayor relevancia al cliente, a la persona que dirige el proyecto, a su interacción

con el cliente y al desarrollo con iteraciones relativamente cortas. Ello hace que a pesar de cambios en las decisiones y requerimientos de los clientes no afecte la calidad del producto ni tampoco en los casos en los cuales se considera necesario reducir tiempos (Penadés, 2008).

Las metodologías tradicionales para el desarrollo de software, al limitar el contacto con el cliente, no respondieron a las exigencias de los momentos actuales donde es necesario centrarse en el conocimiento y el talento de las personas que posibiliten adecuaciones al proyecto hasta el momento de la entrega final, que responda a los intereses y necesidades del cliente.

Los procesos deben ser una ayuda y un soporte para guiar el trabajo. Deben adaptarse a la organización, a los equipos y a las personas; y no al revés. La priorización de procesos, sin tener en cuenta a las personas y su calidad de trabajo, es un riesgo y una alternativa no viable.

3.1.1.2 Se valora más el software que funciona que la documentación exhaustiva.

Considerar con anticipación, el funcionamiento de prototipos o productos de desarrollo, es importante, genera procesos de retroalimentación, posibilidades que difícilmente se podrían incluir en la construcción de un documento de requisitos detallados sin haber comenzado el proyecto. De ahí que tenga un rol más importante el flujo de trabajo, el manejo y desarrollo de los equipos de trabajo, su progreso, y realizar una retroalimentación constante para las mejoras en la gestión de proyectos.

El manifiesto no afirma que no sea necesario documentar el proceso. Los documentos permiten evidenciar el trabajo realizado, evaluar y sistematizar la información, pero para las metodologías ágiles son menos importantes que los productos que funcionan. Menos trascendentales y que no aportan valor al producto, más que las iteraciones e intercambios directos con los clientes.

Los documentos no pueden sustituir, ni pueden ofrecer la riqueza y generación de valor que se logra con la comunicación directa entre las personas y a través de la interacción con los prototipos. Por eso, siempre que sea posible debe preferirse reducir al mínimo indispensable el uso de documentación, que genera trabajo que no aporta un valor directo al producto.

La organización y los equipos deben tener una concepción diferente, donde no se estandarice la comunicación a través de informes o generación de archivos, más allá de estas perspectivas, es el valor de conocer directamente los intereses, necesidades del cliente, para evitar requisitos mal comprendidos, que el producto final no cumpla con los requisitos planificados al inicio en la etapa contractual, así como contrarrestar defectos que tenga el producto final.

3.1.1.3 Se Valora más la colaboración con el cliente que la negociación contractual.

Las prácticas ágiles están especialmente enfocadas en desarrollo de productos donde sus requisitos no se establecen con exactitud desde el inicio del proyecto, o que aún cuando pudieran definirse, su aporte y calidad final sería mínima si se limita el desarrollo incremental del software con iteraciones muy cortas mencionado con anterioridad. También para los casos en los que los requisitos van a ser muy inestables, por la velocidad del entorno de negocio del cliente.

Para el desarrollo ágil el valor del resultado no es consecuencia de haber controlado una ejecución conforme a procesos, sino de haber sido implementado directamente sobre el producto y el contacto con el cliente así como el manejo del equipo de trabajo (Penadés, 2008).

Un contrato no es lo único que aporta valor al producto. Constituyen formalidades que lejos de integrar al cliente, simplemente establecen patrones de distancia y cumplimiento de responsabilidades. En cambio, en el desarrollo ágil el cliente es un miembro más del equipo, que se integra y colabora en el grupo de trabajo.

3.1.1.4 Se Valora más la respuesta al cambio que el seguimiento de un plan

Para un modelo de desarrollo que surge de entornos inestables, que mayormente se centra en la posibilidad del cambio y la evolución rápida y continua, resulta mucho más factible tener en cuenta la capacidad de respuesta que el seguir ya documentos establecidos con anterioridad. Los principales valores de la gestión ágil son la anticipación y la adaptación; diferentes a los de la gestión de proyectos tradicionales donde se consideró que la

planificación y el control evitarían errores, sin tener en cuenta el ajuste a requisitos, a tiempos, a modificaciones a lo largo del desarrollo de software.

El mercado actual es altamente competitivo y cambiante. En ese contexto, el desarrollo de software busca básicamente rapidez, calidad y reducción de costos en la ejecución de sus proyectos. Para asumir estos retos, es necesario tener agilidad y flexibilidad. Estas características se constituyen en el fundamento mismo de las metodologías ágiles de desarrollo.

En el ámbito de las metodologías de desarrollo de software existe un gran número de alternativas y los responsables de cada proyecto tienen la difícil tarea de seleccionar la alternativa que mejor se ajuste a sus necesidades y recursos.

Las metodologías ágiles están especialmente orientadas a proyectos que necesitan soluciones a medida, con una elevada simplificación en términos de tiempo y recursos, sin dejar de lado el aseguramiento de la calidad del producto.

Al ser reciente las metodologías ágiles (años 90), no han sido estudiadas suficientemente por la comunidad académica, generando desconocimiento de sus bondades. En el ámbito nacional, no se han realizado muchos estudios sobre el tema y menos aplicaciones prácticas que involucren el paradigma de las metodologías ágiles. Por tal motivo, se plantea la necesidad de explorar a mayor profundidad dicho campo a través de ejercicios prácticos debidamente documentados. Es necesario analizar aspectos claves como las características de los clientes, detalles técnicos de las soluciones y prestar especial atención a la naturaleza cambiante de los requerimientos.

En los postulados de las metodologías ágiles, se observa la intención de rebelarse contra los vicios típicos de los procesos tradicionales de desarrollo de software: planes que no se cumplen o que se extienden más allá de los plazos, documentación que exige demasiado trabajo de elaboración y no refleja la realidad del producto final, contratos que terminan perjudicando a una de las dos partes (desarrollador o cliente) o a ambas, entre otros males.

A continuación se resumen algunas metodologías ágiles. La mayoría de ellas han sido utilizadas con éxito en proyectos reales pero disponen de poca difusión o reconocimiento.

3.2 Programación Extrema (XP)

La programación extrema (del inglés Extreme Programming) es uno de los llamados procesos o metodologías ágiles de desarrollo de software. Consiste en un conjunto de prácticas que a lo largo de los años han demostrado ser las mejores prácticas de desarrollo de software llevadas al extremo, fundamentadas en los valores de las metodologías ágiles.

XP se basa en determinados principios básicos:

3.2.1 Retroalimentación a escala fina

El principio de pruebas: se tiene que establecer un período de pruebas de aceptación del programa (llamado también período de caja negra) a partir del levantamiento de requerimientos, problemas que puedan existir en la etapa de inicio así como el diseño necesario para evitar cambios en el futuro. Es muy recomendable las simulaciones del sistema en funcionamiento (Sierra, 2008).

3.2.2 Proceso de planificación

La planeación es la etapa inicial de todo proyecto en XP. En esta fase, el usuario tendrá que escribir sus necesidades, siendo considerado parte del equipo de trabajo, definiendo las actividades y requerimientos porque esta metodología, mayormente se centra en quienes producen y utilizan el producto software. De ahí que se orienta a reducir el coste de cambio, a través de la retroalimentación continua.

Son muy importantes y tienen que ser una constante las reuniones periódicas durante esta fase de planificación, así como durante el desarrollo del producto. Los encuentros pueden ser a diario, con todo el equipo de desarrollo para identificar problemas, proponer soluciones y señalar aquellos puntos a los que se les ha de dar más importancia por su dificultad o por su punto crítico, por lo que reduce la generación de documentación, principios burocráticos y los desarrolladores se centran más en la programación (Aguilar, 2008).

Desde estas perspectivas, quedan determinadas la cantidad de iteraciones y su duración y se plantean diseños que respondan únicamente al producto que se encuentra desarrollándose y a sus características.

3.2.3 El cliente en el sitio

El cliente juega un rol importante y activo sobre todo en poder definir los requerimientos de su producto, su funcionalidad y estructuración; señalar las prioridades y dar respuesta a inquietudes e interrogantes del equipo de trabajo. Esta fuerte interacción cara a cara con el programador disminuye el tiempo de comunicación y la documentación exhaustiva y mayormente convencional, junto con los altos costes de su creación y mantenimiento. Este representante del cliente estará con el equipo de trabajo durante toda la realización del proyecto con retroalimentaciones sistemáticas (Aguilar, 2008).

3.2.4 Programación en parejas

Requiere que todos los programadores XP escriban su código en parejas, compartiendo una sola máquina, en función de generar mayor calidad en las aplicaciones, además de hacer énfasis en la necesidad de ser flexibles, adaptables a los cambios, más que seguir un plan cerrado y diseñado a priori.

3.2.5 Proceso continuo en lugar de por lotes.

1. Integración continua: permite al equipo hacer un rápido progreso a partir de la integración colaborativa en el trabajo con el cliente. En lugar de crear *builds* (o versiones) estables teniendo en cuenta planificaciones previas y diseños al momento del contrato, los equipos de programadores XP pueden hacer modificaciones continuamente. Esto reduce los problemas de integración comunes en proyectos largos y estilo cascada.

2. Refactorización: favorece cambios y modificaciones en el diseño del sistema de forma transversal, a través de todo el proceso de desarrollo. Los programadores evalúan continuamente el diseño y recodifican lo necesario a partir de iteraciones cortas.

3. Entregas pequeñas: colocan un sistema sencillo en producción rápidamente que se actualiza de forma rápida y constante permitiendo que el verdadero valor de los productos pueda ser evaluado sistémicamente y en períodos

cortos de tiempo. Estas entregas no pueden pasar las 2 o 3 semanas como máximo (Moyota, 2014).

3.2.6 Entendimiento compartido.

1. Diseño simple: se sustenta en la idea que el valor de un producto no depende de la complejidad del desarrollo, sino que los mejores productos son entregados a partir del programa más sencillo que cumpla los requerimientos. *Simple Design*, se enfoca en proporcionar un sistema que cubra las necesidades inmediatas del cliente. Este proceso permite eliminar obsoletismos y elementos que ya no se ajusten a la realidad.

2. Metáfora: desarrollada por los programadores al inicio del proyecto, define una historia de cómo funciona el sistema completo. XP estimula historias, que son breves descripciones de un trabajo, de un sistema en lugar de los tradicionales diagramas y modelos UML (*Unified Modeling Language*). La metáfora expresa sin dudas, la visión evolutiva del proyecto que define el alcance y propósito del sistema.

Las tarjetas CRC (Clase, Responsabilidad y Colaboración) también favorecen el diseño de actividades durante el desarrollo del producto. Cada tarjeta representa una clase en la programación orientada a objetos y define sus responsabilidades (lo que ha de hacer) y las colaboraciones con las otras clases (cómo se comunica con ellas) (Aguilar, 2008).

3. Propiedad colectiva del código: un código con propiedad compartida. Mayormente el equipo tiene un rol determinante, no se establecen direccionamientos de forma individual, aspecto que difiere de las metodologías tradicionales y se orienta a la integración y trabajo en equipo. Los defensores de XP argumentan que mientras haya más gente trabajando en el desarrollo de software, menos errores aparecerán.

4. Estándar de codificación: define la propiedad del código compartido así como las reglas para escribir y documentar el código y la comunicación entre diferentes piezas de código desarrolladas por diferentes equipos. Los programadores las han de seguir de tal manera, que el código en el sistema se vea como si hubiera estado escrito por una sola persona (Aguilar, 2008).

3.2.7 Bienestar del programador.

1. La semana de 40 horas: la programación extrema sostiene que los programadores cansados escriben código de menor calidad. Minimizar las horas extras y mantener los programadores frescos, generará códigos de mayor calidad, para evitar sobrecarga laboral, presión en cuanto a entrega de actividades y productos que condicione la aparición de estrés en los trabajadores.

El modo de operación que XP propone para el desarrollo de software es bastante flexible por lo que puede generar ciertas informalidades en el proyecto de desarrollo como la problemática referente a los requisitos, en la cual en el contexto de XP se reduce a la especificación y seguimiento de los historiales de usuario. El planteamiento que propone al respecto dicha metodología es muy sencillo, pero debido a la gran cantidad de historiales de usuario que puede tener un proyecto y a la volatilidad de los requisitos es que su gestión puede llegar a ser complicada.

3.2.8 Valores

Más que una metodología, XP se considera una disciplina, la cual está sostenida por valores y principios propios de las metodologías ágiles. Existen cuatro valores que cumplen su papel como pilares en el desarrollo de metodologías livianas:

La comunicación. En la metodología XP es muy importante que exista un ambiente de colaboración y comunicación al interior del equipo de desarrollo, así como en la interacción de este con el cliente. En XP la interacción con el cliente es tan estrecha, que es considerado parte del equipo de desarrollo.

La simplicidad. Este valor se aplica en todos los aspectos de la programación extrema. Desde diseños muy sencillos donde lo más relevante es la funcionalidad necesaria que requiere el cliente, hasta la simplificación del código mediante la refactorización del mismo. La programación XP no utiliza sus recursos para la realización de actividades complejas, solo se desarrolla lo que el cliente demanda, de la forma más sencilla.

La retroalimentación. Se presenta desde el comienzo del proyecto, ayuda a encaminarlo y darle forma. Esta se presenta en los dos sentidos, por parte del equipo de trabajo hacia el cliente, con el fin de brindarle información sobre la evolución del sistema y desde el cliente hacia el equipo en los aportes a la construcción del proyecto.

El coraje. El equipo de desarrollo debe estar preparado para enfrentarse a los continuos cambios que se presentarán en el transcurso de la actividad. Cada integrante debe tener el valor de exponer los problemas o dudas que halle en la realización del proyecto. Aún con estas variaciones, las jornadas de trabajo deben proporcionar el máximo rendimiento.

3.2.9 Prácticas

A partir de los valores se plantean una serie de prácticas que sirven de guía para los desarrolladores en esta metodología. Uno de los aspectos más importantes para XP son las doce reglas que se plantean, las cuales se caracterizan por su grado de simplicidad y por su enfoque en la practicidad, además de que cada regla se complementa con las demás. A continuación se realizará una breve descripción de las seis más representativas:

- El desarrollo está dirigido por pruebas, que permitan mitigar errores y riesgos al momento de la entrega final. El programador realiza pruebas dirigidas al funcionamiento de nuevas adiciones o módulos al sistema. El cliente de igual manera actúa y da seguimiento al desarrollo de pruebas, ya que con ayuda del *tester* se encarga de diseñar las pruebas de aceptación, cuyo propósito es verificar que las historias de los usuarios se hayan implementado correctamente y que se dé cumplimiento en tiempo a los requerimientos presentados.
- El juego de la planificación. Desde el comienzo del desarrollo se requiere que el grupo y el cliente tengan una visión general del proyecto, es decir, deben integrar criterios y llegar a consensos que faciliten el desarrollo del proyecto. En el transcurso del proyecto se realizan diferentes reuniones, con el fin de organizar las tareas e ideas que surgen tanto por parte del cliente como por el equipo y que

respondan a requerimientos y adecuaciones necesarias por cambios externos.

- Programación en parejas. XP propone que exista una pareja de programadores, como medida para aumentar la calidad del código. Esta práctica busca reducir los errores de codificación además del trabajo colaborativo entre programadores, si se tiene en cuenta que cada programador tiene establecido sus roles, como por ejemplo, un programador se encarga del funcionamiento óptimo del sistema y de los módulos mientras que el otro programador aprueba los códigos desarrollados y busca la forma de simplificarlos para no generar complejidades en el producto.
- Entregas pequeñas. En la programación extrema se realizan entregas constantes de módulos funcionales completos, de tal forma, que en todo momento el cliente obtiene diferentes etapas del producto y puede evaluar su funcionamiento. En XP no existe el desarrollo incompleto de una tarea, esta se ejecuta en su totalidad o no se hace.
- Diseño simple. Solo se realiza lo necesario ajustado a los requerimientos iniciales para que la aplicación cumpla con la funcionalidad requerida por el cliente sin elementos agregados que dificulten la integridad en cuanto a funcionamiento del producto. No es conveniente realizar diseños complejos que posiblemente no aporten soluciones claras al proyecto y que retrasen entregas por requerir más tiempo.
- Convenciones de código. La aplicación de estándares de programación al código fuente de la aplicación, permite que todas las personas que conforman el grupo de trabajo puedan entender y realizar modificaciones al código del sistema (Juntao, 2007).

3.2.10 Alcance.

La programación extrema genera conveniencia y es factible solo en ciertas situaciones debido a limitantes que posee sobre todo por el trabajo en equipo y la importancia del equipo de trabajo en el desarrollo del producto final. Mayormente puede considerarse su aplicación en casos en que se requieran

resultados positivos en proyectos de mediana y pequeña envergadura, complejidad de requerimientos, que no demanden equipos de trabajo mayores a 20 personas (Juntao, 2007).

También se debe tener en cuenta que XP mayormente debe ser utilizado en casos donde exista un ambiente cambiante que es medido y comprobado en los requerimientos de la aplicación. La metodología XP está encaminada hacia los desarrollos que requieren de cambios continuos en el transcurso de un proyecto. Por ende, debe ser utilizada y recomendada para proyectos en los cuales el costo de cambio no se incremente a medida que transcurre el ciclo de desarrollo del producto.

Los proyectos realizados bajo esta metodología cumplen con lo estrictamente necesario en su funcionalidad de acuerdo a diferentes perspectivas. En XP no es conveniente precipitarse o adelantarse a las tareas que se han establecido previamente sin el consentimiento del cliente, para evitar que exista complejidad en el sistema, alejándolo del concepto de simplicidad, si se tiene en cuenta que mayormente se debe trabajar de la mano con el cliente y de acuerdo a sus criterios y valoración.

La Programación Extrema (XP) es posiblemente el método ágil más conocido y ampliamente utilizado.

La programación extrema implica varias prácticas que se ajustan a los principios de los métodos ágiles:

- El desarrollo incremental se lleva a cabo a través de entregas del sistema pequeñas y frecuentes y por medio de un enfoque para la descripción de requerimientos basados en las historias de cliente o escenarios que pueden ser la base para el proceso de planificación.
- La participación del cliente se lleva a cabo a través del compromiso a tiempo completo del cliente en el equipo de desarrollo. Los representantes de los clientes participan en el desarrollo y son los responsables de definir las pruebas de aceptación del sistema.
- El interés en las personas, en vez de en los procesos, se lleva a cabo a través de la programación en parejas, la propiedad colectiva del código del

sistema, y un proceso de desarrollo sostenible que no implique excesivas jornadas de trabajo.

- El cambio se lleva a cabo a través de las entregas regulares del sistema, un desarrollo previamente probado y la integración continua.
- El mantenimiento de la simplicidad se lleva a cabo a través de la refactorización constante para mejorar la calidad del código y la utilización de diseños sencillos que no prevén cambios futuros en el sistema (Letelier, 2010).

En este sentido, se puede afirmar que la metodología XP, se diferencia de las metodologías tradicionales, principalmente en que pone más énfasis en la adaptabilidad que en el ajuste necesario a las planificaciones previas y a lo que se tuvo en cuenta desde la etapa contractual. Se aplica de manera dinámica durante el ciclo de vida del software y es capaz de adaptarse a los cambios de requisitos. Los individuos e interacciones son más importantes que los procesos y herramientas.

La gente es el principal factor de éxito de un proyecto de desarrollo de software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que este configure su propio entorno de desarrollo en base a sus necesidades, por lo que el hecho de que el software funcione es más importante que generar una documentación exhaustiva. La regla a seguir es no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante. Estos documentos deben ser cortos y centrarse en lo fundamental. La colaboración con el cliente es más importante que la negociación de contratos. Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito. La respuesta ante el cambio es más importante que el seguimiento de un plan.

3.3 Scrum

3.3.1 Caracterización

Es uno de los métodos de gestión de proyectos de la denominada filosofía ágil, que permite un manejo apropiado de las expectativas del cliente, basadas en resultados tangibles siendo uno de los métodos ágiles menos conocidos y utilizados.

Este método desarrollado por Ken Schwaber, Jeff Sutherland y Mike Beedle (2000) (Palacio, 2010), define una metodología para la gestión de proyectos, que ha obtenido importantes resultados. Sus principales características se pueden resumir que el desarrollo de software se realiza mediante iteraciones denominadas *sprints*, con una duración variable de hasta 30 días (Palacio, 2010). El resultado de cada sprint es un incremento ejecutable que se muestra al cliente. Al principio de cada sprint se establece una lista de requerimientos llamada *backlog*, que debe completarse cuando este finalice. A diario se realizan breves reuniones del equipo de desarrollo, en las que se exponen los avances y los problemas encontrados, y se señalan posibles caminos para resolverlos (la resolución detallada de estos problemas no debe determinarse durante la reunión, para mantener su brevedad).

Scrum es una metodología ágil de desarrollo de proyectos que aunque surgió como modelo para el desarrollo de productos tecnológicos, también se emplea en entornos que trabajan con requisitos inestables y que requieren rapidez y flexibilidad; situaciones frecuentes en el desarrollo de determinados sistemas de software. Es una metodología de desarrollo muy simple, que requiere trabajo duro, porque no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto.

Como método ágil:

- Es un modo de desarrollo adaptable, de igual manera que XP, no se centra mayormente en ser predictivo.
- Orientado a las personas, más que a los procesos y documentación.
- Emplea el modelo de diseño e implementación de forma incremental basado en iteraciones y revisiones, como la metodología XP.

- Comparte los principios estructurales del desarrollo ágil: a partir del concepto o visión de la necesidad del cliente, construye el producto de forma incremental a través de iteraciones breves que comprenden fases de especulación, exploración y revisión.

Scrum es un proceso en el que se aplican de manera regular un conjunto de mejores prácticas en función de fortalecer la integración y colaboración sobre todo al trabajar en equipo y de esta forma, lograr obtener el mejor resultado posible de un proyecto, que es la finalidad de todo emprendimiento, servicios o productos empresariales. Estas prácticas se apoyan unas a otras y su selección tiene origen en un estudio, en cuanto a la manera de trabajar de equipos altamente productivos.

Se comienza con la visión general del producto, haciendo énfasis en las etapas de mayor funcionalidad, y que pueden llevarse a cabo en un período de tiempo breve (según los casos pueden tener duraciones desde una semana hasta no más de dos meses). Cada uno de estos períodos de desarrollo, es una iteración que finaliza con la entrega de una parte (incremento) operativa del producto. Estas iteraciones son la base del desarrollo ágil, y Scrum gestiona su evolución en reuniones breves diarias, similar a lo expuesto según la metodología XP (Moyota, 2014).

En Scrum se realizan entregas parciales y sistematizadas en correspondencia con el resultado final del proyecto al cual se pretende llegar, de acuerdo al beneficio que representan para los clientes. De esta manera, Scrum es mayormente una metodología ágil, a diferencia de XP, que se enfoca en resolver y dar continuidad a proyectos en entornos complejos, en función de resultados a corto plazo, donde los requisitos son cambiantes o poco definidos, a diferencia también de los proyectos a los que se ajusta XP como metodología ágil; por tanto, la innovación, la competitividad y la productividad son fundamentales (Padilla, 2014).

Scrum también es utilizada como metodología que permite controlar los riesgos para no afectar los entregables en función de lo que el cliente requiere,

sobre todo riesgos en cuanto a desfases en las entregas, falta de calidad en los entregables o cambios en el presupuesto asignado, siendo importante analizar si es cuestión de que los equipos aún no se consolidan y se genera desmotivación entre sus miembros, que terminan renunciando; cuando es necesario identificar y solucionar ineficiencias sistemáticamente o cuando se quiere trabajar utilizando un proceso especializado en el desarrollo del producto (Moyota, 2014).

Independientemente del tipo de metodología que se utilice, cualquier desarrollo de software parte siempre de un mismo problema: conocer las necesidades de los clientes. Scrum, al igual que el resto de metodologías ágiles, pretende no centrar las tareas de desarrollo en un conjunto de requisitos formalmente definidos, sino que aboga por la incorporación del cliente como un miembro más del equipo de desarrollo. De este modo, no se considera el proceso de definición de requisitos como un fin dentro del desarrollo del proyecto, sino que los requisitos aparecen implícitamente dentro del contenido de las denominadas historias de usuario (Palacio, 2010).

Las historias de usuario son el elemento base que utiliza Scrum para describir las características que el usuario espera que tenga el software que se va a desarrollar. Por lo tanto, pueden incorporar tanto cuestiones relacionadas con las funciones del sistema como con cualquier otro aspecto del mismo (restricciones, rendimiento).

Las historias de usuario se presentan desde la perspectiva del usuario. Así, no se describen utilizando una terminología técnica sino que se escriben utilizando un lenguaje cercano al dominio de la aplicación que se está desarrollando de forma que sea comprensible por los clientes y por los desarrolladores. Las historias de usuario se construyen bajo un mismo esqueleto que centra el foco de las características del producto, por lo que primero se determina quién propone la historia de usuario, luego se describe la característica que se cubre con la historia de usuario y finalmente, se especifica la razón por la que dicha característica es necesaria.

La metodología Scrum asume que el proceso de desarrollo de software es impredecible, y lo trata como a una “caja negra” controlada, en vez de manejarlo como un proceso completamente definido.

Esta es una de las principales diferencias entre Scrum y otras metodologías, como los modelos de espiral o de cascada, en los cuales el proceso de desarrollo se define por completo desde el inicio. Por tratar de planificar el proceso en forma completa desde el principio, las metodologías tradicionales fallan al toparse con algunos problemas habituales del desarrollo de software, como la falta de comprensión de los requerimientos al empezar el proceso, el cambio en los requerimientos durante el proceso, o la dificultad para prever los resultados del uso de nuevas herramientas y tecnologías.

Todo producto de software, durante su creación, enfrenta un proceso complejo de desarrollo debido al ambiente dinámico. A mayor grado de complejidad mayor grado de flexibilidad se requerirá para lograr el éxito. Es entonces donde encaja a la perfección Scrum, en la medida que es como una caja negra donde seguir un proceso lineal, no es la regla. Por el contrario, se está listo para atacar cualquier eventualidad de manera inmediata durante el proceso adaptándose a la nueva realidad. Es ahí donde se encuentra el núcleo y fortaleza de dicha metodología.

3.3.2 Prácticas de la Metodología Scrum

Revisión de las Iteraciones: Al finalizar cada iteración (sprint) se tiene en cuenta la revisión y evaluación del proyecto. Es por tanto, la duración del sprint, el período máximo que se tarda en reconducir una desviación en el proyecto o en las circunstancias del producto, lo que limita los riesgos y errores a futuro.

Desarrollo incremental: Cada iteración define etapas del producto, que se pueden inspeccionar y evaluar y que pueden ser catalogadas como revisiones de las iteraciones.

Desarrollo evolutivo: Los modelos de gestión ágil se emplean para trabajar en entornos de incertidumbre e inestabilidad de requisitos. Intentar predecir en las fases iniciales cómo será el resultado final, y sobre dicha predicción desarrollar el diseño y la arquitectura del producto, no es realista, porque las circunstancias obligarán a remodelarlo muchas veces, por lo que al igual que XP, Scrum no se sujeta a criterios predictivos sino a dar seguimiento a los cambios y adaptaciones necesarios para el funcionamiento del producto y nivel de satisfacción del cliente.

Scrum considera a la inestabilidad como una premisa, y se adoptan técnicas de trabajo para permitir la evolución sin degradar la calidad del diseño y estructuración que también evoluciona durante el desarrollo del producto (Padilla, 2014).

Durante el desarrollo se genera el diseño y la arquitectura final de forma evolutiva. Scrum no los considera como productos que deban realizarse en la primera “fase” del proyecto. (El desarrollo ágil no es un desarrollo en fases).

Auto-organización: En la ejecución de un proyecto son muchos los factores que inciden de forma integral y que pueden ocurrir y manifestarse en el proceso de desarrollo. La gestión predictiva confía la responsabilidad de su resolución al gestor de proyectos. En Scrum los equipos son auto-organizados (no auto-dirigidos), con margen de decisión suficiente para tomar las decisiones que consideren oportunas por tener autonomía y control sobre su gestión y actuación (Palacio, 2006).

Colaboración: El equipo debe estar consolidado, en comunicación constante, considerando que para que funcione la auto organización como un control eficaz, cada miembro del equipo debe colaborar de forma abierta con los demás, según sus capacidades y no según su rol o su puesto (Palacio, 2006).

Visión general del proceso: Scrum denomina “*sprint*” a cada iteración de desarrollo y según las características del proyecto y las circunstancias del

sprint, puede determinarse una duración desde uno hasta dos meses, aunque no suele ser recomendable hacerlos por más de un mes, mencionado con anterioridad. El *sprint* facilita y es el centro de desarrollo iterativo e incremental (Palacio, 2010).

3.3.3 ¿Cuándo se utiliza la Metodología Scrum?

Con Scrum el cliente se entusiasma y se compromete con el proyecto, dado que lo ve crecer iteración a iteración. Asimismo, le permite en cualquier momento realinear el software con los objetivos de negocio de su empresa, ya que puede introducir cambios funcionales o de prioridad en el inicio de cada nueva iteración. Esta metódica de trabajo, promueve la innovación, motivación y compromiso del equipo que forma parte del proyecto, por lo que los profesionales encuentran un ámbito propicio para desarrollar sus capacidades.

3.3.4 Valores de Scrum

Scrum es un basamento a partir del cual se pueden desarrollar los principios ágiles. Es una ayuda para organizar a las personas y el flujo de trabajo, al igual que la metodología ágil Cristal, por mencionar un ejemplo. Sus valores se enfocan en los siguientes criterios:

- Fortalecimiento e implementación de los valores del desarrollo ágil.
- Delegación de atribuciones al equipo para que pueda auto-organizarse y tomar las decisiones sobre el desarrollo.
- Respeto entre las personas.
- Los miembros del equipo se integran, desarrollan trabajo colaborativo sustentado en el respeto a los conocimientos y capacidades de cada miembro.
- Responsabilidad y auto-disciplina (no disciplina impuesta).
- Trabajo centrado en el desarrollo de lo comprometido: información, transparencia y visibilidad del desarrollo del proyecto (Palacio, 2006).

3.3.5 Principios de Scrum

Un principio clave de Scrum, es el reconocimiento de que durante un proyecto los clientes pueden cambiar de idea sobre lo que quieren y necesitan, y que los desafíos impredecibles no pueden ser fácilmente enfrentados de una forma predictiva y planificada. Por lo tanto, Scrum adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes.

3.3.6 Fases de Scrum

De manera general, el proceso de desarrollo de un producto de software, se compone de cinco fases importantes:

- Planes de lanzamientos
- Distribución, revisión y ajuste de los estándares del producto
- *Sprint*
- *Revisión del Sprint*
- Cierre

Scrum distingue distintos actores con diferentes papeles dentro del proceso. De forma general, podemos distinguir propietario del producto o *Product Owner*, maestro de Scrum o *Scrum Master*, equipo de desarrollo o *Scrum Team* y cliente o usuario.

El *Product Owner*, es la única persona encargada de la dirección y control del *Product Backlog*, es decir, de las historias de usuario que debe cumplir el sistema. Se trata de una persona física (solamente una persona para eliminar las posibles confusiones o interferencias), no una organización o Comité. Bien puede ser el propio cliente in situ en el lugar de desarrollo u otra persona que tenga el conocimiento suficiente sobre el producto o pueda estar en continuo contacto con el cliente para marcar las prioridades del proyecto. Es, por tanto,

la persona oficialmente responsable del proyecto que de forma visible, vocal y objetiva, debe tomar todas las decisiones de negocio para el producto. Para que el *Product Owner* tenga éxito, el resto del equipo de la organización tiene que respetar sus decisiones. En cuanto a su implicación, debe estar en constante interacción con el equipo de desarrollo. Debe asistir, al menos, a las 20 reuniones de planificación y de revisión de cada *sprint* y estar en continuo contacto con el equipo para proporcionar detalles sobre las historias de usuario y la continua retroalimentación que dirija el desarrollo del *sprint*.

El *Scrum Master* es la persona responsable del éxito al aplicar la metodología Scrum en el desarrollo del proyecto o producto, asegurando que los valores, prácticas y reglas son seguidos por el resto del equipo. En concreto, es la persona que asegura el seguimiento de la metodología guiando las reuniones, ayudando al equipo ante cualquier problema que pueda aparecer y controlando que el trabajo siga el ritmo adecuado. Por tanto, debe tomar decisiones inmediatas y eliminar los impedimentos que vayan surgiendo en el momento, aunque en ocasiones, no cuente con toda la información necesaria. Su responsabilidad es entre otras, la de hacer de paraguas ante las presiones externas y motivar al resto del equipo. Por tanto, la labor de *Scrum Master* requiere una fuerte personalidad ya que debe facilitar el trabajo del equipo sin imponer autoridad. Las personas que ejercen este rol deben ser capaces de hacer cualquier cosa por ayudar al equipo de desarrollo, incluso aunque estas acciones estén enfrentadas con sus propios intereses.

El *Scrum Team* lo conforman las personas responsables de implementar la funcionalidad o funcionalidades elegidas por el *Product Owner*. Debe ser un conjunto de personas motivadas con habilidades y capacidades complementarias que estén comprometidos por un propósito común: cumplir el objetivo del *sprint*. El equipo tiene plena autoridad para tomar todas las decisiones que consideren adecuadas en el desarrollo del proyecto, auto-organizándose y auto-disciplinándose. Así, por ejemplo, en las reuniones de planificación el *Product Owner* y el *Scrum Team* deben llegar a un acuerdo realista sobre las historias de usuario que se van a completar en el siguiente *sprint* y si en algún momento el *Scrum Team* considera que algunas de las

prioridades del *Product Owner* no es razonable dispone de libertad absoluta para reseñar esta circunstancia y obligar al propietario del producto a variar sus prioridades.

Teóricamente, se estima que debería estar formado por un número de entre 7 u 8 miembros como máximo y 2 como mínimo. Finalmente, el cliente es o son los beneficiarios finales de un producto, y quienes viendo los progresos, pueden aportar ideas, sugerencias o necesidades. Su participación es importantísima e imprescindible en esta metodología.

La efectividad de la metodología Scrum para la gestión de proyectos, se basa en un conjunto de valores fundamentales que deben seguir todos los integrantes del equipo, principios sobre los que reposan el resto de prácticas: compromiso, esmero, franqueza, respeto y valor. Los miembros del equipo deben estar dispuestos a comprometerse con el objetivo de cada *sprint* y del proyecto en general. Scrum proporciona al equipo toda la autoridad que necesiten, para obtener a cambio, su compromiso. El equipo se tiene que comprometer a hacer su trabajo. Cada miembro debe concentrar todos sus esfuerzos y habilidades en cumplir con el trabajo que se han comprometido a realizar sin desviarse en otros aspectos, realizando todas sus labores con esmero.

Todos los aspectos del proyecto son visibles para todo el equipo. Además, los miembros del equipo están avalados por sus conocimientos y experiencias, por lo que el respeto es un valor fundamental entre ellos. Finalmente, cada miembro del equipo tiene que tener el coraje suficiente para comprometerse, actuar, ser transparente en el desarrollo y respetar y hacer que le respeten (Palacio, 2010).

El proceso de desarrollo con Scrum se compone de cinco actividades principales: revisión de los planes de *release*, distribución, revisión y ajuste de los estándares de producto, *sprint*, revisión del *sprint*, y cierre.

La fase de *sprint* es en la que se realiza el desarrollo de software propiamente dicho. Dentro de un *sprint* se efectúan varias sub-actividades: desarrollo,

empaquetado, revisión y ajuste. No existe secuencia dentro de esta fase. Algunas veces, un ítem del *backlog* debe ser desarrollado, empaquetado y revisado, y otras veces, el ítem sólo debe ser revisado y ajustado; todo depende de las características del ítem en cuestión.

Cada *sprint* es seguido por un proceso de revisión. Durante esta etapa, se revisa el software desarrollado en el *sprint* que acaba de finalizar y, de ser necesario, se agregan nuevos ítems en el *backlog*. El grupo de revisores debe incluir a los *stakeholders*, los administradores del proyecto, los desarrolladores y los usuarios. Las actividades del *sprint* y revisión del *sprint*, tienen que repetirse hasta que el producto esté en condiciones de ser distribuido por los accionistas. Luego, el proyecto entra en la fase de cierre, tras la cual el producto queda en condiciones para el cierre de versión (*release*) y distribución.

En la fase de cierre, se realizan las últimas tareas de depuración (*debugging*), luego de lo cual se construyen los entregables y el proyecto se da por finalizado. Debido a lo imprevisible de los procesos de desarrollo de software, no es posible definir exactamente cuándo ocurrirá la fase de cierre, de modo que los proyectos pueden demorarse más o menos de lo planeado. Pero mediante el uso de los controles que provee Scrum, se pueden hacer estimaciones sobre su duración. (Palacio, 2010)

Sin dudas, Scrum constituye un método ágil de gestión de proyectos cuyo objetivo principal es elevar al máximo la productividad del equipo de desarrollo. Reduce al máximo las actividades no orientadas a producir software funcional y produce resultados en períodos cortos de tiempo. Como método, enfatiza valores y prácticas de gestión, sin pronunciarse sobre requerimientos, prácticas de desarrollo, implementación y demás cuestiones técnicas. Más bien delega completamente al equipo la responsabilidad de decidir la mejor manera de trabajar para ser lo más productivos posibles. No se trata únicamente de un método de software sino que puede ser aplicado teóricamente a cualquier contexto en donde un grupo de personas (equipo de trabajo) necesita trabajar juntos para lograr una meta común.

Como metodología para desarrollo de software, se basa en los principios ágiles pero es necesario complementar el método Scrum con otros métodos para su implementación en el ámbito de desarrollo de software.

- Más que una metodología de desarrollo, Scrum es una herramienta para gestionar proyectos.
- Está basado en el hecho de que el trabajo es realizado por equipos autoorganizados y autodirigidos, logrando motivación, responsabilidad y compromiso.
- Es un proceso constructivo, iterativo e incremental donde las iteraciones tienen una duración fija.
- Contiene definición de roles, prácticas y productos de trabajo escritas de forma simple y soportada en un conjunto de valores y principios (métodos ágiles)

Las actividades que se llevan a cabo en Scrum son las siguientes:

3.3.6.1 Planificación de la iteración

1. Desde el comienzo de la iteración se realiza la reunión de planificación de la iteración. Tiene dos partes:
 - Selección de requisitos (4 horas máximo). El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que se compromete a completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.
 - Planificación de la iteración (4 horas máximo). El equipo elabora la lista de tareas de la iteración necesarias para desarrollar los requisitos a que se ha comprometido. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se autoasignan las tareas (Proyectalis, 2014).

3.3.6.2 Ejecución de la iteración

Cada día el equipo realiza una reunión de sincronización (15 minutos máximos). Cada miembro del equipo inspecciona el trabajo que el resto está

realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con el compromiso adquirido. En la reunión cada miembro del equipo responde a tres preguntas:

¿Qué he hecho desde la última reunión de sincronización?

¿Qué voy a hacer a partir de este momento?

¿Qué impedimentos tengo o voy a tener?

Durante la iteración, el facilitador se encarga de que el equipo pueda cumplir con su compromiso y de que no se merme su productividad:

- Elimina los obstáculos que el equipo no puede resolver por sí mismo.
- Protege al equipo de interrupciones externas que puedan afectar su compromiso o su productividad. (Proyectalis, 2014)

3.3.6.3 Inspección y adaptación

El último día de la iteración, se realiza la reunión de revisión de la iteración. Tiene dos partes:

- Demostración (4 horas máximo). El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, replanificando el proyecto.
- Retrospectiva (4 horas máximo). El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad. El facilitador se encargará de ir eliminando los obstáculos identificados.

3.3.6.4 El proceso

En Scrum, un proyecto se ejecuta de forma temporal, con espacios de tiempo cortos y fijos (iteraciones de un mes natural y hasta de dos semanas, si así se

necesita). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite (Proyectalis, 2014).

3.3.7 Beneficios de Scrum

Scrum brinda muchos beneficios a los proyectos de desarrollo que utilizan esta metodología, entre estos se mencionan los siguientes:

- Debido a la adaptabilidad que maneja Scrum, los cambios que se generan en los proyectos, se convierten en innovaciones, esto agrega mucho valor al producto final, cubriendo las necesidades del cliente y del mercado actual que es de alta competitividad.
- Al brindar resultados de forma anticipada, sin requerir que el proyecto llegue a su etapa final, provee al cliente un producto que ya puede ser usado, y de esta manera, pueda ir recuperando su inversión de forma temprana, lo que a la vez genera mayor motivación en el grupo, ya que esto crea un sentimiento de satisfacción, al entregar resultados de forma temprana en el proyecto.
- Scrum promueve la comunicación continua, lo que ayuda a conocer el desarrollo del proyecto en todo momento, y de esta manera, brindar la mejor solución, en el caso de existir inconvenientes; además de poder compartir información necesaria que ayude al proyecto.
- El grupo de trabajo por tener la característica de ser multidisciplinario, con habilidades y conocimientos diversos, ayuda a tomar las decisiones más oportunas y correctas; por ejemplo, en la planificación, cuando se requiere estimar esfuerzo para las tareas, o cuando se requiere plantear soluciones.
- Scrum propone buenas prácticas a los miembros del grupo del proyecto, como autogestión, innovación, adaptabilidad; lo que ayuda a que el proyecto se alinee con las metas del negocio, y así se cree en el cliente la satisfacción de un trabajo bien hecho.
- La retroalimentación que se maneja en Scrum, genera mejora continua en proyectos que son desarrollados con esta metodología.

- Debido a que las iteraciones en Scrum se manejan de forma fija, el compromiso entre los miembros del grupo se logra de manera más concreta, ya que ellos propondrán solo lo que pueden cumplir para el lapso de tiempo acordado, y al manejar iteraciones que no sobrepasaran un mes, ayuda a que el entorno de cada sprint sea más manejable, para la identificación temprana de problemas y como resultado, la máxima reducción de riesgos en el proyecto.

- La inclusión del cliente, como parte del grupo de trabajo, ayuda a que este sienta mayor compromiso con el proyecto, y de esta manera, se logra la generación del trabajo conjunto, para alcanzar un producto de alta calidad.

De forma general, se puede afirmar que con la metodología Scrum se obtiene el software lo más rápido posible y este cumple con los requerimientos más importantes. Se trabaja en iteraciones cortas, de alto enfoque y total transparencia.

Se acepta que el cambio es una constante universal y se adapta el desarrollo para integrar los cambios que son importantes. De este modo, se incentiva a la creatividad de los desarrolladores haciendo que el equipo sea auto administrado y controlado sus avances y resultados.

Se focaliza en priorizar el trabajo en función del valor que tenga para el negocio. Tiene un conjunto de reglas muy pequeñas y simples, basadas en los principios de inspección continua. El cliente crea y gestiona la lista de requisitos del producto o proyecto, donde quedan reflejadas sus expectativas a nivel de requisitos, valor, coste y entregas.

Al inicio de cada iteración el cliente prioriza la lista de requisitos del producto o proyecto en función del valor que le aportan, su coste de desarrollo y los riesgos del proyecto, cambiando los requisitos previstos para reaccionar a cambios de contexto en el proyecto. Con ello, el equipo se evita caminar mucho tiempo por un camino equivocado que le obligue a realizar un gran

esfuerzo para llegar al objetivo esperado. Se asegura la calidad del producto de manera sistemática y objetiva, a nivel de satisfacción del cliente, requisitos listos para ser utilizados y calidad interna del producto.

3.3.8 Limitaciones de Scrum

Entre algunas limitaciones que puede presentar Scrum en proyectos de desarrollo de software, se citan las siguientes:

- Scrum no puede gestionar aquellos riesgos que son considerados externos al proyecto, como aspectos: políticos, legales, o de mercado; que ya no depende de la empresa, para mantener un control.
- Para algunos desarrolladores, Scrum no brinda la guía suficiente para los proyectos de software, debido a su gran apertura para agregar los criterios que sean necesarios al proyecto, sin ninguna limitación.
 - Requiere delegar responsabilidades al equipo, incluso permite fallar si es necesario.
 - Es una metodología que difiere del resto, y esto causa una cierta resistencia en su aplicación para algunas personas.
 - Es recomendable emplearlo solo en proyectos a corto plazo.
 - Altas comisiones en caso de fallar.
 - Menos énfasis en la arquitectura del software.
 - No existe contrato tradicional o al menos es bastante flexible.
 - El cliente establece sus expectativas indicando el valor que le aporta cada requisito del proyecto y cuándo espera que esté completado además de que puede comprobar si se van cumpliendo sus expectativas; puede tomar decisiones a partir de resultados objetivos por lo que se ahorra esfuerzo y tiempo.

Scrum, más que una metodología de desarrollo software, es una forma de autogestión de los equipos de programadores. Un grupo de programadores deciden cómo hacer sus tareas y cuánto van a tardar en ello. Scrum ayuda a que trabajen todos juntos, en la misma dirección, con un objetivo claro.

Scrum permite además seguir de forma clara el avance de las tareas a realizar, de forma que los "jefes" puedan ver día a día cómo progresa el trabajo. Sin embargo, Scrum no es una metodología de desarrollo, puesto que no indica qué se debe hacer para hacer el código. Debería, por tanto, complementarse con alguna otra metodología de desarrollo. Se lleva bien con las metodologías ágiles y en concreto con la programación extrema RUP. Una particularidad de esta metodología es que, en cada ciclo de iteración, se hace exigente el uso de artefactos, siendo por este motivo, una de las metodologías más importantes para alcanzar un grado de certificación en el desarrollo del software. La metodología RUP es más adaptable para proyectos de largo plazo.

En cambio, el Scrum es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de software y el desarrollo se realiza en forma iterativa e incremental. Cada ciclo de iteración termina con una pieza de software ejecutable que incorpora nueva funcionalidad. Las iteraciones en general tienen una duración entre 2 a 4 semanas (Proyectalis, 2014).

3.4 Diferencias entre Scrum y Xp

El Scrum es usado como modelo para el desarrollo de productos tecnológicos, aunque también se emplea en entornos que trabajan con requisitos inestables y que requieren rapidez y flexibilidad; situaciones frecuentes en el desarrollo de determinados sistemas de software, mientras que el XP se usa de una forma adaptable a cualquier punto de la vida de cualquier proyecto, considerándose capaz de obtener resultados más realistas y efectivos.

Tanto Scrum como XP, toman en cuenta los cambios que puedan ir ocurriendo a medida que se vaya desarrollando el proyecto para de esta manera mejorar u optimizar la arquitectura del mismo a medida que este se va llevando a cabo.

3.5 Esfuerzo de adaptación de la metodología

Es también de interés estudiar el esfuerzo que ha supuesto evolucionar desde el estado anterior de desarrollo, considerado más convencional, a las

metodologías ágiles, incluyendo las variaciones incluidas en cada una de las metodologías utilizadas, y la formación de aquellos miembros que no estaban familiarizados con la metodología ágil.

- ¿Cuánto de grande es el esfuerzo de familiarización con la metodología Scrum o XP?
- ¿Cuál es la curva de aprendizaje del equipo en función de aspectos tales como el cumplimiento de objetivos, su satisfacción personal o hábito en las prácticas ágiles?

3.5.1 Satisfacción del cliente y de los desarrolladores

- Dado que el movimiento ágil y más concretamente Scrum, no establece prácticas ingenieriles, si no que más bien se puede entender como una nueva cultura de desarrollo de software, es de interés para el estudio conocer las percepciones que tanto el cliente como el equipo de desarrollo tienen sobre la metodología:

1. ¿La metodología SCRUM satisface al cliente?
2. ¿La metodología SCRUM satisface al equipo de desarrollo?

De acuerdo a la experiencia adquirida, se analizarán las metodologías ágiles y las prácticas que se proponen desde una perspectiva más ingenieril.

- ¿Cuáles son las ventajas o puntos fuertes de estas metodologías en la evolución de productos software?
- ¿Qué problemas presentan las metodologías?
- ¿Cómo valorar la evolución de un producto si se aplica una metodología ágil?

Es importante señalar, que estudios preliminares realizados en la Universidad Politécnica Nacional (2010), detallan ciertas características de algunas metodologías ágiles, asignando un número a los distintos parámetros en comparación, siendo 5 el número máximo y 1 el mínimo. Se observa que la mayoría de metodologías obtienen un buen resultado en las prácticas ágiles seleccionadas, aspectos también a considerar para el presente estudio:

Tabla 1: Comparación de metodologías ágiles

	ASD	CRYSTA L	FDD	SCRUM	XP
Sistema como algo cambiante	5	4	3	5	5
Colaboración	5	5	4	5	5
Resultados	5	5	4	5	5
Simplicidad	4	4	5	5	5
Adaptabilidad	5	5	3	4	3
Excelencia técnica	3	3	4	3	4
Prácticas de colaboración	5	5	3	5	5
Media Total	4.5	4.4	3.7	4.5	4.5

Fuente: Alonso, J. (2010). Propuesta de aplicación de Scrum para minimizar los riesgos en un Proyecto de Desarrollo de Software. Escuela Politécnica Nacional. Quito, Ecuador.

- **Sistema como algo cambiante:** esta característica se refiere a la flexibilidad y capacidad que tendrá el sistema a ser desarrollado, para incorporar nuevos requerimientos sobre la marcha, así como cambios de requerimientos. Como se puede observar en la Tabla 4, Crystal es la metodología ágil que menos puntaje tiene, y esto es porque a diferencia de las demás metodologías ágiles, Crystal no cubre todo el ciclo de vida del proyecto, por lo que los requerimientos deben ser cambiados o añadidos tan solo al principio del proyecto.
- **Colaboración:** son los valores y filosofía en los que se encuentran interrelacionados las partes involucradas en el desarrollo de software. Debido a que es un principio del Manifiesto Ágil, en todas, tanto XP, Scrum, Crystal y ASD, se incluye al cliente como parte del equipo de desarrollo y además se incentiva al compañerismo y la confianza.
- **Resultados:** los resultados se refieren al éxito del sistema desarrollado después de aplicar la metodología. Como se observa en la Tabla 4, todas las metodologías ágiles principales poseen máximo puntaje, puesto que el cliente al colaborar y revisar sus requerimientos constantemente, en la mayoría de casos, el sistema final satisface sus requerimientos.
- **Simplicidad:** es la facilidad para poner en práctica los diferentes principios, reglas y procedimientos de una metodología. XP al tener reglas muy flexibles y precisas, es fácil de aplicar, además la relación con el cliente es directa y continua, por lo que tiene la calificación más alta. Scrum en

cambio, al tener las opiniones del cliente en reuniones periódicas (cada dos a cuatro semanas), puede tener dificultades de comunicación o entendimiento entre las partes. Crystal y ASD, en cambio, requieren muchos responsables para la toma de decisiones, puesto que son diseñados para equipos grandes.

- **Adaptabilidad:** es la capacidad que tiene una metodología para adaptarse a nuevos cambios, y más no luchar contra ellos. XP, Scrum y Crystal, tratan de evitarlos con la comunicación continua y la intervención directa del cliente. ASD (Adaptable Software Development) en cambio toma como principio fundamental la adaptación a los cambios, por lo que la metodología en sí está basada en una adaptación continua a circunstancias cambiantes a lo largo del ciclo de vida.

El mayor beneficio de una metodología específica para el desarrollo, es sin dudas, la organización de tareas, sistematización del procedimiento y sobretodo un modelo estático y predictivo que administre las tareas y además, informe intuitivamente la situación del proyecto en cualquier período de tiempo en que se considere importante.

3.6 Elementos de la Guía del PMBOK determinantes para su integración a las metodologías ágiles

Project Management Institute (PMI) es una institución que constituye una guía rectora de buenas prácticas la gerencia de proyectos, que establece definiciones, metodologías y principios para gestionar adecuadamente los proyectos. Esta organización desarrolla una guía de fundamentos para la dirección de proyectos en varias ediciones, las cuales se denominan PMBOK, que traducido al español, significa Guía de los Fundamentos para la Dirección de Proyectos, las cuales orientan y sirven de base para el desarrollo de proyectos. En la presente investigación mayormente se toman en cuenta principios de la Guía del PMBOK Quinta Edición.

El PMI, considera la norma como una referencia fundamental en el ámbito de la dirección de proyectos para sus certificaciones y programas de desarrollo profesional. En su carácter de referencia fundamental, esta norma no está completa ni abarca todos los conocimientos.

Se trata de una guía, más que de una metodología (Guía del PMBOK, 4ta edición, 2008). Actualmente el PMI, “representa a más de 600.000 miembros en más de 185 países, a 479.237 PMPs, a 17.924 CAPMs y a 265 capítulos. La membresía en América Latina excedió los 22.000 miembros. Los gobiernos de los países de Latinoamérica están adoptando modelos de gestión aplicando estándares del PMI, oficinas de proyectos y titulares CAPM y PMP”. (Boletín Latino de PMI, 2012).

La necesidad e importancia de proponer una integración de la guía del PMBOK a las metodologías ágiles, es precisamente, orientado a mejorar la calidad de los procesos en la gestión de proyectos si se tiene en cuenta que un proyecto lleva implícito una serie de procedimientos, etapas y acciones, que son tan importantes como el resultado final. De hecho, los productos finales que se obtienen tendrán mayor calidad en la medida que los procesos anteriores se hayan realizado eficientemente. Por ende, para el desarrollo de software también se requiere considerar estos procesos, no solo centrarse como mayormente se hace en las empresas, en el producto final, que puede limitar la calidad del proyecto en sí mismo, a partir de omitir ciertos productos y no lograr una planificación adecuada donde todos los niveles y responsables asignados a cada proyecto, estén alineados para que la planificación funcione adecuadamente.

A partir del análisis del PMI en su libro PMBOK, considera que un proyecto tiene diferentes entregables como productos en sí mismos, capacidad de realizar un servicio o un resultado tal como un producto o documento, donde se requiere de la dirección de proyectos y los procesos teniendo en cuenta que según los principios del PMBOK, la dirección de proyectos implica la aplicación

de conocimientos, habilidades, herramientas, técnicas y actividades que permitan cumplir con los requerimientos de clientes naturales y de empresas, lo que puede lograrse a partir de los 42 procesos de la dirección de proyectos, agrupados en 5 grupos: iniciación, planificación, ejecución, seguimiento y control, cierre (Fiallos, 2016).

Desde estas perspectivas, puede entenderse la necesidad e importancia de entender y aplicar ideas en torno a los procesos que implica dirigir un proyecto, como la necesidad de identificar requisitos, abordar las diversas necesidades, inquietudes y expectativas de los interesados según se planifica y efectúa el proyecto así como lograr un análisis pormenorizado y equilibrar aspectos en cuanto a alcance, calidad, tiempo, costos, recursos y evaluación de riesgos.

La planificación de un proyecto conlleva a muchos cambios antes de la entrega final de los productos o servicios, por lo que la elaboración paso a paso permite a los directores de proyectos mayor información para una dirección eficiente.

A continuación se establece la propuesta de cómo deberían integrarse subprocesos necesarios para el direccionamiento de los proyectos desde el ciclo de vida propuesto por la guía del PMBOK a las metodologías ágiles que se utilizan en el desarrollo de proyectos de software.

Tabla 2: Procesos y subprocesos de acuerdo al ciclo de vida del proyecto

PROCESO DE INICIO	Desarrollar el Acta de Constitución:
	Requisitos de Alto Nivel
	Entregables
	Riesgos de alto impacto
	Interesados claves
	Cronogramas de Trabajo
	Alcance y fuera de alcance
	Presupuesto
PROCESOS DE PLANIFICACIÓN	Planificar la Gestión de los Interesados del Proyecto
	Estimar el esfuerzo o tamaño de las Historias de Usuario

	Priorizar las Historias de Usuario
	Planificar el Sprint
	Estimar costos del proyecto
DESARROLLO DE CRONOGRAMAS	Duración del Sprint
	Velocidad del equipo
	Días/ Hombres disponibles
	Horas del equipo
PLANIFICACIÓN DE LA CALIDAD	Tipo prueba
	Descripción
	Rol que ejecuta
	Aplicación al proyecto
PLANIFICACIÓN DE LAS COMUNICACIONES	Lugar Común
	Sprint Planning Meeting
	Daily Standup Meetings
	Review
	Retrospectiva
	Radiadores o tableros de Información
PLANIFICACIÓN DE LOS RECURSOS HUMANOS	Etapa (Desde el Sprint 0)
	Nombre de los recursos
	Rol
	Responsabilidades
	Habilidades requeridas
	Criterio de liberación (al término del sprint, al término del release y al término del proyecto)
PLANIFICACIÓN DE LOS RIESGOS	Impedimentos
	Análisis cualitativo y un plan de contingencia para riesgos de alto impacto.
PLANIFICACIÓN DE LAS ADQUISICIONES	Tipos de Contrato
	Producto, servicio o recurso, cantidades, características.
	Formas de Pago y multas
	Cronograma de entregables
	Garantías
PROCESOS DE EJECUCIÓN	Dirigir y Gestionar la Ejecución del Proyecto
	Efectuar las Adquisiciones
PROCESOS DE SEGUIMIENTO Y CONTROL	Tareas pendientes
	Tareas en curso
	Tareas terminadas
	Objetivos de Sprint
	Trabajos restantes versus tiempo (puntos de historia)

	Definir indicadores y métricas que faciliten la toma de decisiones
	Control de las Adquisiciones del Proyecto
PROCESOS DE CIERRE	Cierre del Proyecto
	Cierre de las Adquisiciones

Fuente: elaboración propia.

La gerencia de proyectos por tanto, implica conocimientos y preparación en cuanto al marco conceptual, ciclo de vida, involucrados e influencias organizacionales de un proyecto además de los grupos de procesos que deben llevarse a cabo eficientemente.

En cuanto a los grupos de procesos conforman el proyecto: iniciación, planificación, ejecución, control y cierre, en correspondencia también con las áreas de conocimiento que establece la guía del PMBOK: gerencia de la integración, gerencia de alcance, tiempo, costos, calidad, recursos humanos, comunicaciones, riesgos y adquisiciones (Fiallos, 2016). Todo ello facilitará la gestión de proyectos.

De igual manera, para el trabajo con los stakeholders como son las personas naturales o empresas que están involucrados en el proyecto, se hace necesario realizar inventarios de todas las partes y sus intereses, en correspondencia con sus problemas e intereses.

3.7 Estado del Arte

La fundamentación teórica de la presente investigación, parte de los antecedentes relacionados al tema, es decir, teniendo en cuenta el desarrollo bibliográfico investigativo existente tanto en el ámbito nacional como internacional relacionados al tema, para determinar el nivel de información con que se cuenta sobre la temática en cuestión. Hace referencia a estudios relacionados en torno a las pequeñas y medianas empresas de la región central del Ecuador, dedicadas al desarrollo de software.

Existen algunos proyectos, que hacen referencia de algún modo a la necesaria evaluación de los métodos ágiles y su integración con diferentes metodologías

(Méndez, 2009) desde diferentes perspectivas, y no de forma integral, como se propone. Por ende, en la literatura consultada no se ha encontrado investigaciones similares donde se plantee un modelo de integración a partir de la caracterización y valoración de cuáles son los aspectos del PMBOK que resultan viables de integrar a los métodos ágiles.

Se constató que existen investigaciones en torno a actividades de gestión de la guía PMBOK con diferentes métodos ágiles (Miranda, 1997), sin embargo, la investigación es novedosa en tanto tiene el propósito de enfocarse a un modelo de integración desde diferentes variables e indicadores, ajustado a los requerimientos de las pequeñas y medianas empresas, que no han sido consideradas en ninguno de los estudios antecedentes.

En el ámbito internacional, la Universidad Católica Andrés Bello, Facultad de Ciencias Administrativas y de Gestión, Área de Ciencias Administrativas y de Gestión, Venezuela, dónde se realizó una investigación titulada Modelo de Evaluación de Metodologías para el Desarrollo de Software, en la que se diseñó el modelo de evaluación de metodologías para el aseguramiento de la calidad en el desarrollo del proyecto, disminución o mitigación de los riesgos. Esta investigación expone diversas metodologías para el desarrollo de software, convencionales o de desarrollo ágil que sirven como guía para conocer las características más importantes de cada una de ellas (Méndez, 2009).

En el ámbito nacional, en la Universidad de las Fuerzas Armadas ESPE, se llevó a cabo una investigación que hace referencia al uso del PMBOK y metodologías Ágiles por separado. Así por ejemplo el trabajo de titulado Implementación del estándar de proyectos del Project Management Institute en Portales Web (Fiallos, 2016). En este mismo contexto, el trabajo titulado Método Ágil Scrum, aplicado a la implantación de un sistema informático para el proceso de recolección masiva de información con Tecnología Móvil, de la Universidad de las Fuerzas Armadas ESPE (Toapanta, 2012).

Como se observa existen trabajos que aportan conocimiento para que la dirección de proyectos de software cada vez vaya mejorando, bajo este contexto, la presente propuesta s se suma como un nuevo enfoque para

orientar a los directores de proyectos y a los equipos de desarrollo en su labor de alcanzar el éxito de los proyectos.

En el ámbito regional, en la Universidad Católica del Ecuador - Matriz, se realizó una investigación titulada Diseño de una Metodología para el Direccionamiento de Proyectos de Software para la empresa GESTORINC S.A la cual tuvo como propósito fundamental, diseñar una metodología para direccionamiento de proyectos de software que contribuya a proporcionar alternativas viables para solucionar las falencias identificadas a nivel de Gerencia de Proyectos. Con este trabajo se pretendía contribuir con el mejoramiento de la calidad en la gestión de proyectos, dirigido a mitigar todos los riesgos identificados dentro del direccionamiento de proyectos de software que pueden constituirse en una amenaza o una debilidad para la empresa de frente a su cartera de clientes (Álvaro, 2014).

De acuerdo a los referentes encontrados, la novedad de la investigación radica en que no existen iniciativas similares al menos en la provincia de Cotopaxi ni en la región central de Ecuador en el ámbito de las PYMES, ni investigaciones que hayan sido sistematizadas sobre la integración de metodologías ágiles con los procesos del PMBOK, como planeación, mitigación de riesgos, reconocimiento de *stakeholders*, matriz de problemas, entre otros (Alonso, 2013). De esta manera, se puede afirmar, que el propósito fundamental, es proveer a las empresas de las herramientas necesarias para lograr mejorar la calidad de la gestión de proyectos.

Capítulo 4

Metodología

4.1. Métodos aplicados

Contiene la descripción y argumentación de las principales decisiones metodológicas adoptadas según el tema de investigación y las posibilidades del investigador. La claridad en el enfoque y estructura metodológica es condición obligada para asegurar la validez de la investigación (Sampieri, 2011).

El objetivo de cualquier ciencia es adquirir conocimientos y la elección del método adecuado que nos permita conocer la realidad es fundamental. (Sampieri, 2011)

- La investigación aborda la fundamentación teórica de las categorías fundamentales desarrollando investigación bibliográfica documental del proceso investigativo.

Para cumplir con el propósito de plantear la propuesta orientada a mejorar la dirección de proyectos de desarrollo de software, se considera necesario tener como punto de partida, el análisis de la situación actual de la dirección de proyectos de desarrollo de software, el estudio teórico conceptual de la Guía del PMBOK y los métodos ágiles así como el análisis de los criterios de selección de las metodologías a utilizarse en el trabajo propuesto, como premisas para el planteamiento del Modelo.

Como resultado de este trabajo se presentan esquemas para la gestión de proyectos de desarrollo de software que pueden ser usados para los proyectos de desarrollo de software dependiendo de la metodología a utilizar.

- La investigación y su propuesta, responden en gran medida, a los resultados obtenidos en la investigación de campo, a partir de las interpretaciones de las encuestas aplicadas. Diseñar un modelo de

integración de metodologías ágiles para el desarrollo de software en integración a la Metodología PMBOK, a partir de identificar variables para la evaluación de metodologías aplicadas en el desarrollo de software que además permita desarrollar la matriz de evaluación de metodologías, con las variables encontradas, su ponderación y las metodologías utilizadas para el desarrollo de software.

Realizar el análisis y procesamiento estadístico desde la tabulación, codificación de la encuesta aplicada.

- Valorar las diferentes metodologías utilizadas para el desarrollo de software, convencionales o de desarrollo ágil que servirán como guía para conocer las características más importantes de cada una de ellas.

Al final obtener un modelo a seguir para la selección de una metodología que pueda ser adecuada a las características de cualquier proyecto de desarrollo de software, donde puedan concatenarse los parámetros de la Metodología PMBOK con los métodos ágiles.

4.2. Población y muestra

La población seleccionada para la investigación de campo, son las empresas pequeñas y medianas, de desarrollo de software de la zona centro del país. Se trabajó con diez empresas en las cuales se aplicó una encuesta a los directores y gestores de proyectos.

Para el presente proyecto, se ha considerado factible trabajar con el total de población al no ser numerosa, a partir de un segmento de la población de las pequeñas y medianas empresas de desarrollo de software, donde se realizó la investigación de campo en el transcurso de la semana del diez al quince de enero de 2017.

La información aportada permitió direccionar la propuesta y conocer la realidad de la gestión de proyectos de software.

4.3. Materiales y Herramientas

La medición nominal fue aplicada en las categorías seleccionadas por cada uno de los ítems.

Para la presente investigación se seleccionó la encuesta de acuerdo a los objetivos y a los indicadores de las variables de la investigación, aplicando un cuestionario.

Para elaborar el cuestionario, se tuvieron en cuenta los siguientes aspectos:

- Diseño de la muestra
- Aplicantes, análisis y selección, en este caso, fue autoaplicación
- Prueba piloto para definir las áreas de estudio y validar el instrumento
- Diseño del cuestionario definitivo
- Aplicación
- Tabulación
- Análisis e interpretación de resultados

En cuanto a los tipos de preguntas, se diseñaron preguntas cerradas y de selección múltiple.

Como variante de las interrogantes de opción múltiple, se consideraron las preguntas de estimación las cuales presentan diversos grados referentes a las preguntas planteadas.

Para el diseño definitivo del cuestionario, se tuvo en cuenta que las preguntas estuvieran en relación con los objetivos de la investigación, así como la necesaria codificación, procesamiento y la claridad, precisión, sencillez al conformar cada ítem. El vocabulario fue adecuado a las características de la población objeto de estudio (nivel cultural, estrato social, costumbres, valores, entre otros).

Para el procesamiento de los datos (edición, codificación y tabulación) se utilizó el análisis estadístico (estadísticas descriptivas)

En cuanto a la tabulación, se resumieron los datos en tablas estadísticas. Los datos recogidos se procesaron siguiendo los procedimientos que se mencionan a continuación:

- Revisión crítica de la información
- Tablas de frecuencia si se considera que la norma APA permite elegir entre la representación gráfica y las tablas de frecuencia.

4.4 Procedimientos

1. Identificar componentes o factores que influyen en la evaluación de metodologías aplicadas en el desarrollo de software y las áreas de conocimiento del Project Management Body of Knowledge para la gestión de proyectos de software.

2. Determinar los parámetros de integración entre el Project Management Body of Knowledge y los métodos ágiles que permitan la sistematización del ciclo de vida de los proyectos de desarrollo de software y formular el modelo de integración de ambas metodologías.

4.4.1 PMBOK

Los parámetros de integración son los siguientes:

Figura 1: Áreas de conocimientos del PMBOK



Fuente: Guía PMBOK Quinta Edición

A partir de las áreas de conocimiento del PMBOK, que pueden ser utilizadas en la gestión de proyectos de software, se considera pertinente la integración a los métodos ágiles teniendo en cuenta los parámetros de integración como son los siguientes:

1. Tipo de proyecto
2. Tamaño del proyecto
3. Metodología
4. Alcance
5. Tipo de proceso
6. Fases
7. Ciclo de vida del proyecto
8. Nombre de los productos obtenidos
9. Actividades
10. Agrupamiento de actividades

Se consideró un análisis primeramente, entre PMBOK y metodologías ágiles más utilizadas como XP y Scrum, lo que permitió establecer el modelo final de integración con las metodologías ágiles de forma general. Primeramente, se consideró determinar los puntos de integración necesarios en la gestión de proyectos con el uso de las metodologías ágiles para el desarrollo de software.

Tabla 3: Parámetros de integración

<i>ÁREAS</i>	<i>PROCESOS</i>	<i>ACTIVIDADES</i>	<i>REFERENCIAS</i>
Procesos de la gestión de la integración del	Acta de Constitución del proyecto y registro de interesados	Desarrollar un documento que autoriza formalmente la existencia de un proyecto y confiere al director del proyecto la autoridad para asignar los recursos de la organización a las actividades del proyecto.	PMBOK
	Desarrollo del plan para la dirección del proyecto	Definir, preparar y coordinar todos los planes secundarios e incorporarlos en un plan integral para la dirección del proyecto	PMBOK

proyecto	La colaboración con el cliente es más importante que la negociación de contratos.	Implementar los cambios aprobados, con el fin de alcanzar los objetivos del proyecto.	XP
	Historias de usuario	Analizar todas las solicitudes de cambio; aprobar y gestionar los cambios a los entregables, activos de los procesos de la organización, documentos del proyecto y plan para la dirección del proyecto; y comunicar las decisiones correspondientes.	XP SCRUM
	Al principio de cada sprint se establece una lista de requerimientos llamada backlog, que debe completarse cuando este finalice	Dar seguimiento, revisar e informar del avance del proyecto con respecto a los objetivos de desempeño definidos en el plan para la dirección del proyecto.	SCRUM
Gerencia del alcance	Programación en parejas	Crear un plan para la gestión del alcance que documente cómo se va a definir, validar y controlar el alcance del proyecto.	XP
	Tareas		XP
	Recopilar requisitos	Determinar, documentar y gestionar las necesidades y los requisitos de los interesados para cumplir con los objetivos del proyecto.	PMBOK
	Definir el alcance	Desarrollar una descripción detallada del proyecto y del producto.	PMBOK
	Crear la EDT	Subdividir los entregables y el trabajo del proyecto en componentes más pequeños y más fáciles de manejar.	PMBOK
Gerencia del tiempo	Desarrollo incremental a través de entregas del sistema basados en las historias del cliente	Monitorear el estado del proyecto y de la línea base del alcance del producto, y de gestionar cambios a la línea base del alcance	XP
	Definir las actividades	Identificar y documentar las acciones específicas que se deben realizar para generar los entregables del proyecto.	PMBOK
	Secuenciar las actividades	Identificar y documentar las relaciones existentes entre las actividades del proyecto.	PMBOK
	Estimar los recursos de las actividades	Estimar el tipo y las cantidades de materiales, recursos humanos, equipos o suministros requeridos para ejecutar cada una de las actividades.	PMBOK
	Desarrollar el cronograma	Establecer las políticas, los procedimientos y la	PMBOK

		documentación para planificar, desarrollar, gestionar, ejecutar y controlar el cronograma del proyecto.	
	Reuniones Diarias	Es necesario que los desarrolladores se reúnan diariamente y expongan sus problemas, soluciones e ideas de forma conjunta. Las reuniones tienen que ser fluidas y con la participación de todo el equipo	XP
Gerencia de costos	Estimar los costos	Desarrollar una aproximación de los recursos financieros necesarios para completar las actividades del proyecto	PMBOK
	Determinar el presupuesto	Establecer las políticas, los procedimientos y la documentación necesarios para planificar, gestionar, ejecutar el gasto y controlar los costos del proyecto. Sumar los costos estimados de las actividades individuales o de los paquetes de trabajo para establecer una línea base de costo autorizada.	PMBOK
Gerencia de la calidad	Planificar la calidad	Identificar los requisitos y/o estándares de calidad para el proyecto y sus entregables, así como de documentar la manera en que el proyecto probará el cumplimiento con los requisitos de calidad.	PMBOK
	Refactorización del código	Auditar los requisitos de calidad y los resultados de las mediciones de control de calidad, para asegurar que se utilicen las normas de calidad y las definiciones operacionales adecuadas.	XP
Gerencia de recursos humanos	Adquirir el equipo del proyecto	Confirmar la disponibilidad de los recursos humanos y conseguir el equipo necesario para completar las actividades del proyecto.	PMBOK
	Desarrollar el equipo del proyecto	Mejorar las competencias, la interacción entre los miembros del equipo y el ambiente general del equipo para lograr un mejor desempeño del proyecto.	PMBOK

	Dirigir el equipo del proyecto	Realizar el seguimiento del desempeño de los miembros del equipo, proporcionar retroalimentación, resolver problemas y gestionar cambios a fin de optimizar el desempeño del proyecto	PMBOK
Gerencia de comunicaciones	Identificación de los stakeholders	<p>Crear, recopilar, distribuir, almacenar, recuperar y realizar la disposición final de la información del proyecto de acuerdo con el plan de gestión de las comunicaciones.</p> <p>Desarrollar un enfoque y un plan adecuados para las comunicaciones del proyecto sobre la base de las necesidades y requisitos de información de los interesados y de los activos de la organización disponibles.</p> <p>Monitorear y controlar las comunicaciones a lo largo de todo el ciclo de vida del proyecto para asegurar que se satisfagan las necesidades de información de los interesados del proyecto.</p>	PMBOK
	Planificar las comunicaciones		PMBOK
	La participación del cliente se lleva a cabo a través de su compromiso a tiempo completo en el equipo de desarrollo. Los representantes de los clientes participan en el desarrollo y son los responsables de definir las pruebas de aceptación del sistema.		XP
	Los programadores se comunican constantemente gracias a la programación por parejas. La comunicación con el cliente es fluida ya que el cliente forma parte del equipo de desarrollo. El cliente decide qué características tienen prioridad y siempre debe estar disponible para solucionar dudas.		XP
	Los individuos e interacciones son más importantes que los procesos y herramientas.		XP
	Desarrollo iterativo		XP
	Iteraciones (sprints)		SCRUM
	El resultado de cada sprint es un incremento ejecutable que se muestra al cliente		SCRUM
	Scrum, al igual que el resto de metodologías ágiles, pretende no centrar las tareas de desarrollo en un conjunto de requisitos formalmente definidos, sino que aboga por la incorporación del cliente como un miembro más del equipo de desarrollo		SCRUM
	Distribuir la información		

Gerencia de riesgos	Gestionar las expectativas	Determinar los riesgos que pueden afectar al proyecto y documentar sus características.	PMBOK
	Informar el desempeño		
	Si surgen problemas potenciales durante el diseño, X.P sugiere utilizar una pareja de desarrolladores para que investiguen y reduzcan al máximo el riesgo que supone ese problema.	Definir cómo realizar las actividades de gestión de riesgos de un proyecto	XP
	Planificar la gestión de riesgos	Priorizar riesgos para análisis o acción posterior, evaluando y combinando la probabilidad de ocurrencia e impacto de dichos riesgos. Analizar numéricamente el efecto de los riesgos identificados sobre los objetivos generales del proyecto.	PMBOK
Identificar los riesgos			
Realizar el análisis cualitativo del riesgo			
Realizar el análisis cuantitativo del riesgo			
Gerencia de adquisiciones	Planificar las respuestas a los riesgos	Documentar las decisiones de adquisiciones del proyecto, especificar el enfoque e identificar a los proveedores potenciales. Gestionar las relaciones de adquisiciones, monitorear la ejecución de los contratos y efectuar cambios y correcciones según corresponda.	PMBOK
	Planificar las adquisiciones		
	Ejecutar las adquisiciones		
	Dirigir y gestionar la ejecución del proyecto		
	Monitorear y controlar el trabajo del proyecto		
Realizar el control integrado de cambios			

Fuente: elaboración propia.

Desde lo antes expuesto, es evidente que PMBOK ofrece alternativas para mayor eficiencia en la gestión, control, administración y dirección del proyecto si se considera que varios aspectos no son manejados desde las metodologías ágiles, además de que permitiría reducir los riesgos y desventajas que presentan las metodologías ágiles, como XP que es recomendable emplearla solo en proyectos a corto plazo debido a sus altas comisiones en caso de fallar, así como al ser una metodología que responde a un desarrollo ágil, más que a un desarrollo tradicional como RUP o MSF, está diseñada para proyectos pequeños, los cuales no sobrepasan los 3 meses. Por esta razón, XP puede

conducir a problemas organizacionales cuando se aplica a proyectos de gran magnitud.

4.4.2 Análisis Metodología XP

XP se centra en la continua retroalimentación entre el cliente y el equipo de desarrollo, la comunicación directa e indirecta, contrarrestar el nivel de complejidad en los entregables así como mantenimiento de ajustes y control de cambios. Constituye una herramienta para proyectos donde puedan existir varios cambios y modificaciones, así como un nivel considerable de riesgos. Como metodología pragmática, recoge las que considera mejores prácticas para el desarrollo software, cuya aplicación disciplinada pretende disminuir riesgos y pérdida de recursos económicos. Se trata de doce prácticas: el juego de la planificación, entregas diseños, cambios, programación en parejas, promedio de horas trabajadas, relación con el cliente y estándares.

El modo de operación que XP propone para el desarrollo de software es bastante flexible por lo que puede generar ciertas informalidades en el proyecto de desarrollo como la problemática referente a los requisitos, en la cual en el contexto de XP se reduce a la especificación y seguimiento de los historiales de usuario. El planteamiento que propone al respecto dicha metodología es muy sencillo, pero debido a la gran cantidad de historiales de usuario que puede tener un proyecto y a la volatilidad de los requisitos es que su gestión puede llegar a ser complicada. Con la finalidad de cubrir este punto débil de XP, el estándar ISO 9001:2000 plantea la siguiente cláusula: Los registros deberán ser establecidos y mantenidos para proveer evidencia de la conformidad de los requisitos y la efectividad del Sistema de Gestión de Calidad. Los registros deberán ser legibles, fácilmente identificables y recuperables. Se establecerá un procedimiento documentado para la

identificación, almacenamiento, protección, recuperación, tiempo de retención y disposición de registros.

De forma general, para las metodologías ágiles el cliente es tomado como un recurso más del grupo de trabajo, lo que ayuda de manera impresionante en el intercambio de información, esto ayuda a que el cliente no se visualiza a sí mismo como alguien externo sino como alguien que afecta el desarrollo del producto significativamente. Dado que el cliente es quien conoce el negocio, por lo tanto, podrá priorizar los requerimientos que son indispensables para la organización, logrando que mientras el sistema se vaya completando con todos los requerimientos durante todas las iteraciones, pueda ir ya representando beneficios económicos para la organización, en las primeras presentaciones o iteraciones, lo que también puede significar la reducción de gastos para el negocio, a diferencia de las metodologías tradicionales que gastan muchos recursos y presupuesto en las primeras fases de desarrollo, sin brindar algún producto previo que ya le pueda generar ingresos extras a la empresa, sino solamente hasta el final del proyecto.

Las metodologías ágiles brindan flexibilidad, para que estas se adapten a la organización y no viceversa, lo que podrá agregar valor a la empresa con la aplicación de prácticas útiles. Al manejar el desarrollo por medio de iteraciones se intenta reducir los tiempos de desarrollo, lo que de ninguna forma afectará la calidad del producto, ya que no se debe olvidar que en cada iteración se entrega un subproducto que ya es usable y de calidad.

4.5 Diagnóstico

Análisis e Interpretación de la Encuesta aplicada a las pequeñas y medianas empresas de desarrollo de software.

Las empresas que formaron parte de la población están ubicadas en la ciudad de Quito, Latacunga y Ambato. La investigación se llevó a cabo en las fechas comprendidas del 18 de abril al 18 de mayo del 2017.

Las empresas fueron las siguientes:

Tabla 4: Denominación, ubicación de las empresas y ocupación de los encuestados

Denominaciones	Ubicación	Ocupación de los encuestados
ALTERBIOS SOLUCIONES DE SOFTWARE CIA LTDA	QUITO	Gerente de Tecnología
ADS SOFTWARE CIA LTDA	LATACUNGA	Director de Proyectos
ANDES TI	AMBATO	Director de Proyectos
BESIXPLUS CIA LTDA	AMBATO	Gerente Propietario
ASINFO SOFTWARE & DESARROLLO	QUITO	Director de Proyectos
GESPROJECT	LATACUNGA	Director de Proyectos
SEPRYTSA S.A	LATACUNGA	Director de Proyectos
INTECSA CIA LTDA	AMBATO	Director de Proyectos
INGENIUS SOLUCIONES S.A	AMBATO	Director de Proyectos
TROPIDATOS S.A	AMBATO	Director de Proyectos

Fuente: elaboración propia.

Tabla 5: Años desde su creación

Denominaciones	Años de Creada
ALTERBIOS SOLUCIONES DE SOFTWARE CIA LTDA	5 años
ADS SOFTWARE CIA LTA	15 años
ANDES TI	4 años
BESIXPLUS CIA LTDA	11 años
ASINFO SOFTWARE & DESARROLLO	23 años
GESPROJECT	7 años
SEPRYTSA S.A	10 años
INTECSA CIA LTDA	12 años
INGENIUS SOLUCIONES S.A	22 años
TROPIDATOS S.A	30 años

Fuente: elaboración propia.

Las empresas en su mayoría tienen entre 10 a 20 años de creadas, lo que evidencia que poseen una experiencia notable en el desarrollo de proyectos de software y su posicionamiento en el mercado es significativo.

Tabla 6: Experiencia profesional

Experiencia profesional	Años
ALTERBIOS SOLUCIONES DE SOFTWARE CIA LTDA	Más de 5 años
ADS SOFTWARE CIA LTA	Más de 5 años
ANDES TI	Más de 5 años
BESIXPLUS CIA LTDA	Más de 5 años

ASINFO SOFTWARE & DESARROLLO	Más de 5 años
GESPROJECT	Más de 5 años
SEPRY TSA S.A	Más de 5 años
INTECSA CIA LTDA	Más de 5 años
INGENIUS SOLUCIONES S.A	Más de 5 años
TROPIDATOS S.A	Más de 5 años

Fuente: elaboración propia.

El 100 % de los encuestados posee una experiencia de más de cinco años en la Gestión de Proyectos de Desarrollo de Software, precisamente, como Directores de Proyectos y Gerentes.

1. En el desarrollo de proyectos de software, ¿se requiere organizar el proyecto en cuatro componentes básicos: personal, proceso, producto y tecnología?

Tabla 7: Organización del Proyecto

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	3	30
Siempre	2	20
Ocasionalmente	2	20
Rara Vez	1	10
Total	10	100

Fuente: elaboración propia.

El 20 % de los encuestados afirma que siempre tienen en cuenta la organización del proyecto, sin embargo, es una minoría en la muestra encuestada, por lo que se hace necesario orientar a las empresas en cuanto a la importancia del desarrollo de proyectos de software desde una perspectiva metodológica y de planificación.

Los que responden que Siempre se realiza la Organización del Proyecto en cuatro fases, argumentan que la organización del proyecto es fundamental en la ejecución de un desarrollo e implementación de Software. Estas 4 variables permiten realizar desde la evaluación del proyecto hasta la ejecución y puesta en marcha.

De igual manera, afirma el 20 % restante que muy frecuentemente realiza la organización del proyecto en cuatro fases argumentando que por lo general, se asignan los recursos pensando en los componentes básicos descritos en la pregunta (equipo de trabajo, cronograma y metodología, producto o proyecto,

y plataforma o tecnología), pero existen casos especiales en los que no se organiza el proyecto considerando todos los componentes, sino uno o dos de ellos, siendo necesario que siempre se tuviera en cuenta esta planificación previa.

El 30 % que frecuentemente realiza la planificación mencionada, argumenta que mayormente no se realiza siempre por cuestiones de Productividad, recursos y alcance del proyecto, así como el 20 % que afirma que solo ocasionalmente se estructuran estas etapas, argumentan que es porque dependen del personal que está en el desarrollo de proyectos y el 10 % que rara vez realiza la planificación, es mayormente porque su metodología es basada en proyectos y en tecnologías.

2. ¿El proceso de desarrollo de proyectos está constituido por actividades técnicas y de gestión?

Tabla 8: Actividades Técnicas y de Gestión

Respuestas	Frecuencia	Porcentaje
Siempre	5	50
A veces	2	20
Ocasionalmente	2	20
Rara Vez	1	10
Total	10	100

Fuente: elaboración propia.

Es favorable que el 50 % de las empresas encuestadas, afirmen que el desarrollo de proyectos está constituido por actividades técnicas y de gestión, lo que evidencia que existe planificación, organización para llevar a cabo el proyecto, no solo se centran en el producto final.

1. ¿La etapa de gestión implica planificación, seguimiento, control del proyecto y evaluación de resultados?

Tabla 9: Etapa de Gestión

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	3	30
Siempre	2	20

Ocasionalmente	2	20
Rara Vez	1	10
Total	10	100

Fuente: elaboración propia.

Solo el 20 % afirma que las actividades de gestión siempre implican planificación, control del proyecto y evaluación de resultados, mientras que el 30 % restante, afirma que frecuentemente se realizan las actividades de gestión desde estas perspectivas, por lo que las empresas requieren un asesoramiento, acompañamiento que les permita incorporar como metodología de trabajo la gestión de forma integral no solo brindar importancia o mayor relevancia a las actividades técnicas.

2. ¿La etapa técnica implica, ingeniería del software aplicada a la obtención del producto?

Tabla 10: Etapa Técnica

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	5	50
Ocasionalmente	1	10
Total	10	100

Fuente: elaboración propia.

Es favorable que el 50 % de las empresas encuestadas, afirmen que el desarrollo de proyectos está constituido por actividades técnicas y de gestión mencionado con anterioridad y que el 50 % afirme que la etapa técnica implica, ingeniería del software aplicada a la obtención del producto, lo que evidencia que la tendencia a priorizar las actividades técnicas es significativa, requiriendo una orientación y asesoramiento para que puedan integrar actividades de gestión con el mismo nivel de importancia.

3. ¿Prioriza mayormente en su empresa?

Tabla 11: Priorización en la Empresa

Respuestas	Frecuencia	Porcentaje
Software funcionando	7	70
Documentación	3	30
Total	10	100

Fuente: elaboración propia.

El 70 % de las empresas encuestadas, afirman que priorizan el software funcionando solo un 30 % tiene en cuenta la documentación, lo que evidencia que aún las empresas no se encuentran familiarizadas con la gestión de proyectos.

4. ¿Se pueden complementar las actividades de gestión dentro del desarrollo de proyectos de software con los procesos propuestos por la guía del PMBOK?

Tabla 12: Guía del PMBOK

Respuestas	Frecuencia	Porcentaje
Si	5	50
No	3	30
No podría decir	2	20
Total	10	100

Fuente: elaboración propia.

Es favorable que el 50 % de las empresas encuestadas, afirmen que las actividades de gestión si pueden complementarse con los procesos propuestos en la Guía del PMBOK que favorecería la organización, planificación y control del proyecto hasta su ejecución, mientras que el 30 % afirma que no considera y el 20 % restante, afirma que no conoce sobre esta metodología, siendo importante dar a conocer y fomentar entre las empresas de software, lo que evidencia que la tendencia a priorizar las actividades técnicas es significativa. Las empresas que utilizan la Metodología PMBOK, afirman que mayormente es porque se requiere tener un control y entendimiento claro de la gestión de proyectos de software, y PMBOK presenta una base sólida para el mismo, aunque en ciertos casos, sea mucho más específica de lo necesario, además para otras empresas, la Guía del PMBOK establece lineamientos generales para el desarrollo de un proyecto, en la empresa ASINFO, por ejemplo, afirman que siempre procuran seguir cada uno de los lineamientos para la toma de decisiones, seguimiento del proyecto, evaluación y análisis de resultados.

¿En su empresa, se han sustituido las actividades de gestión por las actividades propuestas en la guía del PMBOK?

Tabla 13: Sustitución por actividades de la Guía del PMBOK

Respuestas	Frecuencia	Porcentaje
Si	5	50
No	5	50
Total	10	100

Fuente: elaboración propia.

Es favorable que ya el 50 % de las empresas afirman que si han sustituido sus actividades de gestión por las propuestas en la Guía del PMBOK, sin embargo, el 50 % restante afirma que no se utiliza PMBOK en su empresa, mayormente porque no cuentan con personal especializado en PMBOK, Se requiere desarrollo con metodologías ágiles y el resto afirma que requiere costos en capacitación, costos en administración mucho mayores y la empresa no asume estos gastos, por lo que se hace necesario que los directores de proyectos se capaciten y actualicen en cuanto a metodologías de proyectos.

- 5. Si su respuesta anterior fue afirmativa, ¿Los procesos de la guía del PMBOK se adaptan a proyectos de desarrollo de software, utilizando solo un subgrupo de estos procesos primarios de acuerdo a las necesidades y alcance que vaya a suplir el proyecto?**

Tabla 14: Argumentaciones

Respuestas	Frecuencia	Porcentaje
Nunca/Rara Vez	2	25
Frecuentemente	2	25
Total	5	50

Fuente: elaboración propia.

Del 50 % que afirma que en su empresa se utiliza la Guía del PMBOK, en cuanto a si los procesos de la Guía se adaptan correctamente al desarrollo de proyectos de software desde un subgrupo de procesos primarios en correspondencia con las necesidades y alcance del proyecto, el 25 % afirma que rara vez y el 25 % restante, que frecuentemente, por lo que es evidente que aún no se establece un procedimiento de estandarización de cómo debe integrarse la Guía a las metodologías que se utilizan para el desarrollo de proyectos.

6. ¿El proceso de gestión de proyectos se organiza en cinco grupos de procesos: inicio, planificación, ejecución, seguimiento y control, y cierre del proyecto?

Tabla 15: Ciclo del Proyecto

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	3	30
Frecuentemente	2	20
Siempre	3	30
Ocasionalmente	2	20
Total	10	100

Fuente: elaboración propia.

Solo un 30 % de las personas encuestadas afirman que la gestión de proyectos se realiza desde el Ciclo de Vida de un proyecto establecido en la Guía del PMBOK lo que evidencia que aún es insuficiente la integración de actividades de gestión en correspondencia con las actividades técnicas.

7. ¿El proceso de elaboración del producto, se realiza a partir del diseño y construcción de la estructura de desglose de trabajo del producto (EDT)?

Tabla 16: Estructura de Desglose de Trabajo

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	2	20
Ocasionalmente	4	40
Total	10	100

Fuente: elaboración propia.

El 40 % de las empresas encuestadas representando a la mayoría afirman que solo ocasionalmente se desarrolla la EDT propuesta en la Guía del PMBOK, siendo necesario establecer pautas metodológicas de la integración de esta metodología al desarrollo de proyectos si se considera su pertinencia e importancia para la ejecución óptima de un proyecto.

8. ¿Los procesos de inicio y planificación son ejecutados antes que los procesos propios del producto?

Tabla 17: Procesos de Inicio y Planificación

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	2	20
Ocasionalmente	4	40
Total	10	100

Fuente: elaboración propia.

El 40 % de las empresas encuestadas representando a la mayoría afirman que solo ocasionalmente se desarrollan los procesos de inicio y planificación antes que los procesos propios del producto, procesos propuestos en la Guía del PMBOK, siendo necesario establecer pautas metodológicas de la integración de esta metodología al desarrollo de proyectos si se considera su pertinencia e importancia para la ejecución óptima de un proyecto.

9. ¿Una vez terminada la planeación, se inicia con las actividades propias del producto en paralelo con los procesos de ejecución, seguimiento y control?

Tabla 18: Actividades propias del producto

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	2	20
Ocasionalmente	4	40
Total	10	100

Fuente: elaboración propia.

El 40 % de las empresas encuestadas representando a la mayoría afirman que solo ocasionalmente se inicia con actividades propias del producto en paralelo con los procesos de ejecución, seguimiento y control, propuesta en la Guía del PMBOK, siendo necesario establecer pautas metodológicas de la integración de esta metodología al desarrollo de proyectos si se considera su pertinencia e importancia para la ejecución óptima de un proyecto.

10. ¿Al terminar con el proceso de seguimiento y control, se continúa con los procesos de cierre del proyecto?

Tabla 19: Procesos de Cierre del Proyecto

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	2	20
Ocasionalmente	4	40
Total	10	100

Fuente: elaboración propia.

El 40 % de las empresas encuestadas representando a la mayoría afirman que solo ocasionalmente se continúa con los procesos de cierre del proyecto, propuestos en la Guía del PMBOK, lo que evidencia que la insuficiente utilización de metodologías de gestión de proyectos válidas, incide en la calidad del producto final porque no se planifican adecuadamente, siendo necesario establecer pautas metodológicas de la integración de esta metodología al desarrollo de proyectos si se considera su pertinencia e importancia para la ejecución óptima de un proyecto.

11. ¿Los procesos de planificación dependen de la actividad técnica de captura de requerimientos?

Tabla 20: Captura de requerimientos

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	3	30
Frecuentemente	3	30
Siempre	4	40
Ocasionalmente	0	0
Total	10	100

Fuente: elaboración propia.

Es favorable que en la mayoría de los casos siempre se realizan los procesos de planificación teniendo en cuenta las actividades técnicas.

12. ¿La captura de requerimientos iniciales, se desarrolla en paralelo a la planificación de alcance?

Tabla 21: Captura de Requerimientos Iniciales

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	4	40
Ocasionalmente	2	20
Total	10	100

Fuente: elaboración propia.

Es favorable que el 40 % de empresas encuestadas, afirman que siempre se realiza la captura de requerimientos iniciales en paralelo a la planificación de alcance, aspectos que pudieran considerarse al momento de plantear una integración de metodologías de gestión y de desarrollo de software.

13. ¿ La planificación de alcance aborda:

Tabla 22: Planificación del Alcance

Respuestas	Frecuencia	Porcentaje
Decisiones del número de iteraciones que se requieren para construir un producto software	7	70
La metodología es tipo cascada, por lo que los procesos técnicos se inician una vez terminado el proceso de planificación	3	30
Total	10	100

Fuente: elaboración propia.

Sin dudas, aún es insuficiente la utilización y aplicación de la Guía del PMBOK integrando sus procesos a las metodologías para el desarrollo de software, por lo que no es común en las empresas, utilizar la metodología tipo cascada, donde los procesos técnicos se inician una vez terminada la planificación siendo necesario una propuesta de integración de metodologías, mayormente ágiles a la Guía del PMBOK.

14. Evalúe ¿con qué frecuencia se manifiestan cada uno de los procesos de la guía del PMBOK en el desarrollo de proyectos de software?

Tabla 23: Frecuencias

INICIO	Desarrollar el acta de constitución del proyecto	Ocasionalmente (4) Frecuentemente (4) Siempre (2)	40 % 40 % 20 %	ANÁLISIS Mayormente frecuentemente desarrollan el acta de constitución
	Identificar a los interesados	Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2) Con mucha frecuencia (1) Siempre (2)	20 % 30 % 20 % 10 % 20 %	Los interesados se identifican mayormente de forma ocasional
	Procesos de planificación	Nunca/Rara Vez (2) Ocasionalmente (2) Frecuentemente (3) Con mucha frecuencia (2) Siempre (1)	20 % 20 % 30 % 20 % 10 %	Los procesos de planificación mayormente se realizan frecuentemente
	Desarrollar el plan para la dirección del proyecto	Nunca/Rara Vez (2) Ocasionalmente (2) Frecuentemente (3) Siempre (3)	20 % 20 % 30 % 30 %	El plan para la dirección del proyecto solo se realiza frecuentemente
	Recopilar requisitos	Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2) Siempre (3)	20 % 30 % 20 % 30 %	Recopilar requisitos, se realiza ocasionalmente
	Definir el alcance	Ocasionalmente (2) Frecuentemente (3) Con mucha frecuencia (2)	20 % 30 % 20 % 30 %	Definir el alcance se tiene en cuenta frecuentemente

		Siempre (3)		
	Crear la EDT(Estructura de descomposición del trabajo)	Nunca/Rara Vez (3) Frecuentemente (3) Con mucha frecuencia (2) Siempre (2)	30 % 30 % 20 % 20 %	La creación de la EDT se realiza frecuentemente
	Definir las actividades	Nunca/Rara Vez (2) Frecuentemente (6) Siempre (2)	20 % 60 % 20 %	Definir las actividades se realiza frecuentemente
	Secuencia de actividades	Frecuentemente (4) Con mucha frecuencia (4) Siempre (2)	40 % 40 % 20 %	La secuencia de actividades se realiza mayormente con mucha frecuencia, siendo un aspecto positivo
	Estimar los recursos de las actividades	Frecuentemente (4) Con mucha frecuencia (4) Siempre (2)	40 % 40 % 20 %	Estimar los recursos de las actividades se realiza con mucha frecuencia, de forma general.
	Estimar la duración de las actividades	Nunca/Rara Vez (3) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (3)	30 % 20 % 20 % 30 %	Estimar la duración de las actividades se realiza nunca / rara vez mayormente
	Desarrollar el cronograma	Nunca/Rara Vez (2) Frecuentemente (6) Siempre (2)	20 % 60 % 20 %	Desarrollar el cronograma se realiza frecuentemente
	Estimar costos	Frecuentemente (4) Con mucha frecuencia (4) Siempre (2)	40 % 40 % 20 %	Estimar costos con mucha frecuencia
	Determinar el presupuesto	Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2)	20 % 30 % 20 %	Determinar el presupuesto ocasionalmente

		Con mucha frecuencia (1) Siempre (2)	10 % 20 %	
Planificar la calidad		Nunca/Rara Vez (3) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (3)	30 % 20 % 20 % 30 %	Planificar la calidad se realiza nunca / rara vez mayormente
Desarrollar el plan de recursos humanos		Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Desarrollar el plan de recursos humanos se realiza nunca / rara vez con mayor frecuencia
Planificar comunicaciones		Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Planificar comunicaciones solo se realiza nunca / rara vez con mayor frecuencia
Planificar la gestión de riesgos		Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Planificar la gestión de riesgos se realiza nunca / rara vez mayormente
Identificar riesgos		Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Identificar riesgos se realiza nunca / rara vez mayormente
Realizar análisis cualitativo de riesgos		Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2) Con mucha frecuencia (1) Siempre (2)	20 % 30 % 20 % 10 % 20 %	Realizar análisis cualitativo de riesgos se realiza mayormente de forma ocasional
Realizar análisis cuantitativo de riesgos		Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2)	40 % 20 % 20 %	Realizar análisis cuantitativo de riesgos se realiza mayormente nunca / rara vez

		Con mucha frecuencia (2)	20 %	
	Planificar la respuesta a los riesgos	Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Planificar la respuesta a los riesgos nunca / rara vez con mayor frecuencia
	Planificar las adquisiciones	Frecuentemente (4) Con mucha frecuencia (4) Siempre (2)	40 % 40 % 20 %	Planificar las adquisiciones se realiza con mayor frecuencia
EJECUCIÓN	Dirigir y gestionar la ejecución del proyecto	Nunca/Rara Vez (2) Frecuentemente (6) Siempre (2)	20 % 60 % 20 %	Dirigir y gestionar la ejecución del proyecto se realiza frecuentemente
	Realizar aseguramiento de la calidad	Nunca/Rara Vez (2) Frecuentemente (6) Siempre (2)	20 % 60 % 20 %	Realizar aseguramiento de la calidad
	Adquirir el equipo del proyecto	Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Adquirir el equipo del proyecto nunca / rara vez se realiza mayormente
	Desarrollar el equipo del proyecto	Frecuentemente (4) Con mucha frecuencia (4) Siempre (2)	40 % 40 % 20 %	Desarrollar el equipo del proyecto se realiza frecuentemente
	Dirigir el equipo del proyecto	Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Dirigir el equipo del proyecto se realiza nunca / rara vez
	Gestionar las expectativas de los interesados	Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2) Con mucha frecuencia (1)	20 % 30 % 20 % 10 % 20 %	Gestionar las expectativas de los interesados se realiza ocasionalmente

		Siempre (2)		
	Efectuar adquisiciones	Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Efectuar adquisiciones se realiza nunca / rara vez
PROCESOS DE SEGUIMIENTO Y CONTROL	Dar seguimiento y controlar el trabajo del proyecto	Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2) Con mucha frecuencia (1) Siempre (2)	20 % 30 % 20 % 10 % 20 %	Dar seguimiento y controlar el trabajo del proyecto se realiza ocasionalmente con mayor frecuencia
	Realizar control integrado de cambios	Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Realizar control integrado de cambios mayormente nunca / rara vez
	Verificar el alcance	Nunca/Rara Vez (2) Frecuentemente (6) Siempre (2)	20 % 60 % 20 %	Verificar el alcance se realiza frecuentemente
	Controlar el alcance	Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2) Con mucha frecuencia (2)	40 % 20 % 20 % 20 %	Controlar el alcance se realiza nunca / rara vez
	Controlar el cronograma	Nunca/Rara Vez (2) Frecuentemente (6) Siempre (2)	20 % 60 % 20 %	Controlar el cronograma se realiza frecuentemente
	Controlar costos	Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Controlar costos se realiza nunca / rara vez
	Realizar control de calidad	Nunca/Rara Vez (4) Ocasionalmente (2) Frecuentemente (2)	40 % 20 % 20 %	Realizar control de calidad se realiza nunca / rara vez

		Con mucha frecuencia (2)	20 %	
	Informar el desempeño	Nunca/Rara Vez (2) Ocasionalmente (3) Frecuentemente (2) Con mucha frecuencia (1) Siempre (2)	20 % 30 % 20 % 10 % 20 %	Informar el desempeño se realiza ocasionalmente
	Dar seguimiento y controlar los riesgos	Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Dar seguimiento y controlar los riesgos se realiza nunca / rara vez
	Administrar las adquisiciones	Frecuentemente (4) Con mucha frecuencia (4) Siempre (2)	40 % 40 % 20 %	Administrar las adquisiciones se realiza frecuentemente
GRUPO DE PROCESOS DE CIERRE	Cerrar el proyecto o fase	Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Cerrar el proyecto o fase se realiza nunca / rara vez
	Cerrar las adquisiciones	Nunca/Rara Vez (7) Ocasionalmente (1) Frecuentemente (1) Con mucha frecuencia (1)	70 % 10 % 10 % 10 %	Cerrar las adquisiciones se realiza nunca / rara vez

Fuente: elaboración propia.

15.¿Cuáles modelos utiliza mayormente para el desarrollo de proyectos?

Tabla 24: Priorización en la Empresa

Respuestas	Frecuencia	Porcentaje
Modelo en espiral (plantea el desarrollo iterativo e incremental, hace explícita las actividades de gestión)	3	30
Modelo en cascada (de forma implícita se encuentran las actividades de gestión)	7	70
Total	10	100

Fuente: elaboración propia.

El 70 % de las empresas encuestadas, afirman que utilizan mayormente el modelo en cascada, lo que evidencia que aún las empresas no se encuentran familiarizadas con la gestión de proyectos, afirmación manifestada por los encuestados en las empresas, de acuerdo a cuáles modelos utilizan.

16.Si su respuesta anterior fue el modelo en espiral, evalúe en una escala del 1 al 5, considerando uno como nivel mínimo y 5 como nivel máximo, la importancia de los aspectos de gestión dentro del modelo:

Tabla 25: Evaluación

Respuestas	Nivel de importancia	Porcentaje
Planificación Gestión de Riesgos	4	30
Gestión de alcance	4	
Total	3	30

Fuente: elaboración propia.

El 30 % de las empresas encuestadas, que afirmaron utilizar el modelo en espiral, afirman que el grado de importancia de la planificación de gestión de riesgos es a un nivel 4, mientras que de igual manera, el 30 % afirmó que la gestión de alcance tiene una importancia a un nivel 4, siendo aspectos favorables que afirman que las empresas en menor porcentaje, se encuentran familiarizadas con la gestión de proyectos.

17.¿Las actividades de gestión presentan el problema que no son lo suficientemente detalladas en los procesos de gestión de proyectos de software?

Tabla 26: Actividades de Gestión

Respuestas	Frecuencia	Porcentaje
Frecuentemente	3	30
Nunca/Rara Vez	3	30
Ocasionalmente	2	20
Muy frecuentemente	2	20
Total	10	100

Fuente: elaboración propia.

El 30 % de las empresas encuestadas, afirman que frecuentemente las actividades de gestión presentan el problema que no son lo suficientemente detalladas en los procesos de gestión de proyectos de software, lo que evidencia que aún las empresas no se encuentran familiarizadas con la gestión de proyectos.

18.¿Considera que la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles?

Tabla 27: Gestión Ciclo de Vida

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Nunca/Rara Vez	2	20
Ocasionalmente	4	40
Total	10	100

Fuente: elaboración propia.

El 40 % de las empresas encuestadas, afirman que ocasionalmente la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles, lo que evidencia que aún las empresas no se encuentran familiarizadas

con la gestión de proyectos por lo que no eligen categorías de mayor frecuencia.

19.¿Utiliza metodologías ágiles o tradicionales para el desarrollo de proyectos?

Tabla 28: Utilización Metodologías Ágiles

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	2	20
Frecuentemente	2	20
Siempre	6	60
Total	10	100

Fuente: elaboración propia.

Un aspecto favorable es que las empresas en un 60 % afirman que siempre utilizan metodologías ágiles lo que es importante para la integración a metodologías como PMBOK.

20.¿Cuáles son las metodologías ágiles que utiliza con mayor frecuencia?

Tabla 29: Metodologías más utilizadas

XP	Nunca/Rara Vez (6)	60 %
	Frecuentemente (2)	20 %
	Con mucha frecuencia (2)	20 %
SCRUM	Frecuentemente (2)	20 %
	Con mucha frecuencia (3)	30 %
	Siempre (5)	50 %
CRYSTAL	Nunca/Rara Vez (7)	70 %
CLEAR	Ocasionalmente (2)	20 %
	Frecuentemente (1)	10 %

Fuente: elaboración propia.

La metodología ágil más utilizada es Scrum por lo que este aspecto resulta importante y relevante para la conformación de la propuesta.

21.¿Con qué frecuencia se manifiestan las siguientes actividades de gestión en correspondencia con la metodología ágil que se utilice, en el desarrollo de proyectos de software?

Tabla 30: Frecuencia de Actividades

XP	Ciclos de entrega, iteración y desarrollo	Nunca/Rara Vez (6) Ocasionalmente (2) Frecuentemente (2)	60 % 20 % 20 %
	Trabajos de máximo 40 horas semanales, el juego de la planificación y la rotación del personal en la programación por parejas	Nunca/Rara Vez (6) Ocasionalmente (2) Con mucha frecuencia (2)	60 % 20 % 20 %
	Juego de planificación: desarrollo y seguimiento de planes de entregas, iteraciones y pruebas	Nunca/Rara Vez (6) Ocasionalmente (2) Frecuentemente (2)	60 % 20 % 20 %
SCRUM	Etapas de Juego	Nunca/Rara Vez (6) Ocasionalmente (2) Frecuentemente (2)	60 % 20 % 20 %
Crystal Clear	Un proyecto se divide en entregas, las entregas en iteraciones por lo que en cada uno de los ciclos (proyecto, entrega, iteración) se requieren actividades de gestión	Nunca/Rara Vez (6) Ocasionalmente (2) Frecuentemente (2)	60 % 20 % 20 %

Fuente: elaboración propia.

En cuanto a las actividades de gestión y su frecuencia de realización en las empresas, se pudo constatar que el ciclo de entrega, iteración y desarrollo, mayormente se utiliza nunca / rara vez según el criterio del 60 % de los encuestados, mientras que el Juego de planificación: desarrollo y seguimiento de planes de entregas, iteraciones y pruebas, se realiza de igual manera, con una frecuencia de nunca / rara vez, de acuerdo al 60 % de encuestados, así como la concepción de que un proyecto se divide en entregas, iteraciones, no se utiliza en las empresas, mayormente con una frecuencia de nunca / rara vez, en el 60 % de las empresas. Ello corrobora que las empresas aún no incorporan aspectos importantes de la gestión de proyectos.

22.¿Los procesos que plantean las metodologías ágiles, son totalmente efectivos en cuanto a actividades de gestión y técnicas?

Tabla 31: Procesos

Respuestas	Frecuencia	Porcentaje
Muy Frecuentemente	3	30
Frecuentemente	3	30
Siempre	2	20
Ocasionalmente	2	20
Total	10	100

Fuente: elaboración propia.

Es importante que el 30 % de las empresas encuestadas frecuentemente reconocen que los procesos que plantean las metodologías ágiles, son totalmente efectivos en cuanto a actividades de gestión y técnicas, por lo que la integración a metodologías como PMBOK resulta factible.

23. En un proyecto bajo XP, ¿se debe construir un plan que determina el número de entregas parciales que se le harán al cliente y cada una de dichas entregas, se organizan en iteración de desarrollo, que a su vez tendrán actividades de gestión propias de cada iteración?

Tabla 32: Proyecto XP. Consideraciones

Respuestas	Frecuencia	Porcentaje
Nunca/Rara Vez	4	40
Frecuentemente	2	20
Muy Frecuentemente	2	20
Ocasionalmente	2	20
Total	10	100

Fuente: elaboración propia.

Mayormente Nunca/ Rara Vez se considera cuando se utiliza XP las consideraciones antes mencionadas, por lo que es evidente la necesidad de integrar a metodologías de gestión de proyectos.

24. ¿Cuáles de los siguientes requisitos, deben tenerse en cuenta para la selección de la metodología idónea para el desarrollo de proyectos de software, de acuerdo a su experiencia?

- 1 = Totalmente en desacuerdo
- 2 = Desacuerdo
- 3 = indeciso/neutral
- 4 = De acuerdo
- 5 = Totalmente de acuerdo

Tabla 33: Requisitos

Indicadores	1	2	3	4	5
La metodología debe ajustarse a los objetivos			1	4	5
La metodología debe cubrir el ciclo entero de desarrollo de software			2	3	4
La metodología debe integrar las distintas fases del ciclo de desarrollo			4	3	3
La metodología debe incluir la realización de validaciones			4	3	3

La metodología debe soportar la determinación de la exactitud del sistema a través del ciclo de desarrollo			3	3	4
La metodología debe ser la base de una comunicación efectiva			2	4	4
La metodología debe funcionar en un entorno dinámico orientado al Usuario	1		3	3	3
La metodología debe especificar claramente los responsables de Resultados			4	3	3
La metodología debería de funcionar en un entorno dinámico orientado al usuario			3	3	3
La metodología debe poder emplearse en un entorno amplio de proyectos software			1	4	5
La metodología se debe de poder enseñar			3	3	3
Debe estar soportada en herramientas CASE			1	4	5
Debe soportar la eventual evolución del sistema					
Debe contener actividades conducentes a mejorar el proceso de desarrollo de software			1	4	5

Fuente: elaboración propia.

En cuanto a requisitos se toma en cuenta que la metodología debe ajustarse a los objetivos mayormente, así como debe poder emplearse en un entorno amplio de proyectos de software.

Capítulo 5

Resultados

PROPUESTA: Modelo de Integración del Project Management Body of Knowledge con las Metodologías Ágiles de Desarrollo de Software

5.1. Producto final del proyecto de titulación

GUÍA METODOLOGICA PARA LA GESTIÓN DE PROYECTOS DE DESARROLLO DE SOFTWARE

La Guía del PMBOK facilita el conocimiento y acercamiento a la gestión de proyectos y favorece como metodología, la dirección de proyectos por lo que para la presente investigación, se considera necesario entender el PMBOK como metodología, desde el proceso de gestión de proyectos en cinco grupos de procesos: Inicio, Planificación, Ejecución, Seguimiento y Control, Cierre del Proyecto.

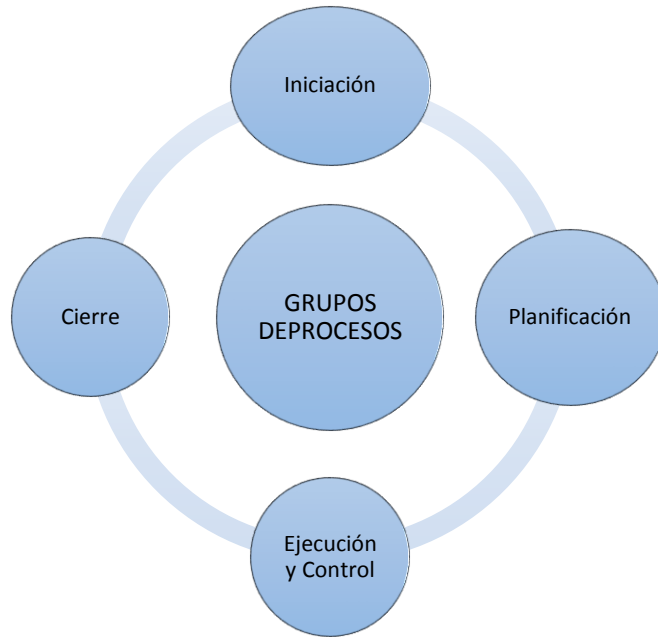
Desde los subprocesos de cada etapa de la gestión de proyectos es que se considerará su integración a las metodologías ágiles. Para ello, se toma en cuenta que como se ha mencionado con anterioridad, los proyectos de desarrollo de software mayormente se sustentan en actividades técnicas y no de gestión del proyecto, lo que incide negativamente en el poder cumplir con indicadores de alcance, tiempo y costos (Contreras, 2011).

Para el modelo de integración se tendrán en cuenta los siguientes principios sustentados en la Guía del PMBOK sobre la Gestión de Proyectos:

INTRODUCCIÓN A LA GERENCIA DE PROYECTOS

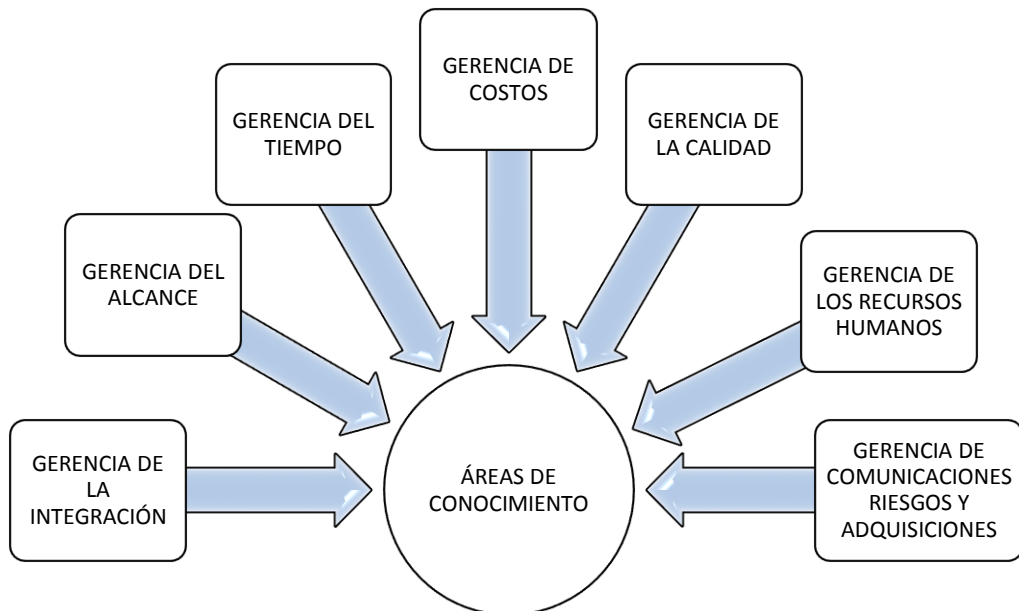
- Marco conceptual
- Ciclo de vida del proyecto
- Involucrados
- Influencias organizacionales

Figura 2: Grupos de Procesos para la Integración



Fuente: Elaboración propia

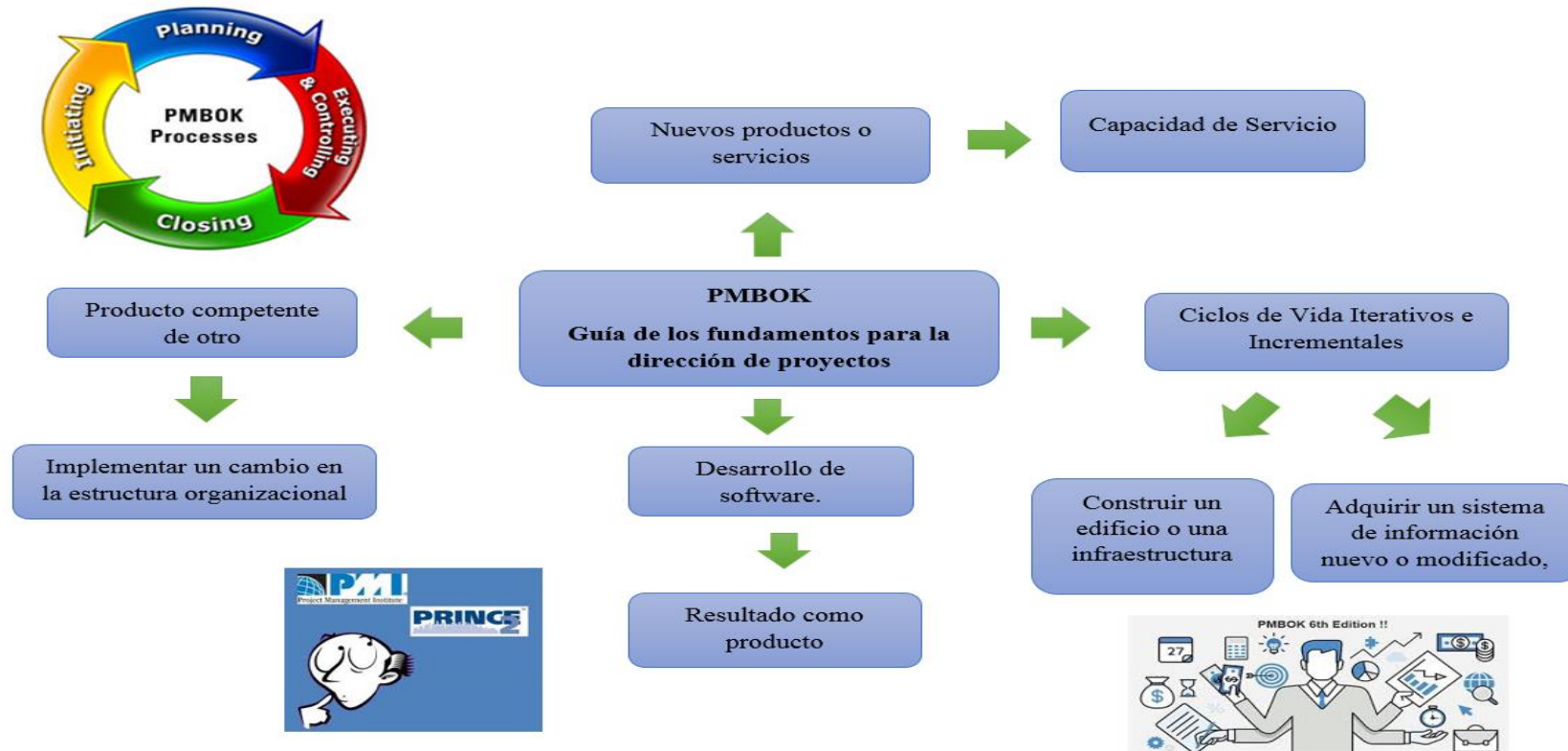
Figura 3: Áreas de Conocimiento para la integración



Fuente: Elaboración propia

De acuerdo al PMI en su libro PMBOK, que traducido estas siglas al español significa: guía de los fundamentos para la dirección de proyectos, se considera que un proyecto implica los siguientes aspectos:

Figura 4: Aspectos fundamentales



Fuente: Elaboración propia

De igual manera, se considerarán los Ciclos de Vida Iterativos e Incrementales y los Ciclos de Vida Adaptativos:

Ciclos de Vida Iterativos e Incrementales

Según PMBOK (2013), llevan implícito las fases o iteraciones del proyecto, las cuales se repiten de manera intencionada en una o más actividades del proyecto, (...) en estas iteraciones se desarrollan entregables del producto a través de una serie de ciclos repetidos, mientras que los incrementos van añadiendo gradualmente funcionalidad al producto (pág. 44)".

Ciclos de Vida Adaptativos

El PMBOK (2013), indica que los ciclos de vida adaptativos (también conocidos como métodos orientados al cambio o métodos ágiles) pretenden responder a niveles altos de cambio y a la participación continua de los interesados. Los métodos adaptativos también son iterativos e incrementales, pero difieren de los anteriores en que las iteraciones son muy rápidas y de duración y costo fijos, los representantes del patrocinador y del cliente deben estar continuamente involucrados en el proyecto para proporcionar retroalimentación sobre los entregables a medida que son generados y para garantizar que el trabajo pendiente asociado al producto, refleje sus necesidades actuales.

Lo antes expuesto evidencia que un principio importante para la integración de PMBOK a la gestión de proyectos de desarrollo de software, es el hecho de considerar que los equipos de proyectos primeramente deben manejar la interdependencia entre el equipo con los clientes, las partes interesadas y los involucrados así como las actividades de gestión.

PRINCIPIOS A SEGUIR PARA LOGRAR LA INTEGRACIÓN DEL PMBOK A LAS METODOLOGÍAS ÁGILES. MANUAL DE USO Y APLICACIÓN

Los principios de integración se sustentan en la propuesta metodológica de la Guía del PMBOK Quinta Edición, los cuales son los siguientes:

1. Dirección de Proyectos: lleva implícito aplicación de conocimientos, habilidades, herramientas y técnicas en correspondencia con las actividades del proyecto para cumplir con los requisitos establecidos.

Los procesos de iniciación requieren de identificar las necesidades de los clientes, lo que persigue con el nuevo producto, se proyectan objetivos, estimaciones en recursos y otros.

Dentro de los formatos se tiene la matriz de Identificación de Involucrados, documento que indica para cada uno de los involucrados, el nivel de impacto, la descripción del impacto, la expectativa y la acción a seguir por parte del equipo de proyecto para cumplir con esa expectativa. Los involucrados se califican por Poder e interés y se clasifican en alguno de los cuatro cuadrantes que permiten identificar el tipo de estrategia a seguir con cada uno.

2. Gestión de integración: que incluye comunicación, acciones integradoras en función de manejar con éxito las expectativas de los clientes y el cumplimiento de requisitos.

3. Desarrollar el acta de constitución del proyecto. Desde este momento el director de proyecto tiene autoridad para la asignación de recursos por actividades planificadas.

4. Desarrollar el plan para la dirección del proyecto. Se define desde este plan, la ejecución, monitoreo, control y cierre.

5. Dirigir y gestionar la ejecución del proyecto. Se pueden ejecutar las acciones y el trabajo diseñado en el plan del proyecto.

6. Herramientas y técnicas. Se diseña el sistema de información que permite el direccionamiento del proyecto desde diseño de cronogramas de trabajo, sistemas de información y comunicación, reuniones que conllevan a los entregables.

7. Monitorear y controlar el trabajo del proyecto. Ello permitirá realizar análisis de las situaciones de riesgo y emitir informes de desempeño de trabajo para la toma de decisiones.

8. Realizar el control integrado de cambios. Se deben analizar las propuestas de cambio, la factibilidad de aceptar o rechazar así como tener en cuenta que

de aceptar los cambios, deben estar presentes en los entregables. Es evidente que ante propuestas de cambios, se deben ajustar costos, actividades, análisis de alternativas lo que exige control y manejo adecuado por parte del equipo del proyecto.

9. Cerrar el proyecto o fase. Es función del director de proyectos realizar una verificación de la documentación que se generó, generando los documentos de cierre del proyecto.

10. Planificar la gestión del alcance, lo que permite la creación del EDT, la estructura de desglose de trabajo.

11. Recopilar requisitos

12. Crear la EDT, descomposición del alcance total del trabajo, donde se establece la línea base del alcance

13. Control del alcance

14. Gestión del cronograma de tiempos del proyecto y planificación.

15. Definir las actividades que pueden secuenciarse y sistematizarse así como la duración por actividades.

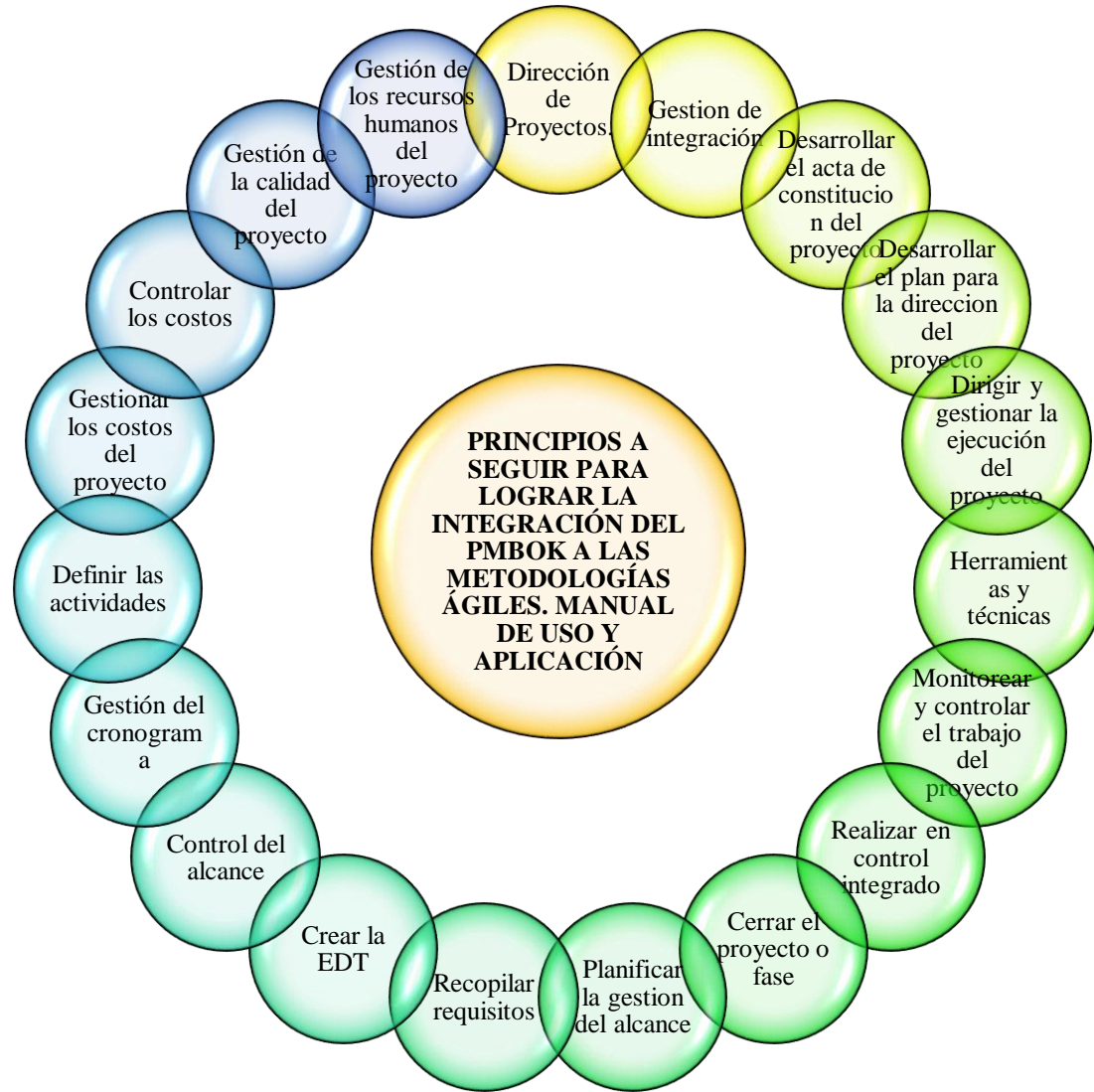
16. Gestionar los costos del proyecto a partir del establecimiento de políticas, procedimientos y la documentación necesaria.

17. Controlar los costos

18. Gestión de la calidad del proyecto.

19. Gestión de los recursos humanos del proyecto.

Figura 5: Principios a seguir para lograr la integración del PMBOK a las metodologías ágiles. Manual de uso y aplicación



El Alcance se relaciona directamente con el Tiempo y el Costo. La Calidad se afectará al cambiar el Alcance, el Tiempo de entrega o el Costo. Si el Tiempo o el Costo se reducen, la Calidad puede verse afectada.

El triángulo está cimentado por dos áreas de alta repercusión para el éxito del proyecto: Recursos Humanos y Comunicación ya que muchos proyectos fallan por deficiencias en la selección e integración de equipos de trabajo y/o por comunicación deficiente.

Una de las estrategias para transferir ciertos riesgos es el manejo efectivo de los Abastecimientos, evaluando las estrategias de contratación, el número de contratos, el tipo de contrato, la forma de pago, la preselección de empresas, la administración y supervisión de los acuerdos contractuales.

Para el desarrollo de proyectos de software se consideran los fundamentos propuestos por la Guía del PMBOK pero también se adecuan a la naturaleza de los proyectos de desarrollo de software, por lo que se considera pertinente los criterios de Presuman (2015) para la integración de la Guía a la ingeniería del software, donde se enfoca la gestión eficaz de proyectos en las cuatro P: personal, producto, proceso y proyecto.

Las características que siempre predominan en el proceso de desarrollo ágil para la mayoría de los proyectos son las siguientes:

FIGURA6: Características que predominan en el proceso de desarrollo ágil



Fuente: Elaboración propia

Para la integración de los parámetros de PMBOK con las metodologías ágiles, es importante situarse en el ciclo de vida de las metodologías, como por ejemplo, XP, mayormente deben integrarse los fundamentos en las etapas de refabricación donde se utiliza de forma repetitiva códigos, para lo cual se establecen patrones; programación en pares y pruebas, donde se tienen las pruebas de unidad y pruebas de aceptación, estas últimas con la participación del cliente (Méndez, 2008).

No obstante, se proponen criterios sobre los cuales se debe construir o elegir una metodología:

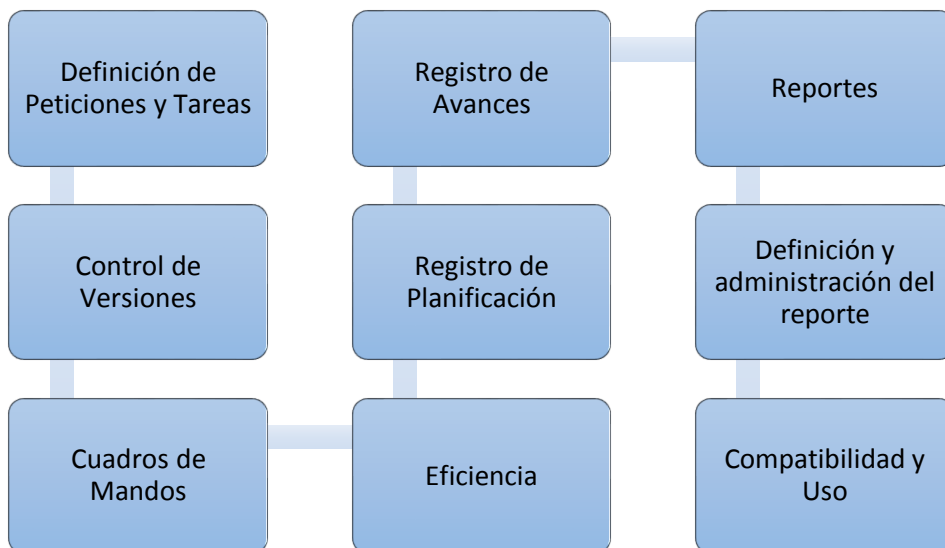
- Deben ajustarse a los objetivos que permita su implementación en la metodología de desarrollo.
- Las metodologías deben cubrir el ciclo entero de desarrollo de software donde debe considerar las etapas: investigación, análisis de requisitos y diseño.
- Debe incluir validaciones, que permitan minimizar riesgos, identificarlos y corregir errores.
- Debe ser base de comunicación efectiva, entre analistas programadores, usuarios y gestores.
- Debe funcionar de forma dinámica y siempre orientada al usuario, donde todos los involucrados intercambien información.
- Deben ser capaces de abordar sistemas de distintos tamaños y rangos de vida (Barzallana, 2008)

Figura 7: Gestión de Proyectos

Gestión de Proyectos.-
Es un proceso que lleva implícito la planificación y la organización del equipo de trabajo, en base al cumplimiento de tareas así como la gestión de información y comunicación al interior del equipo y con el responsable del proyecto; control de cambios y entregables con la finalidad de evitar retrasos en el tiempo de entrega (Moyota, 2014).

Fuente: elaboración propia

Figura 8: Parámetros que se consideran para la integración del MODELO.



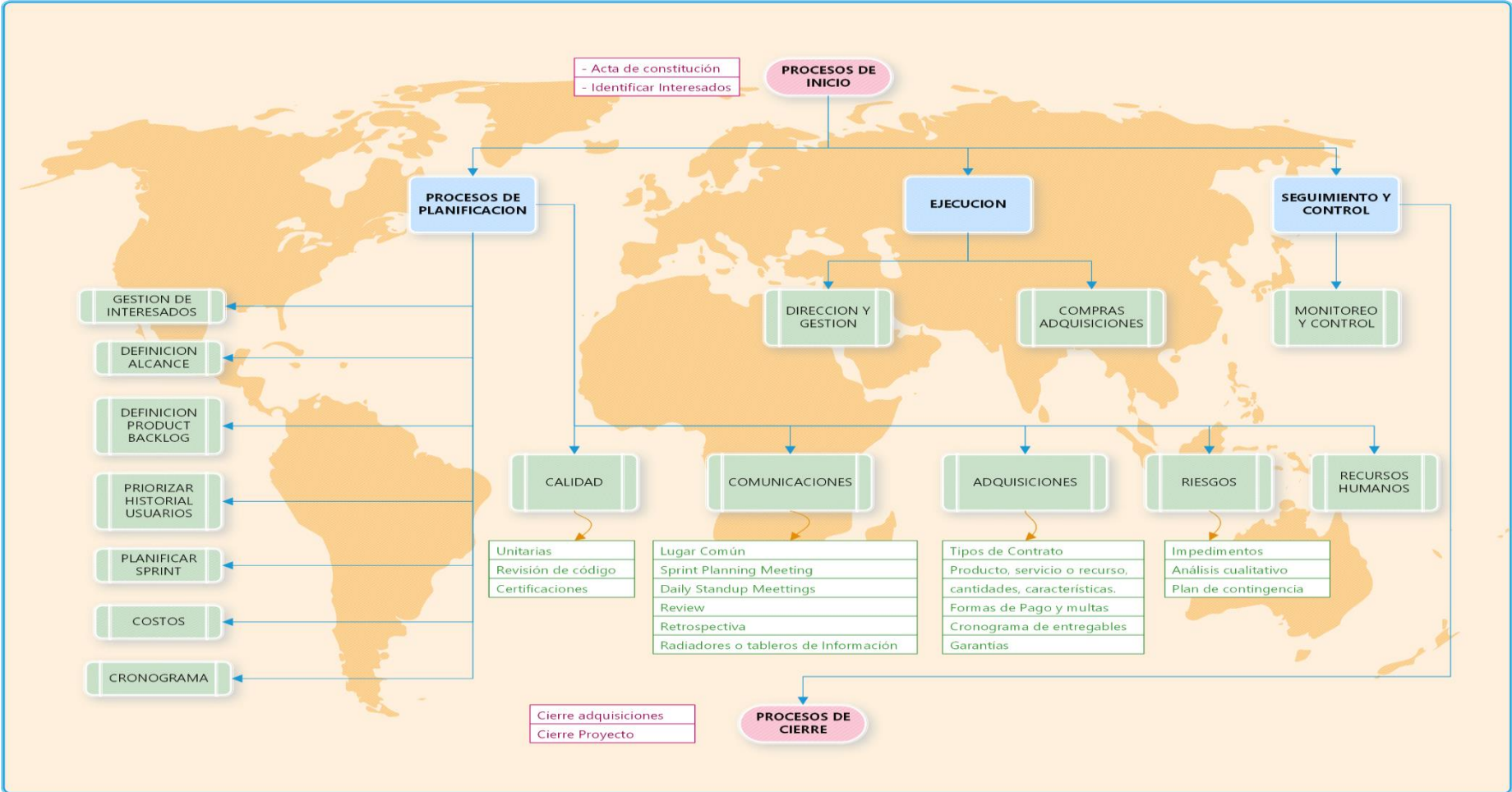
Fuente: elaboración propia

Figura 9: Parámetros de metodologías ágiles para la integración del MODELO.



Fuente: elaboración propia

Figura 10: Procesos para el manejo de un proyecto de software



Fuente: elaboración propia

5.2. Evaluación preliminar

Validación del modelo de integración de la Guía del PMBOK Quinta Edición con las metodologías ágiles, en la realización de un proyecto de software en la Empresa ADS SOFTWARE CIA LTDA ubicada en el cantón Latacunga.

Para la validación del modelo, se procedió a la revisión de la propuesta de un proyecto de desarrollo de software, que la empresa se encontraba iniciando. Se planteó la necesidad al equipo que se designó para trabajar el proyecto y al Director de proyectos de la empresa, de realizarlo desde los principios del PMBOK integrados a la metodología Scrum, bajo la cual se llevaría a cabo el proyecto. Para ello, se realizaron dos reuniones de trabajo con el equipo, con la finalidad de demostrar la importancia de llevar a cabo estos procesos desde una gestión de proyectos planificada, organizada y sistematizada, procediendo a realizar cada uno de los documentos y actividades propuestas por la Guía del PMBOK como parte del desarrollo del proyecto.

Se pudieron identificar las fases y procedimientos que comúnmente utiliza la empresa para la realización de proyectos:

Tabla 34: Procesos actuales en la empresa para el desarrollo de software

FASES	METODOLOGÍAS
Arranque del Proyecto	SCRUM
Levantamiento de Requerimientos	
Capacitación	
Entrega y Pruebas	
Apoyo Inicial	
Acompañamiento y Cierre de Proyecto	

Fuente: elaboración propia.

El Director de Proyectos afirma que existen fallas en los proyectos mayormente por falta de experiencia en la administración de proyectos, falta

de gestión y de organización así como sistematización de procesos y documentación. Se reconoce que las actividades de planeación, seguimiento y control mayormente, deben realizarse en cada proyecto para evitar este tipo de riesgos en la ejecución.

De igual manera, la falta de comunicación con el cliente, el direccionamiento del equipo técnico al desarrollo del producto y que en muchos casos, no tienen en cuenta la gestión de actividades de comunicación, gestión de tiempo y recursos.

Se utilizó la metodología Scrum para la conformación del proyecto, considerando que esta metodología es un marco de trabajo iterativo e incremental para el desarrollo de proyectos y se estructura en ciclos de trabajo llamados sprints. Estos son iteraciones de 1 a 4 semanas, y se suceden una detrás de otra. Al comienzo de cada Sprint, el equipo multi-funcional seleccionó los elementos (requisitos del cliente) de una lista priorizada, pero también llevó a cabo el Acta de Constitución del proyecto, además del análisis de stakeholders. Se comprometieron a terminar los elementos al final del sprint. Durante el Sprint no se pudieron cambiar los elementos elegidos. Al final del Sprint, el equipo lo revisó con los interesados en el proyecto, y les mostraron lo que habían construido. No obstante ya se había realizado un Plan de Comunicación, un Plan de Recursos y costos y el cronograma de actividades, que permitiera contrarrestar riesgos.

Se identificaron los hitos históricos más relevantes de la dirección de proyectos, se realizó un análisis crítico de las metodologías que viene adoptando la empresa, de sus prácticas en la gestión de proyectos así como el levantamiento de información que ya se había realizado al aplicar las

encuestas. Todo ello permitió plantear la guía de fundamentos para la dirección de proyectos, orientada a mejorar la gestión de desarrollo de software, en cuanto a tiempos, costos, calidad, riesgos, recursos humanos y alcance, para el logro de una mayor satisfacción en los clientes con el producto final.

De las reuniones de trabajo, se pudo constatar que muchos de los procesos necesarios para la gestión eficiente del desarrollo de software, la empresa aún no los define, no existen estándares siendo necesario la creación de una herramienta que permita controlar el desarrollo de proyectos desde que se ingresan los requerimientos de los usuarios hasta la generación de productos finales.

Figura 11: Problemas identificados



Fuente: elaboración propia

Desde estas perspectivas, el análisis realizado en la empresa, desde la evaluación realizada con la aplicación de las encuestas que permitió establecer una línea base, tuvo como propósito fundamental, determinar los procesos necesarios para la dirección de proyectos de desarrollo de software en base a la guía del PMBOK y las metodologías ágiles, para que los procesos del proyecto tengan un orden lógico y flexible.

Con estos antecedentes, se planteó la necesidad de incluir procedimientos desde las propuestas del PMBOK, procesos desde la integración a las metodologías ágiles, teniendo en cuenta lo siguiente:

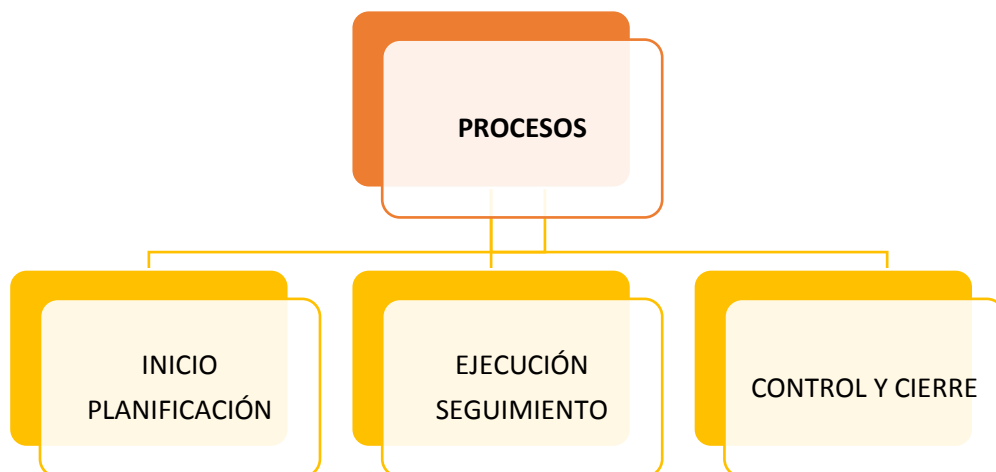
1. Acta de constitución del proyecto con un contrato firmado por parte del cliente y por parte de la empresa
2. Desarrollo del plan para la dirección del proyecto: líneas bases fundamentalmente.
3. Direccionamiento y gestión del trabajo del proyecto hasta su finalización.
4. Monitoreo y control del trabajo, mediante el seguimiento, revisión de avances.
5. Gestión de comunicación continua y fluida entre los sponsor, clientes y grupo de trabajo
3. Identificación de los stakeholders y seguimiento para el cumplimiento de sus intereses y expectativas
4. Gestión de cambio coordinado por el sponsor del proyecto, control integrado de cambios desde el análisis de las solicitudes de cambio, gestión de cambios a los entregables.
5. Gestión de riesgos desde el diseño de plan de control de riesgos
6. Determinación del flujo a seguir, en correspondencia con el direccionamiento de proyectos que puedan tener características y resultados óptimos.

7. Cierre del proyecto mediante el análisis de finalización de actividades en cada uno de los grupos de procesos de la dirección de proyectos.

Los 47 procesos de la nueva Guía del PMBOK versión 5ta edición (2012) se clasifican en cinco grupos de procesos: Inicio, Planificación, Ejecución, Seguimiento y Control, y Cierre; en 10 áreas de conocimiento: Integración, Alcance, Tiempo, Costos, Calidad, Recursos Humanos, Comunicaciones, Riesgos,

Compras o Adquisiciones, como se muestra en la Figura siguiente:

Figura12: Grupo de procesos de la Guía del PMBOK versión 5ta edición (2012)



Fuente: adaptado de Guía del PMBOK versión 5ta edición (2012)

En base a ello, se propuso el modelo de integración.

Tabla 35: Entregables solicitados para la evaluación

Entradas	Entregables
Contrato final	Acta de constitución
Cronogramas del proyecto	Presentación inicial
Matriz de gestión de riesgos	Planificación
Estructura de descomposición del Trabajo (EDT)	Planificación

Matriz de gestión de calidad	Planificación
Requerimientos	Planificación
Casos de prueba aceptados	Ejecución
Informe de Cambios	Ejecución
Informes semanales	Planificación
Adquisición del equipo de trabajo	Planificación e Iniciación

Fuente: elaboración propia.

El éxito o fracaso de la ejecución de un proyecto está en la conjunción de varios factores como la definición del alcance, objetivos, expectativas de los interesados, requisitos del cliente, manejo de presupuesto y tiempo de ejecución. Todos estos elementos, al no estar bien delimitados se tornan en riesgos potenciales que pueden conllevar a fallas y la disminución de los niveles de satisfacción del producto o servicio entregado.

Del análisis de diagnóstico realizado, se concluye que la implementación de los lineamientos del PMI en la gestión de proyectos se encuentra en proceso de Desarrollo; solo se observan esfuerzos aislados; el nivel de madurez en la gestión de proyectos está en Estado Inicial y el principal entregable a obtener para pasar a estado Estable es la elaboración de una Metodología.

Para construir un producto de software es importante diferenciar que existen varias metodologías aplicadas en la vida del software; se definen las siguientes:

- La gestión de proyectos de software
- Desarrollo de software (ciclo de vida del software)
- Mantenimiento del software (metodologías poco o casi nunca utilizadas)

El PMBOK del PMI es una guía de fundamentos que describe normas, métodos, procesos y prácticas establecidos, para la gestión y control de proyectos de diferentes ámbitos, que recoge un conjunto de buenas prácticas y estándares, y que permite que las empresas puedan organizarse para alcanzar el máximo beneficio.

CRITERIOS DE EVALUACIÓN

Se manejó el seguimiento y evaluación de un proyecto de desarrollo y cómo se materializa el ciclo de vida del proyecto planteado desde la guía del PMBOK desde sus aspectos fundamentales:

- Identificación de los interesados.
- Desarrollo del acta de constitución del proyecto.
- Procesos de planificación.
- Desarrollo del plan para la dirección del proyecto.
- Recopilación de requisitos.
- Definición del alcance.
- Creación de la EDT (Estructura de descomposición del trabajo).
- Definición de las actividades.
- Planificación de las comunicaciones.
- Realización de aseguramiento de la calidad.
- Adquisición del equipo del proyecto.
- Desarrollo del equipo del proyecto.
- Dirección y control del equipo del proyecto.
- Gestión de las expectativas de los interesados.
- Efecto y desarrollo de adquisiciones.
- Seguimiento y control del trabajo del proyecto.
- Realización del control integrado de cambios.
- Verificación del alcance.

5.3. REGULACIONES Y NORMATIVAS PARA LA PUESTA EN PRÁCTICA DE LA GUÍA DE INTEGRACIÓN:

1. Los proyectos de desarrollo de software deben estar sustentados en un plan de gestión, el cual debe estar previamente aprobado por el patrocinador del proyecto.
2. Se deberá establecer un Contrato previo al comienzo de un proyecto de desarrollo de software donde queden establecidos los compromisos que se adquieren hasta finalizado el proyecto, entre la empresa desarrolladora y el cliente.
3. Si se realizan cambios a lo largo del proyecto, que no se tuvieron en cuenta en el plan de gestión, deben existir procesos de comunicación al director del proyecto o Scrum Master y a los Interesados del proyecto.
4. El control de cambios debe contar con una aprobación de acuerdo a los términos establecidos en el contrato.
5. Cada proyecto deberá contar con una planificación, procesos de inicio y un plan de aseguramiento de la calidad legalizado y aprobado.
7. Antes de cierres de fase o etapas, iteración, release del proyecto, se debe contar con los entregables finalizados y verificar que no estén incompletos.
8. Se debe lograr una comunicación directa del equipo del proyecto con el responsable del proyecto (Product Owner) a fin de poder medir la presencia de riesgos, cambios que afecten a la calidad, tiempo, alcance, recursos, para la toma de decisiones correspondiente.
9. Cada etapa de cierre de la iteración, debe incluir y contener pruebas que certifiquen la calidad y el cumplimiento de los objetivos de entregar el producto que funcione y con la calidad necesaria.
10. Todo proyecto deberá aplicar los procedimientos, estándares, formatos y recomendaciones definidas para el desarrollo de software.

Si bien las metodologías ágiles son más livianas para la gestión y el desarrollo de proyectos, estas no mencionan algunos procesos de gestión que son importantes a lo largo del proyecto, por esta razón, es necesario combinarlas con los lineamientos de la Guía del PMBOK debido a que presenta una estructura más completa con sus grupos de procesos en cada una de las fases de Inicio, Planificación, Seguimiento y Control, y Cierre. Es un error creer que el uso de metodologías ágiles implica no realizar documentación, lo que se hace es documentar la información que el equipo y el director de proyecto consideren necesaria para que se facilite las actividades de dar seguimiento y mantenimiento al proyecto a más de la información que por regulaciones u obligaciones contractuales se deban realizar.

La Guía se realizará a través del uso de la metodología Scrum por lo que mayormente se establecieron pautas en cuanto al cumplimiento de las fechas de entrega, definición de requisitos, cumplimiento, sprints cortos, colaboración e integración con el equipo al igual para el control y manejo de cambios; la consideración de la interacción con el cliente, las pruebas de aceptación, planificación de release y del sprint. Por último, se consideró pertinente el control de impedimentos, la realización del tablero o pizarra scrum así como gráfico de control Burndown y la medición de la satisfacción del cliente a través de las reuniones de trabajo.

La propuesta se orienta a ser una guía para que la gestión del desarrollo de software en la Empresa, sea tratada como un proyecto y no como un simple desarrollo que a más de incrementar la productividad y eficiencia vaya de la mano con la calidad de los productos de software. Además de combinar las mejores prácticas de dirección de proyectos descritos en la guía del PMBOK y

la metodología híbrida ágil SCRUM, que se fusionan, para cumplir con todo el ciclo para la entrega de un buen producto de software, dando lineamientos como por ejemplo: criterios estandarizados para la medición del desempeño, plantillas, comunicación dentro del proyecto, estimación del tiempo, la identificación y la solución de problemas y defectos, así como se aprobará y validará cualquier cambio en los requerimientos.

SCRUM es una metodología ágil que se orienta a la gestión de proyectos las actividades cubiertas serán:

- Respeto de la fechas de entrega
- Definición de requisito
- Cumplimiento de requisitos
- Trabaja en entornos cambiantes
- Sprints cortos.
- Colaboración
- Trabaja en esquema de Integración de cambios
- Tamaño de proyectos pequeños, medianos, grandes.
- Interacción con el cliente alto
- Los equipos son autos organizados.
- Pruebas de aceptación
- Planificación de la versión o release
- Planificación del Sprint
- Velocidad del equipo
- Impedimentos
- Retrospectiva
- Tablero o pizarra Scrum
- Gráfico de Control Burndown
- Satisfacción del cliente a través reuniones y demos presentadas

A continuación se detallan cada uno de los procesos que se consideran necesarios para la creación de la guía de fundamentos para la gestión de proyectos de desarrollo de software para las pequeñas y medianas empresas estudiadas, mayormente desde la aplicación de SCRUM, por estar relacionado además, a la validación de la propuesta:

I. PROCESOS DE INICIO

1.1 Desarrollo del Acta de Constitución del Proyecto, que una vez aprobado por el patrocinador o sponsor, da lugar al inicio del proyecto.

Consideraciones:

Se debe determinar los requisitos de alto nivel al inicio de proyecto de desarrollo de software debido a que el detalle de los requisitos se los obtiene en la planificación de cada Sprint o iteración.

Una vez que el Acta de Constitución fue aprobada por el patrocinado se debe proceder a asignar el equipo del proyecto y los recursos.

Procesos:

Tabla 36: Acta de Constitución del proyecto

Requerimientos	Detalles
Participantes	Clientes Coordinadores o líderes de equipo Usuarios
Responsables	Clientes Scrum Master
Herramientas y técnicas	Criterio de Expertos Validación de ideas
Entradas	Información general de las necesidades funcionales del producto que requiere el cliente
Proceso	Desarrollar el acta de Constitución del proyecto donde se incluye el detalle de aspectos en cuanto a objetivos, acuerdos, alcance, recursos, riesgos, interesados, cronogramas. Aprobación del acta de constitución por el patrocinador
Salidas	Acta de Constitución (Formato propuesto)

Fuente: elaboración propia.

Formato del Acta propuesto

INFORMACIÓN GENERAL		
Nombre de la Empresa		
Nombre del proyecto		
Nombre del Documento		
Responsables:	Elaborado por:	Aprobado por:
DATOS DEL SOLICITANTE		
Empresa solicitante o cliente persona natural		
Objeto social		
Área solicitante		
Nombre del solicitante		
Información del solicitante		
DESCRIPCIÓN DEL PRODUCTO O SERVICIO		
REQUISITOS		
ENTREGABLES		
IDENTIFICACIÓN DE RIESGOS		
INTERESADOS CLAVES		
CRONOGRAMA DE HITOS DEL PROYECTO		
ALCANCE		
PRESUPUESTO		
SPONSOR QUE AUTORIZA EL PROYECTO		

Fuente: elaboración propia.

1.2 Identificación de los interesados del proyecto

Objetivo

El propósito es identificar a los interesados del proyecto de desarrollo de software para tener claro cada uno de los roles que ejercerán dentro del equipo y fuera de él de acuerdo a los intereses del proyecto.

Se muestra el proceso para identificar a los Interesados de un Proyecto de desarrollo de software.

Tabla 37: Proceso de Identificación de los Interesados

Descripción	Detalles
Participantes	Product Owner o Cliente Scrum Master Experto Técnico

Responsables	Product Owner o Cliente Scrum Master
Herramientas y técnicas	Análisis de interesados de cómo intervienen en las diferentes etapas del proyecto Entrevistas Juicio de Expertos Reuniones
Entradas	Acta de Constitución del proyecto Contrato del proyecto (aplica para proyectos de clientes externos) Registros de interesados de proyectos anteriores
Proceso	1. Identificar a los interesados del proyecto tanto internos como externos así como sus roles, departamentos a los que pertenecen, además de su nivel de interés e influencia en el proyecto. 2. Priorizar a los interesados para conocer su nivel de interés en el proyecto y poder canalizar sus expectativas y la comunicación. 3. Generar documentación de los Interesados.
Salidas	Registro de los Interesados del proyecto según el formato propuesto

Fuente: elaboración propia.

Formato para la identificación de interesados

INFORMACIÓN GENERAL		
Nombre de la Empresa		
Nombre del proyecto		
Nombre del Documento		
Responsables:	Elaborado por:	Aprobado por:
IDENTIFICACION DE INTERESADOS		
Nombre del interesado		
Rol	Interno/ Externo	
Departamento/ Empresa		
Contacto		
Influencia		
Firma del responsable		

Fuente: elaboración propia.

Los requerimientos, definición del control de riesgos para una definición de la probabilidad e impacto, guía y lineamientos de cómo priorizar los requerimientos.

Otro punto importante es la base de conocimientos para el registro y recuperación de información de acuerdo a las necesidades para que den soporten a la toma de decisiones.

II. PROCESOS DE PLANIFICACIÓN

Tabla 38: Proceso de planificación

Descripción	Detalles
Participantes	Product Owner o Cliente Scrum Master
Responsables	Product Owner o Cliente Scrum Master
Herramientas y técnicas	Análisis de interesados de cómo intervienen en las diferentes etapas del proyecto Entrevistas Juicio de Expertos Reuniones
Entradas	Plan del direccionamiento del proyecto Registro de interesados
Proceso	1. Análisis del nivel de participación de los interesados desde la identificación realizada. 2. Plan de gestión de los interesados
Salidas	Plan de Gestión

Fuente: elaboración propia.

Formato Plan de Gestión de los Interesados

INFORMACIÓN GENERAL		
Nombre de la Empresa		
Nombre del proyecto		
Nombre del Documento		
Responsables:	Elaborado por:	Aprobado por:
DETALLE DEL PLAN DE GESTIÓN DE LOS INTERESADOS		
Procesos		
Rol	Nivel de participación	Responsabilidades
Experiencias		
Medios o canales de comunicación		
Frecuencia		
Firma del responsable		

Fuente: elaboración propia.

2.1 Planificación del alcance del proyecto

Tabla 39: Planificación del alcance

Descripción	Detalles
Participantes	Product Owner o Cliente Scrum Master como asesores del Product Owner
Responsables	Product Owner
Herramientas y técnicas	Análisis de documentación Reuniones con expertos
Entradas	Acta de Constitución del proyecto Registros de interesados Plan de Gestión de los interesados
Proceso	En base a los documentos de entrada, se deben definir los entregables
Salidas	Plan de Gestión de Alcance

Fuente: elaboración propia.

Formato Alcance del proyecto

INFORMACIÓN GENERAL		
Nombre de la Empresa		
Nombre del proyecto		
Nombre del Documento		
Responsables:	Elaborado por:	Aprobado por:
Descripción del alcance		
Nombre del Release	Registro de mercaderías	Grupos de requerimientos
Entregables		
Criterios de aceptación de entregables		
Fuera de alcance		
Firma del responsable		

Fuente: elaboración propia.

III. HISTORIAS DE USUARIO

Tabla 40: Priorización de historias de usuario

Descripción	Detalles
Participantes	Product Owner o Cliente
Responsables	Product Owner
Herramientas y técnicas	Análisis de documentación
Entradas	Plan de Gestión de alcance
Proceso	Priorización de las historias de usuario, actualizaciones realizadas desde cada sprint
Salidas	Formato propuesto para priorización de historias de usuario

Fuente: elaboración propia.

Formato Priorización Historias de usuario

ID RELEASE		
ID		
Nombre Historia		
Importancia		
Esfuerzo		
Priorizar (de mayor a menor prioridad)		

Fuente: elaboración propia.

3.1 Planificación del sprint

Tabla 41: Proceso para planificar el sprint

Descripción	Detalles
Participantes	Product Owner o Cliente Scrum Master
Responsables	Scrum Master Equipo del proyecto
Herramientas y técnicas	Reuniones Técnicas Criterio de expertos
Entradas	Documento product backlog
Proceso	Definición del tiempo de los sprint
Salidas	Formato para planificación del sprint

Fuente: elaboración propia.

Formato Planificación del sprint

ID		
Nómina Historia		
Esf.		
Sprint		
Estado		
Actividad		
Recurso	Tiempo (Horas)	

Fuente: elaboración propia.

3.2 Desarrollo del Cronograma

Tabla 42: Desarrollo del cronograma

Descripción	Detalles
Participantes	Product Owner o Cliente

	Scrum Master
Responsables	Scrum Master Equipo del proyecto
Herramientas y técnicas	Reuniones Información histórica Criterio de expertos Software para gestión de proyectos
Entradas	Acta de constitución del proyecto Plan de gestión de alcance
Proceso	Desarrollo del cronograma
Salidas	Cronograma del proyecto

Fuente: elaboración propia.

Formato Planificación del sprint

REL
SPRINT
VELOCIDAD EQUIPO
Duración Sprint
Fecha desde
Fecha hasta

Fuente: elaboración propia.

3.3 Costos

Tabla 43: Planificación de costos

Descripción	Detalles
Participantes	Patrocinador Product Owner o Cliente Scrum Master Miembros del equipo
Responsables	Scrum Master Patrocinador Product Owner
Herramientas y técnicas	Reuniones Técnicas Criterio de expertos
Entradas	Cronograma del proyecto Registro de riesgos
Proceso	Costos de recursos humanos y materiales
Salidas	Presupuesto

Fuente: elaboración propia.

Formato Planificación del sprint

Recursos Humanos
Recursos Materiales
Subtotal
Fondo de contingencia
Costos

Fuente: elaboración propia.

3.4 Calidad

Tabla 44: Proceso para planificar la calidad del proyecto

Descripción	Detalles
Participantes	Product Owner o Cliente Scrum Master
Responsables	Scrum Master Product Owner
Herramientas y técnicas	Reuniones Juicio de expertos Diagrama causa efecto
Entradas	Línea base el proyecto Alcance Cronograma Costos Registro de interesados
Proceso	Plan de Gestión de Calidad
Salidas	Plan de Calidad del proyecto

Fuente: elaboración propia.

Formato Planificación del sprint

Alcance
Fuera de alcance
Ambiente hardware
Ambiente software
Equipo de trabajo de prueba
Detalle de las pruebas con herramientas y resultados esperados

Fuente: elaboración propia.

3.5 Roles y responsabilidades

Tabla 45: Roles y responsabilidades

Roles	Responsabilidades
Product Owner	Definición de los requerimientos funcionales Historias de usuario Resolución de cambios y errores
Scrum Master	Gestión de problemas Gestión de reuniones Apoyo y orientación al equipo para la metodología de desarrollo
Experto técnico	Coordinación de actividades con el equipo del proyecto Establecimiento de la arquitectura del proyecto Identificación de riesgos tecnológicos
Desarrolladores	Estimación del tiempo de cada historia de usuario Codificación de las historias de usuario

Fuente: elaboración propia.

IV. EJECUCIÓN, SEGUIMIENTO, CONTROL Y CIERRE

4.1 Ejecución

Tabla 46: Ejecución

Rubros	Descripciones
Participantes	Scrum Master Expertos Interesados
Responsables	Product Owner Equipo
Herramientas y técnicas	Juicio de expertos Reuniones
Entradas	Plan para el direccionamiento del proyecto Plan del sprint a ejecutar Bases de datos Cambios reconocidos con las planificaciones del alcance, costos, calidad
Proceso	Desarrollo de historias de usuario, reuniones de revisión, conformación del equipo, control del avance del sprint así como verificar la aplicación del producto final y la documentación del proyecto
Salidas	Manuales, fuentes de software, desarrollo de las historias de usuario, documentos de registro, actualizaciones

Fuente: elaboración propia.

4.2 Seguimiento y Control

Tabla 47: Seguimiento y Control

Rubros	Descripciones
Participantes	Scrum Master Expertos Interesados
Responsables	Product Owner Equipo
Herramientas y técnicas	Juicio de expertos Reuniones
Entradas	Plan para el direccionamiento del proyecto Plan del sprint a ejecutar Análisis del cronograma, costos y cambios aprobados
Proceso	Actualización de las historias de usuario, seguimiento del cronograma, control de la calidad y registro de costos Reuniones del sprint
Salidas	Informes de desempeño, entregables verificados, actualización de planes de alcance, registro de riesgos.

Fuente: elaboración propia.

Tabla 48: Áreas de estudio

Área	Metodología Ágil	Métodos Clásicos
Equipo de desarrollo	Colaborativos, integrados y en comunicación constante	Enfoque en la planificación
Análisis de Clientes	Elementos determinantes dentro de la gestión de proyectos	Se mide la aptitud y compromiso del cliente
Confiabilidad	Análisis interno de entregables y el equipo de trabajo	Documentación explícita y sistematizada
Entregables, requisitos del cliente	Cambios versátiles	Estabilidad y control de cambios dentro de los tiempos establecidos
Diseño del producto	Sobre los entregables que se van trabajando	Se planifica a futuro

Fuente: elaboración propia.

- Las metodologías ágiles generan beneficios y ventajas para el cliente y para la empresa desarrolladora de los productos. Se considera relevante el nivel de integración de todos los involucrados en el proyecto para la ejecución óptima de los entregables, de forma concatenada e integrada. Se ajustan y están en correspondencia con las exigencias actuales, teniendo un margen que le permite controlar posibles cambios o modificaciones.
- Scrum es una de las metodologías más utilizadas actualmente para el desarrollo de diferentes proyectos de software, sin importar el tamaño y alcance del mismo, debido al grado de flexibilidad que brinda durante todo el proyecto. También es una de las metodologías que permite realizar una oportuna gestión de riesgos en el proyecto, ya que este proceso no es algo independiente a la metodología, sino es una actividad que se realiza durante cada etapa del proyecto ejecutado con esta metodología.
- El equipo multidisciplinario que posee Scrum como característica, permite que la toma de decisiones se realice de forma ágil y de manera oportuna,

logrando así que cada miembro del proyecto tenga la iniciativa correcta para aportarla durante el proyecto, además se fomenta la comunicación constante, ya que es difícil pensar que los proyectos de desarrollo puedan fluir de manera correcta sin este intercambio de información continua; de manera que esto se convierte en algo habitual para los miembros del grupo de desarrollo.

- Una de las grandes ventajas que proporciona Scrum, es la entrega de productos ya funcionales en las primeras iteraciones, lo cual hace que el cliente visualice el compromiso y responsabilidad por parte del resto del grupo de desarrollo, lo que ayuda mucho para que la colaboración del cliente de forma óptima en el proyecto, ayudando al intercambio de información en todo sentido. Esto logra de mejor manera la inclusión del propietario del producto en el grupo de trabajo; promoviendo la responsabilidad compartida para el éxito del proyecto.

De esta manera, por parte de los dos entes involucrados se plantean objetivos que van encaminados hacia la misma meta, sumando esfuerzos para lograrlos.

- Los riesgos en proyectos de desarrollo de software son inevitables, sin embargo, si se toman decisiones correctas, si se maneja adecuadamente la planificación y gestión de actividades, se puede lograr un impacto mínimo y un control sobre ellos. Muchas veces, el análisis de riesgos supone encontrar deficiencias que puedan modificarse sin afectar la calidad de los entregables y producto final con el trabajo organizado del equipo encargado.
- Las metodologías de software no son las que resolverán todos los problemas que se presenten durante los proyectos de desarrollo de software, ni son una guía de cómo resolver dichos inconvenientes; sino más bien son una herramienta, que al usarla de manera adecuada para lograr

su máximo beneficio, serán de mucha ayuda al momento de enfrentar un problema en cualquier tipo de proyecto.

Metodologías Ágiles vs. Tradicionales

En el contexto del desarrollo del software, la importancia de la finalización y concreción exitosa de los proyectos, que el cliente se sienta satisfecho, conlleva a múltiples cambios y modificaciones en las metodologías con que trabajan desde las empresas y equipos, analizando el nivel de adaptación que presentan dichas metodologías al contexto del proyecto.

El éxito del proyecto y la calidad del producto dependen en gran parte de la metodología escogida por el equipo, ya sea tradicional o ágil, donde los equipos maximicen su potencial, aumenten la calidad del producto con los recursos y tiempos establecidos.

Tabla 49: Análisis Comparativo Metodologías ágiles y Tradicionales

Metodologías ágiles	Metodologías Tradicionales
Basadas en heurísticas aplicadas en las prácticas de producción de códigos	Basadas en estándares de acuerdo al contexto en que se desarrollan
Flexibilidad en cuanto a la adaptación a cambios a lo largo del proyecto	Poca flexibilidad a los cambios, mayormente si son imprevistos
Resultado del análisis interno (por el equipo)	Son determinadas por factores externos
Procesos flexibles, sin controles excesivos	Proceso bajo control a partir de normas y reglamentos establecidos
No existen contratos para dar inicio al proyecto	Se sustentan en un contrato prefijado
El cliente se mantiene en interacción con el equipo	El cliente es parte del equipo de desarrollo
Los equipos de trabajo no se componen de muchos miembros	Equipos de muchos miembros
Limitado número de protocolos	Varios protocolos
Pocos roles	Varios roles
El diseño del software no es más importante que su funcionalidad	La arquitectura del software es lo que determina y es relevante

Generalmente se sustentan en proyectos medianos y pequeños	Mayormente se emplean en proyectos de mayor magnitud
--	--

Fuente: elaboración propia.

Tabla 50: Enfoques entre Metodologías

Enfoque de la metodología	
<p>Tradicional: El enfoque se sustenta en el modelo en cascada, con actividades sistematizadas, en un período determinado de tiempo, sin mayor flexibilidad para retroceder y realizar cambios, en fases anteriores o durante el proyecto.</p>	<p>Ágil El desarrollo por medio de iteraciones es desde donde se realizan todas las fases del proceso de desarrollo, por lo que existen entregables que pueden ser utilizados por el cliente en cada iteración, y pueden manejarse cambios y ajustes, para las etapas futuras.</p>
Grupo de Trabajo	
<p>Tradicional: Los roles de los grupos de trabajo no son flexibles, sin la posibilidad de cambios y ajustes; si se tiene en cuenta que el equipo no se considera trascendental para el proceso de desarrollo.</p>	<p>Ágil: La multifuncionalidad y la autoorganización del grupo de trabajo permite la flexibilidad de cambios, modificaciones y ajustes en los roles asignados a cada miembro del equipo. El equipo es altamente importante, y la contribución de cada miembro al proyecto, fomentando la colaboración sistemática.</p>
Planificación	
<p>Tradicional: La planificación comienza desde la etapa inicial, analizada y desarrollada para contrarrestar riesgos en la etapa final de entrega del proyecto. En este sentido, durante el desarrollo de los productos, el grupo de trabajo requiere de su mayor esfuerzo para su desarrollo, anticipando el análisis de tiempo y recursos que procuren ajustarse a la realidad del proyecto.</p>	<p>Ágil: Si bien en este tipo de metodologías también se establece una planificación de trabajo previa, esta debe ser cumplida a medida de lo posible, ya que el cambio y modificación de fechas, recursos o alcance, no afecta el éxito del proyecto. Estas metodologías no enfocan la funcionalidad del sistema en la planificación previa, sino en la adaptación a cambios que pueden presentarse durante el desarrollo del proyecto.</p>
Flexibilidad para cambios	
<p>Tradicional: Las metodologías tradicionales debido a la filosofía de cumplimiento estricto de la planificación realizada previamente, no brindan mayor apertura a realizar cambios en el proyecto, ya que estos afectan a la predictibilidad que se intenta manejar en el desarrollo.</p>	<p>Ágil: Las metodologías ágiles tienen un alto nivel de adaptabilidad a los cambios, tomando a estos no como riesgos altos en la mayoría de los casos, sino como posibilidades para retroalimentar el proyecto y obtener el mayor beneficio.</p>
Requerimientos	
<p>Tradicional: Los requerimientos no son cambiantes o con posibilidad de modificación, es decir, desde el inicio se establece un requerimiento y no se modifica en el transcurso del proyecto, porque formó parte del proceso de planificación y la</p>	<p>Ágil: Los requerimientos pueden ser cambiados al considerarse como mejoras, que agregarán un valor significativo al producto final.</p>

modificación de este, sería tomado como un riesgo alto para el proyecto.	
Riesgos	
Tradicional: Los riesgos se incrementan, en la medida que no existe flexibilidad de cambios, y no se tiene en cuenta la estimación de riesgos y variación en la planeación, considerando los requerimientos de los clientes.	Ágil: El nivel de adaptabilidad a cambios por iteraciones posibilita se acelere la reducción de riesgos, por las relaciones entre iteraciones.
Comunicación	
Tradicional: La comunicación se establece entre los directores y el equipo, el recurso humano no tiene gran importancia para estas metodologías, y la capacidad de decisión es muy limitada, con excepción del líder.	Ágil: La comunicación es imprescindible dados por intercambio de información y retroalimentación entre los miembros del grupo de desarrollo como con el cliente. Se debe recordar que el factor humano es un eje principal para el desarrollo de proyectos.
Documentación	
Tradicional: La documentación exhaustiva que se genera, no aporta gran valor al proyecto.	Ágil: La generación de documentación se realizará solo en el caso de ser necesario, para que de esta manera, el esfuerzo del grupo sea utilizado en otras actividades del proyecto que se crean que aportarán mayor valor al proyecto. La creación de documentación deberá ser manejada por el criterio del grupo.
Cliente	
Tradicional: El cliente como ente externo al grupo de trabajo, es importante porque se acuerdan los requerimientos en la etapa inicial del proyecto, a partir del cual se realizará la negociación contractual, y al cual se le presentará el producto final.	Ágil: El cliente es un miembro más del grupo de trabajo con el cual se mantiene una constante comunicación, para poder conocer las necesidades del negocio y requerimientos
Control	
Tradicional: El control se ejerce sobre los miembros del grupo de una forma muy rigurosa y estricta, con la continua imposición del líder, limitando la capacidad de opinión y autogestión del grupo.	Ágil: Debido a la constante comunicación entre los miembros del grupo, el control se facilita ya que al existir un intercambio constante de información, se puede detectar a tiempo cualquier inconveniente y poder tomar decisiones.
Contrato	
Tradicional: Se sustenta en cumplir siempre con lo planificado inicialmente, por lo que se plantea un contrato con un valor fijo y un tiempo ya preestablecido, tomando en cuenta solamente los imprevistos	Ágil: Se promueve la flexibilidad, por lo que se debe indicar y acordar con el cliente que muchos de los requerimientos pueden estar sujetos a modificación,

comunes de cualquier proyecto, y no aquellos que implicarían cambios drásticos en la planificación.	sobre todo en cuanto a tiempos y costos establecidos, lo que se podrá modificar a medida que el proyecto avance y las necesidades del negocio ameriten cambios
Orientación	
Tradicional: Son orientadas a los procesos, es decir, se centran en el cumplimiento estricto del ciclo de vida de un proyecto de desarrollo de software en forma de cascada.	Ágil: Se orientan hacia las personas, se trata de lograr que las personas entiendan la importancia de su trabajo y su aporte sea significativo.
Tradicional: Las metodologías tradicionales buscan ser predictivas en la mayoría del tiempo de vida del proyecto, por lo que se centran en que la planificación hecha en la etapa inicial del proyecto se cumpla en un 100 por ciento.	Ágil: Se maneja la adaptabilidad a los cambios, evitando problemas que puedan afectar al proyecto.

Fuente: elaboración propia.

El uso de las metodologías ágiles también implica riesgos. El beneficio más relevante es precisamente, el incremento de la calidad y el control de cambios. La calidad de un software se sustenta en la concordancia con los requisitos funcionales y de rendimiento establecidos así como con los estándares de desarrollo y con las características implícitas que se espera de todo software sobre todo su funcionabilidad. Los factores más importantes para su adopción es incrementar la calidad del producto y la productividad en el desarrollo con mayor nivel de comprensión del proyecto, la necesidad de añadir valor producto/ negocio.

Un resultado que puede ser un riesgo, es la disminución en el nivel de participación de stakeholders. Según el Manifiesto Ágil: “Colaboración del cliente negociado sobre contrato” y “Cercana y diaria cooperación entre empresarios y desarrolladores” (Moyota, 2014), son conceptos claves en la aplicación de metodologías ágiles, pero parecen difícilmente alcanzables en diferentes contextos.

Los problemas surgen a partir que los elementos tienen que combinarse e implementarse en la práctica desde circunstancias específicas de cada organización. La ausencia de guías con instrucciones completas y estructuradas para adaptar una metodología ágil a cada caso concreto y dónde y cómo empezar la adopción, impacta evidentemente, en aumentar el grado de incertidumbre en el proceso de adopción.

A modo de conclusión, se puede apreciar que existen indicadores que caracterizan los procesos de desarrollo de software desde metodologías ágiles; es decir, se siguen doce principios que establecen algunas diferencias entre un desarrollo ágil y uno convencional:

1. Se centra en la satisfacción del cliente por las entregas de software que le aporten valor.
2. Existe la flexibilidad para los cambios de requisitos.
3. Funcionabilidad del producto, con el menor intervalo de tiempo posible entre entregas.
4. Coordinación y unidad del equipo de trabajo
5. Diseño y construcción del proyecto con clientes satisfechos.
6. El diálogo directo herramienta para comunicar información en el equipo de trabajo.
7. El software y su funcionabilidad es la principal medida de progreso.
8. La necesidad de verificar la calidad técnica y el diseño para optimización de procesos.
9. Los procesos ágiles para el desarrollo en armonía e integración con los involucrados del proyecto.
10. La simplicidad es esencial.
11. Equipos coordinados, en comunicación y autoorganizados.

12. El equipo puede analizar cómo ser más efectivo y según estas reflexiones ajustar su comportamiento. Estos principios marcan el ciclo de vida de un desarrollo ágil, así como las prácticas y procesos a utilizar.

Una de las características que hace actualmente que el uso de metodologías ágiles sea más común en diferentes tipo de proyectos, es que estas se centran en el factor humano, tanto el equipo de desarrollo como el cliente; además de que las diferentes iteraciones permiten ir desarrollando la constante comunicación, la que debe existir en todo sentido entre los miembros del proyecto. De esta forma, siendo uno de los ejes primordiales el recurso humano en el desarrollo del proyecto, es lo que también ayudará a determinar el éxito o el fracaso del mismo. Hay que tomar en cuenta que no solo es indispensable que cada individuo posea conocimientos, habilidades y virtudes que las pueda manejar de manera individual, sino que además deben tener la capacidad de poder interactuar con el grupo de trabajo y crear un ambiente lo suficientemente estable y cordial, lo que ayudará a que la mejora continua, promovida por las metodologías ágiles, lo que se convertirá en una costumbre en el grupo de trabajo, ayudando a generar retroalimentación, que aporte valor para el proyecto, como para el grupo, y además para cada miembro de manera individual.

Tabla 51: Análisis Comparativo

SCRUM	XP (EXTREME PROGRAMMING)
Las iteraciones de entregas son de 2 a 4 semanas.	Las iteraciones de entrega son de 1 a 3 semanas.
Lo que se termina, funciona y si está bien, ya no se modifica.	Las tareas que se van entregando a los diferentes clientes, son susceptibles de modificaciones.
Cada miembro del <i>Scrum Team</i> trabaja de forma individual.	Los miembros del programan en pareja en un proyecto XP, se complementan.
El <i>Scrum Team</i> trata de seguir el orden de prioridad que marca el Product Owner y en el <i>Sprint Backlog</i> , pueden ser modificadas.	El equipo de desarrollo sigue estrictamente el orden de prioridad de las tareas definidas por el cliente.
Está basada en la administración del proyecto.	Se centra más en la propia programación o creación del producto.

Fuente: elaboración propia.

5.4 Análisis de Resultados

La validación del Modelo de Integración del Project Management Body of Knowledge con las Metodologías Ágiles de Desarrollo de Software permitió a la empresa ADS SOFTWARE CIA LTDA ubicada en el cantón Latacunga, el mejoramiento de la calidad y gestión de los proyectos, desde el caso de evaluación realizado.

La Guía del PMBOK facilitó el conocimiento y acercamiento a la gestión de proyectos y también favorece como metodología, la dirección de proyectos por lo que se tiene en cuenta la necesidad de guiar y orientar a las empresas a que utilicen el PMBOK como metodología, desde el proceso de gestión de proyectos en cinco grupos de procesos: Inicio, Planificación, Ejecución, Seguimiento y Control, Cierre del Proyecto.

Desde los subprocesos de cada etapa de la gestión de proyectos es que se considerará su integración a las metodologías ágiles. Para ello, se toma en cuenta que como se ha mencionado con anterioridad, los proyectos de desarrollo de software mayormente se sustentan en actividades técnicas y no de gestión del proyecto, lo que incide negativamente en el poder cumplir con indicadores de alcance, tiempo y costos (Contreras, 2011).

De acuerdo al análisis realizado, los directores de proyectos consideran que la Guía de Integración podrá contribuir al mejoramiento de falencias existentes mayormente en cuanto al tiempo de los requerimientos, al encontrarse en una situación de regular e insuficiente los tiempos estimados para el desarrollo de los requerimientos de los proyectos de desarrollo de software, que en muchas ocasiones, afecta a factores como costos, recursos y calidad.

De igual manera, la Guía debe enfocarse a potenciar un aspecto que se debe mejorar y es la tendencia a poco frecuente y frecuente, en el nivel de comunicación entre el director de proyecto y el equipo de proyecto, cuando en el Manifiesto Ágil (1980) se menciona que el mejor método de comunicación es cara a cara.

En cuanto a factores que influyen en desfases de proyectos de desarrollo de software, se consideró que la falta de una metodología de dirección y desarrollo de proyectos, es uno de los factores de mayor influencia.

Capítulo 6

CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

- Fundamentar las teorías y prácticas aplicadas por varios autores relacionados con las áreas de conocimiento del *Project Management Body of Knowledge* y las metodologías ágiles para la gestión de proyectos de software, permitió establecer los parámetros de integración a partir del análisis teórico conceptual del ciclo de vida de las metodologías.
- Diagnosticar la situación actual de la dirección de proyectos de desarrollo de software en las empresas de desarrollo de Software que conforman la muestra del estudio, pertenecientes a la región central del Ecuador, permitió conocer que el 30 % de las empresas encuestadas frecuentemente reconocen que los procesos que plantean las metodologías ágiles, son totalmente efectivos en cuanto a actividades de gestión y técnicas, por lo que la integración a metodologías como PMBOK resulta factible si se considera que las actividades de gestión en correspondencia con las metodologías ágiles no se utilizan como los ciclos de entrega, iteración y desarrollo, así como la importancia de que se realice el desarrollo y seguimiento de planes.
- El 40 % de las empresas encuestadas, afirman que ocasionalmente la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles, lo que evidencia que

aún las empresas no se encuentran familiarizadas con la gestión de proyectos por lo que no eligen categorías de mayor frecuencia.

- Los procesos propuestos desde la guía del PMBOK de forma general, se utilizan en frecuencias de ocasionalmente y nunca / rara vez, por lo que se requiere un análisis de la integración de estos procesos.

- Se identificaron los componentes o factores que influyen en la evaluación de metodologías aplicadas en el desarrollo de software y las áreas de conocimiento del *Project Management Body of Knowledge* para la gestión de proyectos de software.

- Los parámetros de integración entre el *Project Management Body of Knowledge* y los métodos ágiles permitieron la sistematización del ciclo de vida de los proyectos de desarrollo de software y formular el modelo de integración de ambas metodologías, se sustentaron mayormente en los aspectos que propone el Manual de Uso en el epígrafe 5.1

- La matriz de evaluación de metodologías ágiles para los proyectos de desarrollo de software y la aplicación del modelo, se tomó en cuenta desde los parámetros mencionados en la Conclusión 6.

- El Modelo de integración que se propone, permitió a la empresa ADS lo siguiente:
 - La gestión eficiente de proyectos
 - Planificación adecuada del ciclo de vida de los proyectos
 - Determinar roles y funciones
 - Medir tiempos
 - Mantener un equilibrio entre desarrollo de actividades de gestión y actividades técnicas.

- La investigación finalmente, no pretende estandarizar la gestión de proyectos en las empresas que conforman la población y muestra a partir de los modelos propuestos, si se considera que no existe asociatividad entre las empresas, no se integran experiencias y conocimientos en la región.

6.2. Recomendaciones

- Se recomienda se tomen en cuenta criterios como que la metodología debe ajustarse a los objetivos mayormente, así como debe poder emplearse en un entorno amplio de proyectos de software. Se consideren requisitos importantes como que la metodología debe funcionar en un entorno dinámico orientado al usuario y debe integrar las distintas fases del ciclo de desarrollo.
- Se utilice XP, aplicando las consideraciones necesarias en cuanto a tener un plan que determine el número de entregas parciales que se le harán al cliente y cada una de dichas entregas que deben ser organizadas en iteración de desarrollo que a su vez, tendrán actividades de gestión propias de cada iteración, por lo que es evidente la necesidad de integrar a metodologías de gestión de proyectos.
- Tomar en cuenta y aplicar que los procesos que plantean las metodologías ágiles, son totalmente efectivos en cuanto a actividades de gestión y técnicas, por lo que la integración a metodologías como PMBOK resulta factible.
- La metodología ágil más utilizada es Scrum por lo que se debe integrar a la guía del PMBOK.
- Se tome en cuenta que la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles.
- Los procesos propuestos desde la guía del PMBOK de forma general, deben utilizarse con mayor frecuencia, lo que requiere un análisis de la integración de estos procesos a los métodos ágiles.

Apéndice A

Diseño de la Encuesta aplicada a los directores o personal encargado del desarrollo de proyectos de las pequeñas y medianas empresas de desarrollo de software de la región central del Ecuador.

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR



SEDE AMBATO

MAESTRIA EN GERENCIA INFORMÁTICA

TEMA: Modelo de Integración del *Project Management Body of Knowledge* con Metodologías Ágiles de Desarrollo de Software

Elaborado por: Ing. Maribel Uribe

Esta encuesta es anónima y personal dirigida a gerente de proyectos, se agradecerá dar su respuesta con la mayor transparencia y veracidad a las diversas preguntas del cuestionario, lo cual permitirá un acercamiento científico al tema de la problemática de la necesidad de mejorar la gestión de proyectos de desarrollo de software.

Con el objetivo de diseñar, un Modelo de Integración del *Project Management Body of Knowledge* con Metodologías Ágiles de Desarrollo de Software, se ha diseñado la siguiente encuesta, dirigido a personal encargado y con experiencia en el desarrollo de proyectos de software:

Empresa:	Ubicación:	
Años de creada:	Ocupación del encuestado:	Experiencia Profesional: De uno a tres años () De tres a cinco años () Más de cinco años ()
<p>1. En el desarrollo de proyectos de software, ¿se requiere organizar el proyecto en cuatro componentes básicos: personal, proceso, producto y tecnología? Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p> <p>Mencione tres razones que justifiquen la selección de su respuesta</p>		
<p>2. ¿El proceso de desarrollo de proyectos está constituido por actividades técnicas y de gestión? Siempre () A veces () Nunca ()</p>		
<p>3. ¿La etapa de gestión implica planificación, seguimiento, control del proyecto y evaluación de resultados? Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p> <p>¿La etapa técnica implica, ingeniería del software aplicada a la obtención del producto? Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>4. ¿Prioriza mayormente en su empresa? - Software funcionando () - Documentación ()</p>		
<p>5. ¿Se pueden complementar las actividades de gestión dentro del desarrollo de proyectos de software con los procesos propuestos por la guía del PMBOK? Si () No () ¿Por qué?</p>		
<p>6. ¿En su empresa, se han sustituido las actividades de gestión por las actividades propuestas en la guía del PMBOK? Si () No () ¿Por qué?</p>		
<p>7. Si su respuesta anterior fue afirmativa, ¿Los procesos de la guía del PMBOK se adaptan a proyectos de desarrollo de software, utilizando solo un subgrupo de estos procesos primarios de acuerdo a las necesidades y alcance que vaya a suplir el proyecto? Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>8. ¿El proceso de gestión de proyectos se organiza en cinco grupos de procesos: inicio, planificación, ejecución, seguimiento y control, y cierre del proyecto? Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>9. ¿El proceso de elaboración del producto, se realiza a partir del diseño y construcción de la estructura de desglose de trabajo del producto? Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		

<p>10. ¿Los procesos de inicio y planificación son ejecutados antes que los procesos propios del producto?</p> <p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>11. ¿Una vez terminada la planeación, se inicia con las actividades propias del producto en paralelo con los procesos de ejecución, seguimiento y control?</p> <p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>12. ¿Al terminar con el proceso de seguimiento y control, se continúa con los procesos de cierre del proyecto?</p> <p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>13. ¿Los procesos de planificación dependen de la actividad técnica de captura de requerimientos?</p> <p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>14. ¿La captura de requerimientos iniciales, se desarrolla en paralelo a la planificación de alcance?</p> <p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>		
<p>15. ¿ La planificación de alcance aborda:</p> <p>- Decisiones del número de iteraciones que se requieren para construir un producto software ()</p> <p>La metodología es tipo cascada, por lo que los procesos técnicos se inician una vez terminado el proceso de planificación ()</p>		
<p>16. Evalúe ¿con qué frecuencia se manifiestan cada uno de los procesos de la guía del PMBOK en el desarrollo de proyectos de software?</p>		
<p>INICIO</p>	<p>Desarrollar el acta de constitución del proyecto</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Identificar a los interesados</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Procesos de planificación</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Desarrollar el plan para la dirección del proyecto</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Recopilar requisitos</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Definir el alcance</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Crear la EDT</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>
	<p>Definir las actividades</p>	<p>Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()</p>

	Secuencia de actividades	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Estimar los recursos de las actividades	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Estimar la duración de las actividades	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Desarrollar el cronograma	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Estimar costos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Determinar el presupuesto	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Planificar la calidad	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Desarrollar el plan de recursos humanos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Planificar comunicaciones	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Planificar la gestión de riesgos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Identificar riesgos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Realizar análisis cualitativo de riesgos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Realizar análisis cuantitativo de riesgos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Planificar la respuesta a los riesgos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Planificar las adquisiciones	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
EJECUCIÓN	Dirigir y gestionar la ejecución del proyecto	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Realizar aseguramiento de la calidad	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Adquirir el equipo del proyecto	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()

	Desarrollar el equipo del proyecto	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Dirigir el equipo del proyecto	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Gestionar las expectativas de los interesados	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Efectuar adquisiciones	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
PROCESOS DE SEGUIMIENTO Y CONTROL	Dar seguimiento y controlar el trabajo del proyecto	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Realizar control integrado de cambios	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Verificar el alcance	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Controlar el alcance	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Controlar el cronograma	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Controlar costos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Realizar control de calidad	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Informar el desempeño	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Dar seguimiento y controlar los riesgos	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Administrar las adquisiciones	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
GRUPO DE PROCESOS DE CIERRE	Cerrar el proyecto o fase	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()
	Cerrar las adquisiciones	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()

17. ¿Cuáles modelos utiliza mayormente para el desarrollo de proyectos?

Modelo en cascada (de forma implícita se encuentran las actividades de gestión) ()

Modelo en espiral (plantea el desarrollo iterativo e incremental, hace explícita las actividades de gestión)

()

18. **¿Si su respuesta anterior fue el modelo en espiral, evalúe en una escala del 1 al 5, considerando uno como nivel mínimo y 5 como nivel máximo, la importancia de los aspectos de gestión dentro del modelo:**

Aspectos	1	2	3	4	5
Planificación					
Gestión de Riesgos					
Gestión de alcance					

19. **¿Las actividades de gestión presentan el problema que no son lo suficientemente detalladas en los procesos de gestión de proyectos de software?**

Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()

20 **¿Considera que la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles?**

Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()

21. **¿Utiliza metodologías ágiles o tradicionales para el desarrollo de proyectos?**

Siempre () A veces () Nunca () En el caso de no utilizar metodologías ágiles, no continúe con la encuesta.

¿Cuáles son las metodologías ágiles que utiliza con mayor frecuencia?

XP	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()
SCRUM	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()
CRYSTAL CLEAR	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()

22. **¿Con qué frecuencia se manifiestan las siguientes actividades de gestión en correspondencia con la metodología ágil que se utilice, en el desarrollo de proyectos de software?**

XP	Ciclos de entrega, iteración y desarrollo	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()
	Trabajos de máximo 40 horas semanales, el juego de la planificación y la rotación del personal en la programación por parejas	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()
	Juego de planificación: desarrollo y seguimiento de planes de entregas, iteraciones y pruebas	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()
SCRUM	Etapa de Juego	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()
Crystal Clear	Un proyecto se divide en entregas, las entregas en iteraciones por lo que en cada uno de los ciclos (proyecto, entrega, iteración) se requieren actividades de gestión	Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()

23. **¿Los procesos que plantean las metodologías ágiles, son totalmente efectivos en cuanto a actividades de gestión y técnicas?**

Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionamente ()

¿Un proyecto bajo XP, ¿ se debe construir un plan que determina el número de entregas parciales que se le harán al cliente y cada una de dichas entregas, se organizan en iteración de desarrollo, que a su vez tendrán actividades de gestión propias de cada iteración?

Nunca/Rara Vez () Frecuentemente () Con mucha frecuencia () Ocasionalmente ()

24. ¿Cuáles de los siguientes requisitos, deben tenerse en cuenta para la selección de la metodología idónea para el desarrollo de proyectos de software, de acuerdo a su experiencia?

1 = Totalmente en desacuerdo

2 = Desacuerdo

3 = indeciso/neutral

4 = De acuerdo

5 = Totalmente de acuerdo

Indicadores	1	2	3	4	5
La metodología debe ajustarse a los objetivos					
La metodología debe cubrir el ciclo entero de desarrollo de software					
La metodología debe integrar las distintas fases del ciclo de desarrollo					
La metodología debe incluir la realización de validaciones					
La metodología debe soportar la determinación de la exactitud del sistema a través del ciclo de desarrollo					
La metodología debe ser la base de una comunicación efectiva					
La metodología debe funcionar en un entorno dinámico orientado al Usuario					
La metodología debe especificar claramente los responsables de Resultados					
La metodología debería de funcionar en un entorno dinámico orientado al usuario					
La metodología debe poder emplearse en un entorno amplio de proyectos software					
La metodología se debe de poder enseñar					
Debe estar soportada en herramientas CASE					
Debe soportar la eventual evolución del sistema					
Debe contener actividades conducentes a mejorar el proceso de desarrollo de software					

Referencias

- Álvaro, Verónica. (2012). *Diseño de una Metodología para el Direccionamiento de Proyectos de Software para la empresa GESTORINC S.A.* Tesis de Magíster en Administración de Empresas con Mención en Gerencia de la Calidad y Productividad, Departamento de Ciencias Administrativas y Contables, Universidad Católica del Ecuador- Matriz (2014).
- Alonso, Germán. Metodología para la gestión de proyectos bajo los lineamientos del Project Management Institute en una empresa del sector eléctrico. Colombia: Bogotá, (2013).
- Kent Beck: Extreme Programming Explained: Embrace Change, Addison. Grupos Avance del Proyecto Utilización de Recursos Cumplimiento de Expectativas. Wesley, (2010)
- Letelier, Patricio. "Metodologías ágiles para el desarrollo de software: XP". Universidad Politécnica de Valencia. Facultad de Informática (2010).
- Méndez, Elvia. Modelo de Evaluación de Metodologías para el Desarrollo de Software, Venezuela, Caracas: Universidad Católica Andrés Bello, (2009).
- Miranda, Juan José Gestión de Proyectos. Identificación, Formulación, Evaluación Financiera, Económica, Social, Ambiental. MB Editores. Bogotá. 1997.
- Palacios, Anabela. Guía de Fundamentos para la Dirección de Proyectos de Desarrollo de Software, con enfoque PMI y los Métodos Ágiles. Ecuador, Quito. Universidad de las Fuerzas Armadas ESPE, (2014)
- Palacios, Juan. Flexibilidad con Scrum. Sexta Edición. Mc Graw- Hill. México, (2010)

Palacios, Juan. Modelo Scrum. Sexta Edición. Mc Graw- Hill. México, (2010)

Pressman, Roger S. Pressman (2009). Software Engineering: A Practitioner's Approach, McGraw-Hill Science/Engineering/Math.

Rico, D. and H. Sayani, "Use of Agile Methods in Software Engineering Education". En: *Agile Conference 2009 Chicago*, (2009.)

Superintendencia de Bancos, Valores y Seguros; Cámara de Comercio Latacunga; Servicio de Rentas Internas (2015)

Toapanta, K. *Método Ágil Scrum, aplicado a la implantación de un sistema informático para el proceso de recolección masiva de información con Tecnología Móvil*. Tesis de Ingeniería de Sistemas e Informática, Departamento de Ciencias de la Computación, Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador (2012).

Resumen Final

Modelo de Integración del *Project Management Body of Knowledge* con Metodologías Ágiles de Desarrollo de Software

Luisa Maribel Uribe Campaña

133 páginas

Proyecto dirigido por: Ing. Darío Javier Robayo Jácome, Mg.

El objetivo es validar un *Modelo de Integración del Project Management Body of Knowledge* con las Metodologías Ágiles de Desarrollo de Software que permita a las pequeñas y medianas empresas de la región central del Ecuador, dedicadas al desarrollo de Software, el mejoramiento de la calidad y gestión de los proyectos. Las limitaciones de la gestión de proyectos, que inciden en el no cumplimiento de requerimientos, tiempo de duración, presupuesto y calidad, aplicación de metodologías sin estructuración y planificación previa; falta de definición de roles y funciones, así como trabajar con modelos no adecuados que posibiliten controlar cambios, en el alcance, planificación y seguimiento de la calidad del producto final. Se utilizó la metodología cuantitativa, que permitió la realización del estudio diagnóstico obteniendo como resultados, que el 40 % de las empresas encuestadas, afirman que ocasionalmente la guía del PMBOK en su quinta edición, adiciona la posibilidad de gestionar ciclos de vida adaptativos en los proyectos, los cuales bien conducidos, pueden brindar el valor agregado de contar con mejor documentación y controles a los métodos ágiles, lo que evidencia que aún las empresas no se encuentran familiarizadas con la gestión de proyectos. El modelo de integración propuesto, al implementarse en un proceso de gestión de proyectos permitió la optimización de tiempos y recursos, calidad, estructuración y planificación óptima. No se pretende estandarizar la gestión de proyectos en las empresas que conforman la población y muestra a partir del modelo propuesto, si se considera que no existe asociatividad entre ellas, no se integran experiencias y conocimientos en la región.