

UNIVERSIDAD TÉCNICA NACIONAL
SEDE CENTRAL
CARRERA DE INGENIERÍA DEL SOFTWARE

TESIS DE GRADUACIÓN

Sometida a consideración del Tribunal Examinador para optar por el título académico de Licenciatura en Ingeniería del Software

**PROPUESTA DE UNA GUÍA DE ADMINISTRACIÓN DE PROYECTOS
BASADA EN ESTÁNDARES INTERNACIONALES PARA MICRO Y
PEQUEÑAS EMPRESAS DESARROLLADORAS DE SOFTWARE DE LOS
TRES CANTONES CON MAYOR CANTIDAD DE EMPRESAS EN LA GRAN
ÁREA METROPOLITANA DE COSTA RICA**

AUTOR:

Daniel Bello Ramírez

TUTOR:

MGT. Joaquín Artavia Chaves

Alajuela, Costa Rica

Mayo, 2023

TRIBUNAL EXAMINADOR

MGT. Ana Cecilia Odio Ugalde
Directora de Carrera

MGT. Joaquín Artavia Chaves
Tutor

MGT. José Alberto Herrera Alfaro
Lector 1

Lic. Máximo Angulo Chacón
Lector 2

DECLARACIÓN JURADA

Yo, **Daniel Bello Ramírez**, mayor, soltero, estudiante de la carrera de Ingeniería del Software, de la Universidad Técnica Nacional, domiciliado en San José, Desamparados, portador de la cédula de identidad n.º 117170075, en este acto, debidamente apercibido y entendido de las penas y consecuencias con las que se castiga, en el Código Penal, el delito de perjurio, ante quienes son el Tribunal Examinador de mi trabajo de Tesis para optar por el título de licenciatura en Ingeniería del Software, juro solemnemente que mi trabajo de investigación titulado: **“PROPUESTA DE UNA GUÍA DE ADMINISTRACIÓN DE PROYECTOS BASADA EN ESTÁNDARES INTERNACIONALES PARA MICRO Y PEQUEÑAS EMPRESAS DESARROLLADORAS DE SOFTWARE DE LOS TRES CANTONES CON MAYOR CANTIDAD DE EMPRESAS EN LA GRAN ÁREA METROPOLITANA DE COSTA RICA”** es una obra original que ha respetado todo lo preceptuado por las leyes penales, así como la Ley de Derechos de Autor y Derechos Conexos, n.º 6683 de 14 de octubre de 1982 y sus reformas, publicada en La Gaceta n.º 226 de 25 de noviembre de, 1982; lo que incluye el numeral 70 de esta ley que advierte: art. 70º: Es permitido citar a un autor transcribiendo los pasajes pertinentes siempre que estos no sean tantos y seguidos, que puedan considerarse como una producción simulada y sustancial, que redunde en perjuicio del autor y de la obra original. Asimismo, estoy advertido que la Universidad Técnica Nacional se reserva el derecho de protocolizar este documento ante notario público. En fe de lo anterior firmo en la ciudad de Alajuela, el día cuatro del mes de mayo del año dos mil veintitrés.

**DANIEL BELLO
RAMIREZ
(FIRMA)**  Firmado digitalmente
por DANIEL BELLO
RAMIREZ (FIRMA)
Fecha: 2023.05.04
20:45:45 -06'00'

Cédula de identidad: 117170075

DEDICATORIA

A Dios y a todos los que me apoyaron en este proceso.

ACTA DE APROBACIÓN



Licenciatura en Ingeniería del Software

Trabajo Final de Graduación

Acta No. 002 - 2023

Acta de la sesión **No. 002**, del día jueves 04 de mayo de 2023 a partir de las 18:00 horas, en periodo del primer cuatrimestre 2023, y en la que el Tribunal Evaluador recibe la sustentación del proyecto de graduación, realizado por el estudiante: **Daniel Bello Ramírez**, portador de la cédula: 117170075, quien opta por el Grado Académico de Licenciatura en Ingeniería del Software, sita, en la Universidad Técnica Nacional, presentando el trabajo final de graduación con el tema:

Propuesta de una Guía de Administración de Proyectos basada en estándares internacionales para micro y pequeñas empresas desarrolladoras de software de los tres cantones con mayor cantidad de empresas en La Gran Área Metropolitana de Costa Rica

Preside el Tribunal la señora Directora de Carrera de Ingeniería del Software, MGt. Ana Cecilia Odio Ugalde, junto con la participación del académico **Joaquín Artavia Chaves**, tutor del trabajo final de graduación, **José Alberto Herrera Alfaro** y **Máximo Angulo Chacón**, lectores del trabajo final de graduación.

La señora presidenta del Tribunal manifiesta que los miembros del mismo leyeron el informe, que los estudiantes acogieron las recomendaciones indicadas y en consecuencia se procede a recibir la sustentación correspondiente, en la que los estudiantes realizan su exposición, sujeto al tiempo establecido. Terminada la misma, se procede a externar comentarios pertinentes al trabajo presentado, se formulan preguntas que fueron respondidas por parte de los estudiantes de manera exitosa.

Concluida la sustentación, el Tribunal solicita a los estudiantes retirarse de la reunión para proceder a la votación secreta. La votación da como resultado: **Aprobado**, con nota de: **9.5**

De nuevo en la reunión, la señora presidenta les comunica el resultado declarando que ya son: Licenciados en Ingeniería del Software, a la vez indica que, conforme a la normativa existente, debe revisar el Reglamento de Trabajos Finales de Graduación de la Universidad Técnica Nacional (disponible en la página web de la UTN), específicamente en el Capítulo

IV, Artículos 38 y 40, donde se indica el procedimiento a seguir para efectuar la entrega de los ejemplares físicos y digitales. Se les recuerda también la obligación de presentarse al ACTO DE GRADUACIÓN, al que serán convocados oportunamente. Se cierra la sesión a las: 19:00 horas del 04 de mayo del presente año.

JOSE ALBERTO HERRERA ALFARO (FIRMA) Firmado digitalmente por JOSE ALBERTO HERRERA ALFARO (FIRMA) Fecha: 2023.05.04 19:45:53 -06'00'

José Alberto Herrera Alfaro
Miembro del Tribunal Evaluador Lector

JOAQUIN ALBERTO ARTAVIA CHAVES (FIRMA) Firmado digitalmente por JOAQUIN ALBERTO ARTAVIA CHAVES (FIRMA) Fecha: 2023.05.05 08:09:54 -06'00'

Joaquín Artavia Chaves
Miembro del Tribunal Evaluador Tutor

MAXIMO ALBERTO ANGULO CHACON (FIRMA) Firmado digitalmente por MAXIMO ALBERTO ANGULO CHACON (FIRMA) Fecha: 2023.05.04 20:08:05 -06'00'

Máximo Angulo Chacón
Miembro del Tribunal Evaluador Lector

ANA CECILIA ODIU UGALDE (FIRMA) Firmado digitalmente por ANA CECILIA ODIU UGALDE (FIRMA) Fecha: 2023.05.04 19:11:41 -06'00'

Ana Cecilia Odiu Ugalde
Directora de Carrera

DANIEL BELLO RAMIREZ (FIRMA) Firmado digitalmente por DANIEL BELLO RAMIREZ (FIRMA) Fecha: 2023.05.06 11:53:42 -06'00'

CARTA DE AUTORIZACIÓN DEL TUTOR



Carrera Licenciatura en Ingeniería del Software con salida lateral al Bachillerato en Software y al Diplomado en Tecnologías Informáticas

Nota de aval inicial

Por este medio, hago constar que el documento con el nombre “PROPUESTA DE UNA GUÍA DE ADMINISTRACIÓN DE PROYECTOS BASADA EN ESTÁNDARES INTERNACIONALES PARA MICRO Y PEQUEÑAS EMPRESAS DESARROLLADORAS DE SOFTWARE DE LOS TRES CANTONES CON MAYOR CANTIDAD DE EMPRESAS EN LA GRAN ÁREA METROPOLITANA DE COSTA RICA “ elaborado por el estudiante Daniel Bello Ramírez portador de la cédula de identidad número 117170075 ha cumplido con elementos propios de la especialidad, permitiendo la detección de problemas de índole teórica y práctica, el empleo de los conocimientos adquiridos en el transcurso de la preparación profesional.

Además, ha permitido el fortalecimiento y aplicación de las competencias adquiridas durante su formación universitaria, aplicando técnicas y métodos de investigación básica y aplicada conforme las políticas de investigación definidas por la Universidad Técnica Nacional en el análisis, planteamiento y resolución del problema.

El enriquecimiento de la carrera y el fortalecimiento de la universidad, mediante este tipo de aportes investigativos son fundamentales para el fortalecimiento de la Institución dentro del contexto universitario costarricense, contribuyendo con el desarrollo de la comunidad nacional. Por lo anterior, le doy el aval a este documento para que sea trasladado al análisis y revisión exhaustiva por parte de los lectores que la dirección haya definido.

Sin más por el momento suscribo

JOAQUIN
ALBERTO ARTAVIA
CHAVES (FIRMA)
Firmado digitalmente por
JOAQUIN ALBERTO ARTAVIA
CHAVES (FIRMA)
Fecha: 2023.02.10 16:01:39
-06'00'

Joaquín Artavia Chaves

Cédula de identidad: 204900534

Número colegiado 1190

CARTA DE AUTORIZACIÓN DE LECTORES



Carrera Licenciatura en Ingeniería del Software con salida lateral al Bachillerato en Software y al Diplomado en Tecnologías Informáticas

Nota de validación del lector

Por este medio, en calidad de lector del documento "PROPUESTA DE UNA GUÍA DE ADMINISTRACIÓN DE PROYECTOS BASADA EN ESTÁNDARES INTERNACIONALES PARA MICRO Y PEQUEÑAS EMPRESAS DESARROLLADORAS DE SOFTWARE DE LOS TRES CANTONES CON MAYOR CANTIDAD DE EMPRESAS EN EL GRAN ÁREA METROPOLITANA DE COSTA RICA" doy fe de que los ajustes y variaciones propuestas por los lectores, han sido tomados en cuenta en la consistencia y redacción técnica del documento final.

Por lo anterior es que doy mi anuencia para que el documento pase a la etapa final de la defensa, previa revisión filológica.

Sin más por el momento, suscribo

JOSE ALBERTO HERRERA ALFARO (FIRMA)

Firmado digitalmente por
JOSE ALBERTO HERRERA
ALFARO (FIRMA)
Fecha: 2023.03.31
15:27:19 -06'00'

Ing. José Alberto Herrera Alfaro, MGT

Profesor Carrera Ingeniería del Software

Cédula de identidad: 110870760

Número de colegiado: 2288

Cc.- Archivo

Nota de validación del lector

Por este medio, en calidad de lector del documento “PROPUESTA DE UNA GUÍA DE ADMINISTRACIÓN DE PROYECTOS BASADA EN ESTÁNDARES INTERNACIONALES PARA MICRO Y PEQUEÑAS EMPRESAS DESARROLLADORAS DE SOFTWARE DE LOS TRES CANTONES CON MAYOR CANTIDAD DE EMPRESAS EN EL GRAN ÁREA METROPOLITANA DE COSTA RICA” doy fe de que los ajustes y variaciones propuestas por los lectores, han sido tomados en cuenta en la consistencia y redacción técnica del documento final.

Por lo anterior es que doy mi anuencia para que el documento pase a la etapa final de la defensa, previa revisión filológica.

Sin más por el momento, suscribo

MAXIMO ALBERTO
ANGULO CHACON
(FIRMA)

Firmado digitalmente por
MAXIMO ALBERTO ANGULO
CHACON (FIRMA)
Fecha: 2023.03.31 21:08:24 -06'00'

Máximo Angulo Chacón

Cédula de identidad 1-1079-0557

Número de colegiado 3382

Cc.- Archivo

CARTA DE APROBACIÓN DEL FILÓLOGO

Cartago, 7 de abril de 2023

Los suscritos, Elena Redondo Camacho, mayor, casada, filóloga, incorporada a la Asociación Costarricense de Filólogos con el número de carné 0247, portadora de la cédula de identidad número 3-0447-0799 y, Daniel González Monge, mayor, casado, filólogo, incorporado a la Asociación Costarricense de Filólogos con el número de carné 0245, portador de la cédula de identidad número 1-1345-0416, ambos vecinos de Quebradilla de Cartago, revisamos el trabajo final de graduación que se titula: *Propuesta de una guía de administración de proyectos basada en estándares internacionales para micro y pequeñas empresas desarrolladoras de software de los tres cantones con mayor cantidad de empresas en la Gran Área Metropolitana de Costa Rica*, sustentado por Daniel Bello Ramírez.

Hacemos constar que se corrigieron aspectos de ortografía, redacción, estilo y otros vicios del lenguaje que se pudieron trasladar al texto. A pesar de esto, la originalidad y la validez del contenido son responsabilidad directa de la persona autora.

Esperamos que nuestra participación satisfaga los requerimientos de la Universidad Técnica Nacional.

ANA ELENA
REDONDO
X CAMACHO (FIRMA) Firmado digitalmente por ANA ELENA REDONDO CAMACHO (FIRMA)
Fecha: 2023.04.07 14:29:15 -06'00'

Elena Redondo Camacho
Filóloga - Carné ACFIL n.º 0247

DANIEL ALBERTO
GONZALEZ
X MONGE (FIRMA) Firmado digitalmente por DANIEL ALBERTO GONZALEZ MONGE (FIRMA)
Fecha: 2023.04.07 14:29:39 -06'00'

Daniel González Monge
Filólogo - Carné ACFIL n.º 0245

ÍNDICE GENERAL

1	Definición del problema.....	2
1.1	Estado del arte.....	2
1.2	Planteamiento del problema.....	4
1.3	Pregunta problema	11
1.4	Hipótesis	11
1.5	Objetivos.....	12
1.5.1	Objetivo general	12
1.5.2	Objetivos específicos	12
1.6	Alcance y limitaciones	13
1.6.1	Alcance	13
1.6.2	Limitaciones	13
1.7	Matriz de congruencia	15
2	Marco teórico.....	19
2.1	Micro, pequeñas y medianas empresas en Costa Rica.....	19
2.1.2	Principales sectores económicos	22
2.1.2.1	Comercial	22
2.1.2.2	De servicios	23
2.1.3	Desarrolladoras de software	24
2.2	Ingeniería de Software.....	26
2.2.1	Ciclo de vida del software	28
2.2.1.1	Requerimientos	30
2.2.1.2	Diseño	33
2.2.1.3	Codificación	36
2.2.1.4	Pruebas	39
2.2.1.5	Mantenimiento y evolución	41
2.2.2	Modelos de desarrollo.....	42
2.2.2.1	Cascada	43

2.2.2.2	Incremental	46
2.2.2.3	Evolutivo-Iterativo	48
2.2.3	Metodologías de desarrollo ágil	49
2.2.3.1	El manifiesto ágil	50
2.2.3.2	Marcos de trabajo	51
2.2.4	Tipos de software.....	56
2.2.4.1	De Sistemas	56
2.2.4.2	Aplicación	57
2.2.4.3	Incrustado	57
2.2.4.4	Inteligencia artificial	57
2.3	Administración de proyectos de software.....	58
2.3.2	Ciclo de vida del proyecto.....	60
2.3.3	Etapas del proyecto	61
2.3.3.1	Definición y alcance	61
2.3.3.2	Tiempo	65
2.3.3.3	Costo	69
2.3.3.4	Recursos humanos	73
2.3.3.5	Calidad	75
2.3.3.6	Riesgos	76
2.3.3.7	Control	78
2.3.3.8	Cierre	79
2.3.4	Estándares	81
2.3.4.1	El estándar para la dirección de proyectos de PMI y la guía PMBOK	81
2.3.4.2	CMMI	86
2.3.4.3	ISO	92
2.3.4.4	IEEE	98

3	Marco metodológico	102
3.1	Enfoque de investigación	102
3.2	Tipo de investigación	102
3.2.1	Fuentes de investigación	103
3.2.1.1	Fuentes primarias	103
3.2.1.2	Fuentes secundarias	103
3.3	Técnicas de recolección de datos	104
3.4	Consideraciones éticas	105
3.5	Población	105
3.6	Criterios de inclusión	106
3.7	Criterios de exclusión	107
3.8	Muestra	107
3.9	Matriz metodológica.....	109
4	Análisis situacional	116
4.1	Consideraciones sobre el instrumento aplicado.....	116
4.2	Generalidades de las empresas analizadas	116
4.3	121
4.4	Metodologías que se utilizan	122
4.5	Alcance	123
4.6	Tiempo	126
4.7	Costo y recursos.....	131
4.8	Riesgos	133
4.9	Cierre	135
4.10	Calidad	137
5	Conclusiones y recomendaciones	142
5.1	Objetivo específico n.º 1: Conocer la forma en la que se gestionan los proyectos en las mypes del país que desarrollan proyectos de software mediante consultas a las partes interesadas	142
5.1.1	Conclusiones.....	142
5.1.2	Recomendaciones	144

5.2	Objetivo específico n.º 2: Identificar los estándares que más se utilizan por las empresas durante su proceso de administración de proyectos de software mediante la recopilación de datos	145
5.2.1	Conclusiones.....	145
5.2.2	Recomendaciones	146
5.3	Objetivo específico n.º 3: Determinar el estándar de administración de proyectos más adecuado para este tipo de empresas mediante el análisis de la información que se recopiló.....	146
5.3.1	Conclusiones.....	146
5.3.2	Recomendaciones	147
6	Propuesta	150
6.1	Objetivo.....	150
6.2	Alcance	150
6.3	Introducción	151
6.4	Entrega de valor en los proyectos de software	152
6.5	Dominios de desempeño del proyecto y su implementación	154
6.5.1	Interesados	155
6.5.2	Equipo	159
6.5.3	Enfoque de desarrollo y ciclo de vida	162
6.5.4	Planificación	165
6.5.4.1	Alcance	166
6.5.4.2	Tiempo	169
6.5.4.3	Presupuesto	173
6.5.4.4	Recursos físicos y Adquisiciones	174
6.5.5	Trabajo del proyecto	176
6.5.6	Entrega.....	177
6.5.6.1	Entregables	177
6.5.6.2	Calidad	178
6.5.7	Medición.....	181
6.5.7.1	Tableros de control	185

6.5.8	Incertidumbre	187
6.6	Adaptación.....	191
7	REFERENCIAS BIBLIOGRÁFICAS.....	196
8	Anexos y apéndices	201

ÍNDICE DE TABLAS

Tabla 1: Clasificación de las PYMES por Tipo de Actividad según el Reglamento a la Ley de Fortalecimiento de las Pequeñas y Medianas Empresas número 8262.	4
Tabla 2: Matriz de congruencia	15
Tabla 3: Cantidad de pymes por tamaño del año 2019	21
Tabla 4: Porcentaje de pymes del sector comercial en el año 2019.....	23
Tabla 5: Porcentaje de pymes del sector comercial en el año 2019.....	24
Tabla 6: Habilidades del equipo de trabajo	73
Tabla 7: Control de actividades y horas del proyecto.....	73
Tabla 8: Proceso de planificación del proyecto según CMMI	89
Tabla 9: : Entradas y salidas para las actividades de la administración de proyectos según la normativa ISO 21500.....	93
Tabla 10: Formato de un plan de proyecto de software según la norma IEEE 1058	98
Tabla 11: Empresas mypes por región socioeconómica.....	105
Tabla 12: Empresas mypes en los tres cantones con mayor concentración Empresas mypes en los tres cantones con mayor concentración	106
Tabla 13: Variables que se utilizan en el cálculo de la muestra.....	108
Tabla 14: Matriz metodológica.....	109
Tabla 15: Software que se utiliza por las empresas para administrar sus proyectos.....	121
Tabla 16: Criterios que se utilizan en el momento de definir el alcance del proyecto	125
Tabla 17: Técnicas de estimación del tiempo de las actividades de los proyectos	130
Tabla 18: Proceso de adaptación según el PMBOK	193

ÍNDICE DE FIGURAS

Figura 1: Factores de éxito en la administración de proyectos	8
Figura 2: Ciclo de vida de un proyecto	11
Figura 3: Fases del ciclo de vida del software.....	29
Figura 4: Ciclo de vida en cascada.....	44
Figura 5: Modelo en V.....	45
Figura 6: Modelo incremental	46
Figura 7: Modelo en espiral	49
Figura 8: El ciclo Scrum	52
Figura 9: El ciclo XP.....	54
Figura 10: Pizarra Kanban.....	55
Figura 11: Componentes principales de un proyecto	59
Figura 12: Ciclo de vida de un proyecto	61
Figura 13: EDT para el proyecto de desarrollo de <i>software</i>	64
Figura 14: Plan de comunicaciones.....	65
Figura 15: Diagrama de red del proyecto	66
Figura 16: Relación de precedencia entre actividades.....	67
Figura 17: Diagrama de Gantt	67
Figura 18: Estructura de división de costos.....	72
Figura 19: Matriz de roles y responsabilidades	74
Figura 20: Comparación entre la guía PMBOK sexta y séptima edición	82
Figura 21: Niveles de madurez de CMMI	88
Figura 22: Fórmula para calcular el tamaño de la muestra.....	108
Figura 23: Estructura de la guía PMBOK – Séptima Edición	152
Figura 24: Matriz poder-influencia	157
Figura 25: Ciclo de vida en enfoques predictivos.....	163
Figura 26: Ciclo de vida en enfoques adaptativos ágiles	163
Figura 27: EDT para proyectos de desarrollo de <i>software</i>	172
Figura 28: Diagrama de red del proyecto	172
Figura 29: Diagrama de Gantt	173

Figura 30: EDT con el costo de cada una de las actividades	174
Figura 31: Diagrama de estados de ánimo.....	185
Figura 32: Página principal del sistema Jira	185
Figura 33: Matriz de riesgos	189
Figura 34: Proceso de adaptación según la guía PMBOK.....	192

ÍNDICE DE GRÁFICOS

Gráfico 1: Tamaño aproximado de las empresas consultadas	117
Gráfico 2: Relación laboral de las personas que participaron de la encuesta .	118
Gráfico 3: Nivel de conocimiento sobre administración de proyectos.....	119
Gráfico 4: Capacitaciones sobre administración de proyectos	119
Gráfico 5: Tipo de software desarrollado por las empresas	120
Gráfico 6: Aplicativos para gestionar los proyectos.....	121
Gráfico 7: Metodologías de desarrollo en las empresas	122
Gráfico 8: Metodologías de administración de proyectos que utilizan las empresas	123
Gráfico 9: Interesados del proyecto	124
Gráfico 10: Documentación del alcance del proyecto	124
Gráfico 11: Criterios que se utilizan en el momento de definir el alcance del proyecto	125
Gráfico 12: Avance entre los clientes del proyecto	126
Gráfico 13: Duración de las actividades del proyecto	128
Gráfico 14: Cronograma de actividades del proyecto	128
Gráfico 15: Responsabilidades de los miembros del equipo	129
Gráfico 16: Rendimiento de los miembros del equipo.....	130
Gráfico 17: Existencia del plan de adquisiciones	131
Gráfico 18: Existencia del plan de costos.....	132
Gráfico 19: Elaboración del presupuesto.....	132
Gráfico 20: Monitoreo de los costos	133
Gráfico 21: Documentación de los riesgos	134
Gráfico 22: Monitoreo de los riesgos	134
Gráfico 23: Mitigación del riesgo	135
Gráfico 24: Documentación del cierre del proyecto.....	136
Gráfico 25: Documentación de las lecciones aprendidas	137
Gráfico 26: Procedimiento de calidad para los productos	138
Gráfico 27: Registro de habilidades técnicas de los miembros del equipo	138

Gráfico 28: Evaluación de calidad de los productos.....	139
Gráfico 29: Informe de calidad del proyecto.....	140

LISTADO DE ABREVIATURAS

- **CMMI:** Capability Maturity Model Integration.
- **IEEE:** Instituto de Ingenieros Eléctricos y Electrónicos.
- **ISO:** International Organization for Standardization.
- **Mypes:** Micro y Pequeña Empresa.
- **PMBOK:** Project Management Body of Knowledge.
- **PMI:** Project Management Institute.
- **Pymes:** Pequeña y Mediana Empresa.
- **TFG:** Trabajo final de graduación.
- **TI:** Tecnologías de información.

RESUMEN EJECUTIVO

En el presente trabajo final se plantea una propuesta de administración de proyectos para micro y pequeñas empresas (mypes) costarricenses desarrolladoras de software, con base principalmente en la guía PMBOK séptima edición. Algunas de estas compañías están catalogadas y registradas ante el registro público costarricense como micro, pequeñas y medianas empresas (pymes).

Esta propuesta se elabora mediante la recopilación de datos desde fuentes primarias y secundarias. Es decir, por medio de la revisión de la literatura y con la participación de las empresas, para lo cual se aprovechan los instrumentos de recolección de datos establecidos en el marco metodológico.

Con la elaboración de este estudio se pretende guiar a las empresas hacia un proceso estandarizado de administración de proyectos de software. Lo anterior para que se administren de forma eficiente, con base en el principal objetivo de la guía PMBOK séptima edición: **entregar valor**.

Al tener un proceso estandarizado la empresa puede cumplir más fácilmente con los objetivos debido a que la razón de ser de un proyecto es satisfacer la necesidad del cliente. De ahí la relevancia de lo dicho por Gray y Larson (2009): “Un proyecto tiene características que lo diferencian de otras tareas o procesos en la organización, por ejemplo, tiene un objetivo establecido, un inicio y fin y tiene requerimientos específicos de alcance, costo y desempeño” (p. 5).

En cuanto al marco teórico que sustenta la teoría del estudio, se presenta una revisión de la literatura que aborda la disciplina de administración de proyectos: ciclo de vida, metodologías, autores, integración con metodologías ágiles y otros. Además, se expone información relacionada con el proceso de desarrollo de software.

Por otra parte, el abordaje metodológico se enfoca en analizar el proceso de gestión de proyectos de software en las mypes, lo cual incluye metodologías de desarrollo y administración de proyectos que se utilizan, el nivel de conocimiento de los miembros del equipo en esta disciplina, la capacidad técnica,

los estudios realizados y otros elementos más, mediante diferentes instrumentos de recolección de datos. Lo anterior con el objetivo de determinar cuál es el mejor estándar que se adapta a las empresas mypes.

Por otra parte, las conclusiones y recomendaciones están orientadas a responder a los objetivos de investigación y a recomendar una serie de sugerencias para las empresas desarrolladoras de software que entran en la categoría de mypes.

Términos clave: Administración de proyectos, metodologías de administración de proyectos, metodologías de desarrollo de software, PMI, PMBOK, metodologías ágiles, Scrum y pymes.

ABSTRACT

In this thesis, a project management proposal for small Costa Rican companies that develop software is presented, based mainly on the PMBOK guide, seventh edition. Some of these companies are listed and registered with the Costa Rican public registry as SMEs.

This proposal was developed by collecting information from primary and secondary sources, that is, by reviewing the literature and with the participation of companies, taking advantage of the data collection instruments duly established.

With the development of this research, it is intended to guide companies towards a standardized software project management process, so that they are managed efficiently, based on the main objective of the PMBOK guide, seventh edition: **deliver value**.

By having a standardized process, the company could more easily meet the objectives because the main objective of a project is to satisfy the customer's need. Hence the relevance of what was said by Gray & Larson (2009) "A project has characteristics that differentiate it from other tasks or processes in the organization, for example, it has an established objective, a start and end and has specific scope requirements, cost and performance" (p. 5).

Regarding the theoretical framework that supports the research theory, a review of the literature that addresses the discipline of project management is presented: life cycle, methodologies, authors, integration with agile methodologies and others. In addition, information related to the software development process (life cycle, agile methodologies, and others) is exposed.

On the other hand, the methodological approach is focused on analyzing the software project management process of this type of company, which includes development methodologies and project management used, the level of knowledge of team members in this discipline, the technical capacity, the studies carried out and other elements, through different information collection instruments, the above with the objective of determining which is the best standard that adapts to this type of company.

On the other hand, the conclusions and recommendations are aimed at responding to the research objectives and recommending a series of suggestions for companies that fall within the category of small business.

Keywords: Project Management, Project Management Methodologies, Software Development Methodologies, PMI, PMBOK, Agile Methodologies, Scrum, SMEs.

CAPÍTULO I:
DEFINICIÓN DEL PROBLEMA

1 Definición del problema

1.1 Estado del arte

El término proyecto se refiere a la asignación de recursos para realizar actividades dentro de un periodo limitado para lograr metas específicas como respuesta a un problema o necesidad (López y Lankenau, 2017, p. 44). A partir de la premisa descrita en el párrafo anterior, el proceso de desarrollo de software se considera como un proyecto en la organización. Por esto, las empresas que desarrollan software implementan metodologías de administración de sus proyectos de desarrollo.

Con base en lo anterior y por relevancia de interés de esta investigación, se encontraron propuestas diversas de los últimos seis años sobre metodologías de administración de proyectos de software para organizaciones en el territorio costarricense y otras investigaciones en el ámbito internacional.

En el año 2010 se publicó un artículo en la revista indexada brasileña *Gestión y Producción* titulado *Madurez en la gestión de proyectos en pequeñas empresas de desarrollo software del Centro de Alta Tecnología de São Carlos* en donde se realizó un diagnóstico sobre cómo las pequeñas compañías de desarrollo de software del estado de San Carlos manejaban sus proyectos, con énfasis en sus debilidades como la falta de comunicación y gestión de riesgos. Además, se propuso una metodología para que las pequeñas empresas pudieran medir la madurez de los proyectos a través del tiempo.

Por otra parte, en el año 2017 se realizó una tesis de maestría en el Tecnológico de Costa Rica (de ahora en adelante TEC) titulada *Propuesta de metodología para la gestión de proyectos de software en la empresa Go-Labs* elaborado por Huber Espinoza Palma. Esta propuesta para la compañía de capital costarricense Go Labs, planteó mejorar su proceso de administración de proyectos utilizando la metodología PMBOK. Este documento incluyó las áreas de conocimiento de alcance, tiempo, costo, calidad, recursos, gestión de adquisiciones y otros.

En el marco metodológico se propusieron entrevistas a las personas colaboradoras de la empresa que laboran en software, para medir su

conocimiento en gestión de proyectos. Sobre los resultados se obtuvo que la empresa carecía de una metodología definida para la administración de sus proyectos y las personas colaboradoras en su mayoría desconocían aspectos clave en la gestión de proyectos (Espinoza Palma, 2017).

Adicionalmente, en el año 2016 se publicó un artículo en la revista estadounidense Science Direct titulado Toma de decisiones en la gestión de proyectos de software: una revisión sistemática de la literatura en donde se analizó el impacto de la toma de decisiones en la gestión de proyectos de software. Además de las nuevas formas de comunicación al implementar metodologías ágiles de desarrollo de software.

Por otra parte, en el año 2017 se realizó otra tesis de maestría también en el TEC titulada Propuesta de una metodología de gestión de la calidad que apoye la administración de proyectos de software en la empresa Avantica Technologies, elaborado por Óscar Rivera Céspedes. En este trabajo se elaboró una propuesta, no solo de administración de proyectos basada en la metodología PMBOK, sino que también contempló el tema de la calidad.

Como parte de los resultados de la investigación se encontró que la empresa no aplicaba en sus procesos de calidad y gestión de proyectos los estándares de la industria del software. Por lo tanto, esta propuesta tenía como objetivo corregir esos errores mediante las recomendaciones sugeridas por la guía PMBOK (Rivera Céspedes, 2017, p. 175).

En el ámbito de pymes existe una publicación en la Revista Electrónica de Computación, Informática, Biomédica y Electrónica (Recibe) de México titulada Desarrollo de herramientas enfocadas en ayudar a las pymes de desarrollo de Software en la implementación de buenas prácticas de gestión de proyectos. En este estudio se abordan temas como la perspectiva de las pymes en el ámbito de administración de proyectos de software y la gestión de riesgos mediante sistemas automatizados. Se determinó que la herramienta propuesta para la gestión de riesgos proporciona a las pymes el conocimiento de un conjunto de técnicas y herramientas que permiten llevar a cabo una gestión de riesgos básica,

con lo cual se impulsa la implementación de buenas prácticas de ingeniería de software enfocadas en la gestión de riesgos (García et al., 2017).

En el año 2020 se desarrolló una tesis de licenciatura, en la Universidad Latina de Costa Rica denominada Propuesta de una metodología de trabajo basada en un marco de trabajo ágil para el uso y la aplicación de bases de datos multiplataforma, aplicando los principios y buenas prácticas de la administración de proyectos del Project Management Institute, para el centro de desarrollo de Software de la Universidad Latina de Costa Rica, elaborado por Marco Antonio González Muñoz. Esta investigación realizó una propuesta metodológica de trabajo para el centro de desarrollo de software de la Universidad Latina con base en buenas prácticas de la administración de proyectos del Project Management Institute (PMI).

Sobre los resultados de la investigación que se mencionó en el párrafo anterior se adaptó el estándar internacional PMBOK a la propuesta de trabajo mediante una metodología que estandarice los perfiles, lineamientos, procedimientos y buenas prácticas para el desarrollo de base de datos (González Muñoz, 2020).

En el nivel nacional, en el momento de documentar esta investigación no se encontraron otros estudios que se relacionan con la administración de proyectos en empresas desarrolladoras de software pymes.

1.2 Planteamiento del problema

En la actualidad, las pymes del país se han consolidado. Se estima que en Costa Rica alrededor del 98 % de las empresas productivas cuentan con menos de 100 empleados, lo que las define como micro, pequeñas y medianas empresas (pymes) (Monge, 2009). Las pymes en el país están categorizadas según su actividad comercial, algunas de las cuales se detallan en la Tabla 1:

Tabla 1

Clasificación de las pymes por tipo de actividad según el Reglamento a la Ley de Fortalecimiento de las Pequeñas y Medianas Empresas n.º 8262

Tipo de actividad	Descripción
Industrial	<ul style="list-style-type: none"> ● Transformación física y química de materiales para la elaboración de objetos. ● Alteración, construcción y renovación de productos. ● Actividades manufactureras.
Comercial	<ul style="list-style-type: none"> ● Negociación de compra y venta de mercancías.
Servicios	<ul style="list-style-type: none"> ● Actividades que se realizan para obtener retribución. ● No se pueden incluir en el inventario.
Tecnologías de información y comunicación	<ul style="list-style-type: none"> ● Recursos, procedimientos y técnicas para el procesamiento y transferencia de la información.

Para determinar si una empresa entra en la categoría pymes, la Ley n.º 8262, Reglamento a la Ley de Fortalecimiento de las Pequeñas y Medianas Empresas, establece una fórmula que la cataloga como micro, pequeña o mediana empresa. Adicionalmente, la Caja Costarricense de Seguro Social (en adelante CCSS) establece que una empresa entra en la categoría de microempresa si posee de 1 a 5 trabajadores, pequeña si posee de 6 a 30 trabajadores y mediana si posee más de 31 empleados (Cámara de Comercio de Costa Rica, s. f.).

Por otra parte, la clasificación de actividades económicas de Costa Rica del Instituto Nacional de Estadística y Censos (en adelante INEC) categoriza las

diferentes actividades económicas en subsectores. El sector que se encarga del desarrollo de software está catalogado como **actividades de programación informática**.

Según los registros del Ministerio de Economía, Industria y Comercio (de ahora en adelante MEIC) existen 232 empresas tipo pymes en el ámbito nacional que se dedican a este tipo de actividad. Dentro de ese grupo están registradas **219 compañías** que entran en la categoría de micro y pequeñas empresas (en adelante mypes). De esa cantidad, la mayor concentración de empresas (38) se encuentra en los cantones de **Curridabat, Escazú y Montes de Oca**.

Con base en lo anterior, las empresas mypes que desarrollan software pueden adoptar metodologías de administración para sus proyectos que les garanticen que su gestión se haga de manera eficiente. Según menciona Somerville (2011):

Los proyectos necesitan administrarse porque la ingeniería de software profesional está sujeta siempre a restricciones organizacionales de presupuesto y fecha. El trabajo del administrador del proyecto es asegurarse de que el proyecto de software cumpla y supere tales restricciones, además de que entregue software de alta calidad (p. 594).

De esta manera, al implementar metodologías de gestión de proyectos para el desarrollo de software, las empresas tendrían una herramienta que permita el aumento en la posibilidad de entregar y poner en producción el software, con el presupuesto y el tiempo establecidos, sin dejar de lado la calidad del producto entregado.

Por otra parte, según mencionan Gray y Larson (2009): “Un proyecto es un esfuerzo complejo, no rutinario, limitado por el tiempo, el presupuesto, los recursos y las especificaciones de desempeño y que se diseña para cumplir las necesidades del cliente” (p. 5).

Existen diferentes estándares de nivel internacional para la administración de proyectos de cualquier disciplina, lo que incluye a los proyectos de software. Uno de los estándares que más se utilizan en la actualidad es la Guía de los Fundamentos para la Dirección de Proyectos, sexta edición (de ahora en adelante se hace referencia a las siglas PMBOK “Project Management Book of Knowledge”) de la empresa privada *Project Management Institute* (en adelante PMI). En esta guía se define cómo se administra un proyecto de cualquier disciplina, tomando en cuenta aspectos clave como el alcance, costo y tiempo.

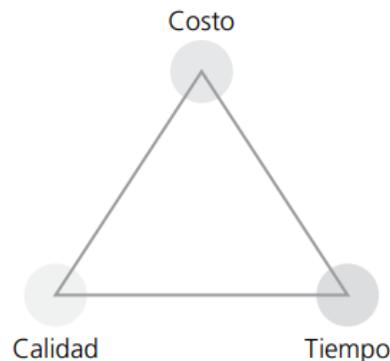
El objetivo principal de un proyecto es satisfacer la necesidad del cliente. Un proyecto tiene características que lo diferencian de otras tareas o procesos en la organización, por ejemplo, tiene un objetivo establecido, un inicio y fin y tiene requerimientos específicos de alcance, costo y desempeño (Gray y Larson, 2009).

De esta manera, es necesario que los proyectos de software de las mypes cuenten con metodologías con base en estándares internacionales, como PMBOK para que el proyecto se concluya en el alcance, costo, tiempo y recursos establecidos. Además, que impacte la calidad que se brinda a los usuarios finales.

Con base en lo anterior, la administración de proyectos tiene varios pilares principales: la administración del alcance, costo, tiempo y calidad. Respecto a la definición del alcance Gray y Larson (2009) mencionan que: “El alcance describe lo que usted espera entregarle a su cliente cuando termine el proyecto. Su enfoque debe definir los resultados a obtener en términos específicos, tangibles y que puedan ser medidos” (p. 86).

Figura 1

Factores de éxito en la administración de proyectos



Nota. Tomado de Administración de Proyectos, por López y Lankenau, 2017.

Debido a lo anterior es que desde el inicio del proyecto de software debe establecerse de manera estandarizada y en coordinación con el cliente y patrocinadores qué es lo que se desea realizar, el producto de software como tal, entregables, cronogramas y otros.

Asimismo, el manejo de costos del proyecto es otro aspecto que debe tomarse en cuenta en la administración de proyectos de software. López y Lankenau (2017) aseveran que: “La gestión del presupuesto del proyecto consiste en estimar, detallar y, posteriormente, controlar los costos que se originan con la utilización de recursos humanos y materiales” (p. 5).

Por lo tanto, en la administración de proyectos de software se debe definir una forma de establecer los presupuestos que se necesitarán para que el proyecto se desarrolle. Para esto se deben tomar en cuenta los costos que involucran los recursos humanos (horas hombre), materiales y costos fijos.

El factor tiempo se refiere a la duración de las actividades del proyecto y se mide en días, meses o años (López y Lankenau, 2017). Los proyectos de software no se eximen de esta realidad, debido a que es necesario estimar la duración del proyecto para presupuestar los recursos necesarios. En la realidad estimar el tiempo de un desarrollo es una tarea difícil, generalmente ocurren retrasos, sin embargo, gracias a las técnicas que se plantearon en diferentes estándares, se pueden tener estimaciones más certeras.

Por otra parte, la gestión de la calidad en proyectos de software desempeña un papel trascendental, tanto en la calidad del proceso como en la calidad del producto, mediante el uso de estándares internacionales como el ISO 9000 que es una familia de normas internacionales para la administración y el aseguramiento de la calidad. De esta forma, estos estándares abarcan el diseño, la búsqueda, el aseguramiento de la calidad y los procesos de entrega, desde servicios bancarios hasta manufactura. La administración y el mejoramiento de la calidad implican, de manera invariable, administración de proyectos (Gray y Larson, 2009).

Debido a lo anterior, es necesario que las empresas que desarrollan software cuenten con procesos de aseguramiento de la calidad en el proceso de desarrollo. Estos procesos deben definirse en el proceso de administración de proyectos de la empresa. Schulmeyer (2008) menciona que:

La relación entre la calidad de un producto y la organización responsable del desarrollo de ese producto es multidimensional. La relación depende de muchos factores, como la estrategia comercial y la estructura comercial de la organización, el talento disponible y los recursos necesarios para producir el producto (p. 1).

Por otra parte, la falta de una adecuada planificación y administración de proyectos puede repercutir en la calidad del proyecto, o bien en su fracaso. Gray y Larson (2009) mencionan que: “La carencia de un plan de proyecto definido con nitidez surge de manera constante como la principal razón para el fracaso de los proyectos” (p. 103). Carranza (2016) menciona que:

Uno de los factores que pueden ser decisivos para el fracaso de los proyectos es la falta de apropiación y conocimiento de las causas que pueda abarcar la mala toma de decisiones de un gerente de proyecto y

que este no aplique las mejores prácticas y no aprenda de las lecciones en pasados proyectos (p. 18).

Con base en lo mencionado, se plantea el siguiente problema de investigación:

¿De qué manera gestionan sus proyectos de software las mypes de los tres cantones con mayor cantidad de empresas para que, mediante la propuesta de una guía de administración de proyectos, les permita alinearse a estándares internacionales y así estandarizar su proceso?

De esta forma, el propósito de la investigación es **proponer una guía de administración de proyectos de desarrollo de software orientada a mypes que desarrollan software con base en estándares internacionales.**

1.3 Pregunta problema

Las empresas, sin importar su tipo de actividad comercial, gestionan sus proyectos utilizando diferentes metodologías y estándares internacionales que permitan definir con claridad el alcance, costo, tiempo y otros. En el caso de las mypes que desarrollan software en el país, el implementar una metodología de administración de sus proyectos permite tener una visión clara de las distintas áreas que se tienen que gestionar en el proceso de desarrollo de software y cuáles artefactos generar en cada una de las etapas del proceso. La pregunta que guía la presente investigación es la siguiente:

¿De qué manera gestionan sus proyectos de software las mypes de los tres cantones con mayor cantidad de empresas en la Gran Área Metropolitana para que, mediante la propuesta de una guía de administración de proyectos, les permita alinearse a estándares internacionales y así estandarizar su proceso?

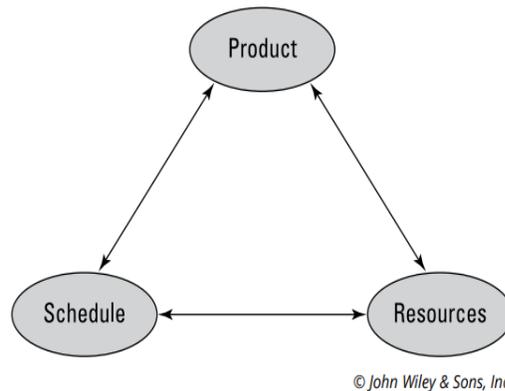
1.4 Hipótesis

Es importante que cada organización que desarrolle productos de software cuente con una metodología definida que permita administrar de manera eficiente sus proyectos de software, tanto el proceso como el producto, cumpliendo con el alcance, costo y tiempo establecidos al inicio del proyecto. Esta metodología de gestión de proyectos de software se pretende que esté basada en estándares para la administración de proyectos como la expuesta en la guía PMBOK del PMI, además del uso de metodologías ágiles que permitan entregar software de calidad, con el menor costo y tiempo posible.

Se busca que, al implementar una propuesta para la administración de proyectos de software en las mypes costarricenses, se establezca un proceso estructurado de desarrollo de software que permita la gestión eficiente durante todo el ciclo de vida del proyecto de software. Esto incluye las tres grandes áreas de los proyectos: el alcance, costo y tiempo.

Figura 2

Ciclo de vida de un proyecto



Nota. Project Management for Dummies, por Portny y Portny, 2022.

Con lo mencionado, la hipótesis que guía la presente investigación es la siguiente:

La información obtenida sobre el proceso de administración de proyectos que se utiliza en las pequeñas y medianas empresas de los tres cantones con mayor cantidad de compañías en la Gran Área Metropolitana permite la elaboración de una guía para gestionar sus proyectos con base principalmente en estándares internacionales.

1.5 Objetivos

1.5.1 Objetivo general

- Proponer una guía de administración de proyectos para las mypes desarrolladoras de software de los tres cantones que concentran la mayor cantidad de empresas en la Gran Área Metropolitana utilizando estándares internacionales que permitan la estandarización de este proceso en este tipo de organizaciones.

1.5.2 Objetivos específicos

- Conocer la forma en la que se gestionan los proyectos de software en las mypes desarrolladoras de software de los tres cantones que

concentran la mayor cantidad de empresas en la Gran Área Metropolitana mediante consultas a las partes interesadas.

- Identificar los estándares que más utilizan las empresas durante su proceso de administración de proyectos de software mediante la recopilación de datos.
- Determinar el estándar de administración de proyectos más adecuado para este tipo de empresas mediante el análisis de la información que se recopiló.
- Diseñar la guía de administración de proyectos de software que requiere este tipo de organizaciones, mediante los estándares internacionales consultados.

1.6 Alcance y limitaciones

1.6.1 Alcance

La presente investigación pretende ser de utilidad para las mypes desarrolladoras de software ubicadas en los tres cantones con mayor concentración de empresas de la Gran Área Metropolitana de Costa Rica. Lo anterior mediante la propuesta de una guía de administración de proyectos de software que les permitan administrar sus proyectos de acuerdo con lo establecido en estándares internacionales.

De esta manera, se busca que, aunque la empresa no tenga administradores de los proyectos en planilla, se cuente con una guía que contemple la administración de los tres pilares de un proyecto: alcance, costo, tiempo, sin dejar de lado el control de calidad. De esta forma, se pretende que todo el equipo de desarrollo interiorice el proceso de gestión del proyecto y pueda ser parte de este.

Con base en lo anterior, esta investigación no se enfoca en metodologías de desarrollo de software, sino en metodologías de gestión de proyectos aplicadas al desarrollo de software para empresas mypes.

1.6.2 Limitaciones

Entre las limitaciones se prevé la falta de interés o disponibilidad de las empresas para aplicar los instrumentos de recolección de datos o tener acceso a información que pueda catalogarse como confidencial. Además, se prevé que, para obtener información de las empresas, se deban realizar procesos administrativos que impacten negativamente el tiempo de la investigación.

1.7 Matriz de congruencia

Tabla 2

Matriz de congruencia

Tema	Título	Objetivo general	Objetivos específicos	Problemática	Hipótesis
Administración de proyectos de <i>software</i>	Propuesta de una guía de administración de proyectos basada en estándares internacionales para micro y pequeñas empresas desarrolladoras de <i>software</i> de los tres cantones con mayor cantidad de compañías en la Gran	Proponer una guía de administración de proyectos para las <i>mypes</i> desarrolladoras de <i>software</i> de los tres cantones que concentran la mayor cantidad de empresas en la Gran Área Metropolitana utilizando estándares internacionales que permitan la estandarización de	<ul style="list-style-type: none"> Conocer la forma en la que se gestionan los proyectos de <i>software</i> en las <i>mypes</i> desarrolladoras de los tres cantones que concentran la mayor cantidad de empresas en la 	¿De qué manera gestionan sus proyectos de <i>software</i> las <i>mypes</i> de los tres cantones con mayor cantidad de empresas en la Gran Área Metropolitana para que, mediante la propuesta de una guía de administración de proyectos, les permita alinearse a estándares	La información obtenida sobre el proceso de administración de proyectos que se utiliza en las <i>mypes</i> de la Gran Área Metropolitana de Costa Rica permite la elaboración de una guía que permita gestionar sus proyectos de acuerdo con lo establecido en

	<p>Área Metropolitana de Costa Rica</p>	<p>este proceso en este tipo de organizaciones.</p>	<p>Gran Área Metropolitana mediante consultas a las partes interesadas.</p> <ul style="list-style-type: none"> ● Identificar los estándares que más utilizan las empresas durante su proceso de administración de proyectos de <i>software</i> mediante la recopilación de datos. ● Determinar el estándar de administración 	<p>internacionales y así estandarizar su proceso?</p>	<p>los estándares internacionales.</p>
--	---	---	--	---	--

			<p>de proyectos más adecuado para este tipo de empresas mediante el análisis de la información que se recopiló.</p> <ul style="list-style-type: none">● Diseñar la guía de administración de proyectos de <i>software</i> que requiere este tipo de organizaciones, mediante los estándares internacionales consultados.		
--	--	--	--	--	--

CAPÍTULO II:
MARCO TEÓRICO

2 Marco teórico

2.1 Micro, pequeñas y medianas empresas en Costa Rica

En la actualidad, existen organizaciones empresariales que por su tamaño y recursos se denominan micro, pequeñas y medianas empresas, cuyo acrónimo se conoce como pymes. Iavarone (2012) se refiere al concepto de pymes:

Se consideran pequeñas empresas aquellas que tienen menos de 20 trabajadores y medianas las que tienen entre 20 y 500 empleados. Esta definición es susceptible de variar en función de los distintos contextos económicos e históricos, no existe una definición única que categorice a la pequeña y mediana empresa dado que se utilizan diversos criterios (p. 10).

Debido a lo anterior es que cada país establece su propia definición de pymes, de acuerdo con los criterios definidos por su legislación. Según menciona Filion et al. (2011):

Estas empresas contribuyen en gran medida al producto interno bruto (PIB) de la mayoría de los países del mundo. Además, a ellas se atribuyen el desarrollo económico de las naciones, la creación constante de empleos (el 80% de los nuevos puestos) y la generación de innovaciones (p. 13).

De esta manera, las pymes desempeñan un papel importante en el crecimiento y desarrollo de cualquier país. A esto se debe que los gobiernos implementan acciones para fortalecer estas empresas, ya que este tipo de compañías fortalecen al sector productivo nacional.

Por otra parte, Cousin (2020) explica que: “Las PYMES son la primera forma de surgimiento de los negocios, siendo su ciclo de desarrollo la estructuración de

una pequeña empresa que crece hasta convertirse en una empresa grande, son pocos los casos que logran tal crecimiento” (p. 4).

Debido a lo anterior es que la mayoría de las empresas en el momento de su constitución comienzan con la categoría pymes y conforme evolucionan en sus recursos y su capital se consideran como empresas grandes. A esto se debe que es importante que los gobiernos implementen políticas a favor de este tipo de empresas. Iavarone (2012) menciona que:

Las pequeñas y medianas empresas (PYMES) desempeñan un papel trascendental en la industria nacional ya que son las empresas con mayor capacidad de creación de empleo, realizan un alto porcentaje de las actividades manufactureras, y constituyen una base para la expansión de las industrias (p. 7).

Por otra parte, en Costa Rica para que una empresa se catalogue como pymes debe contar con diferentes parámetros, uno de ellos lo determina la Ley n.º 8262 Reglamento a la Ley de Fortalecimiento de las Pequeñas y Medianas Empresas. Esta ley menciona que el Ministerio de Economía, Industria y Comercio (en adelante MEIC) otorgará la categoría PYMES a una empresa que cumpla con algunos criterios, tanto cualitativos como cuantitativos, entre ellos:

1. La actividad productiva que realiza la empresa debe estar prevista en los códigos de las clases de la Clasificación Industrial Internacional Uniforme de todas las Actividades Económicas de las Naciones Unidas (en adelante CIIU).
2. Debe estar constituida bajo personería física o jurídica.
3. Que la empresa esté dentro del tamaño establecido en el reglamento.
4. Que cumplan con el pago de las cargas sociales y todas las obligaciones laborales.

Las actividades productivas del CIIU están divididas en 21 secciones, que van desde la agricultura y ganadería hasta actividades de organizaciones y órganos extraterritoriales. Para efectos de la presente investigación, las

actividades comerciales de las pymes se catalogan como **comerciales y de servicios**.

De la misma forma, según la Cámara de Comercio de Costa Rica (s. f.), la CCSS establece que una empresa entra en la categoría micro si posee de 1 a 5 trabajadores, pequeñas de 6 a 30, medianas de 31 a 100 y grandes si poseen más de 100 empleados. De esta manera, para que en Costa Rica una empresa sea catalogada como pymes debe cumplir con las disposiciones de la Ley n.º 8262, establecidos en el art. 14. Por otra parte, Cousin (2020) menciona que:

Según estadísticas de la Caja Costarricense de Seguro Social para el año 2018, un 74% de las PYMES (sic) eran empresas de tipo micro. Es decir, una gran mayoría de estas empresas poseen un número limitado de trabajadores, quizás menos de cinco, y disponen de pocos recursos financieros para invertir en tecnología e investigación (p. 28).

Debido a lo anterior es que las pymes son un motor importante en la economía de cualquier país. Según el informe Estado de situación pymes en Costa Rica 2021 del Ministerio de Economía, Industria y Comercio, en el 2019 existía un total de 113 845 empresas catalogadas como pymes, como se detalla en la Tabla 3:

Tabla 3

Cantidad de pymes por tamaño del año 2019

Tamaño de la empresa	Año 2019
Micro	110 973
Pequeña	17 237
Mediana	5635
Total	133 845

Nota. Tomado de Ministerio de Economía, Industria y Comercio, 2021.

De la misma manera, este informe menciona que la cantidad de empresas grandes corresponden a 3230. Debido a lo anterior es que este tipo de compañías corresponde al 96 % del parque empresarial (Ministerio de Economía, Industria y Comercio, 2021).

2.1.2 Principales sectores económicos

Según el informe Estado de situación pymes en Costa Rica 2021 del MEIC los principales sectores económicos de este tipo de empresas corresponden al sector comercial y de servicios. La cantidad de empresas del sector comercial en términos absolutos corresponde a 29 045 y a 21.1 % en términos relativos.

Por otra parte, las pymes cuya actividad económica corresponde al sector servicios son 90 637, un 66 % en términos relativos (MEIC, 2021). Otras empresas que corresponden a otros sectores económicos como el industrial y el agropecuario tienen un porcentaje menor con 5 % y 6.2 %. De esta manera, en Costa Rica existen sectores económicos dominantes: el sector comercial y de servicios, los cuales se detallan a continuación.

2.1.2.1 Comercial

Según menciona Fernández (2013): “Las empresas comerciales, a diferencia de las empresas industriales, adquieren productos terminados (productos comerciales) para su venta en diversos mercados sin transformación” (p. 4). Es decir, las empresas de actividad comercial son aquellas que venden productos tangibles al usuario y de ahí proviene su principal fuente de ingresos.

En el ámbito nacional, las pymes que corresponden al sector comercial están distribuidas de la siguiente manera, de acuerdo con el tamaño de la empresa:

Tabla 4

porcentaje de pymes del sector comercial en el año 2019

Tamaño de la empresa	Porcentaje
Micro	71,9
Pequeña	18,4
Mediana	6,3
Grande	3,4

Nota. Tomado de Ministerio de Economía, Industria y Comercio, 2021.

De esta manera, la mayor cantidad de empresas del sector comercial corresponde a las microempresas. Además, el sector comercio es el segundo sector que genera más empleos con un total de 82.416. Por otra parte, uno de los sectores comerciales más importantes en el ámbito nacional es el de servicios, donde se encuentra la categoría de programación informática.

2.1.2.2 De servicios

Sobre las empresas que pertenecen al sector servicios González (1989) menciona lo siguiente:

Las actividades de los servicios que pertenecen al sector terciario se suelen definir en un sentido muy general como «las actividades que no producen bienes». Entre ellas se encuentran la distribución, el transporte y las comunicaciones, las instituciones financieras y los servicios a las empresas y los servicios sociales y personales (p. 12).

Debido a lo anterior, el objetivo de las empresas del sector servicios es ofrecer servicios en lugar de productos, es decir, su capital proviene de productos intangibles en lugar de tangibles, muchas de ellas aplicando el conocimiento y la innovación.

Adicionalmente, en el informe Estado de situación pymes en Costa Rica 2021 del MEIC se menciona que solo las empresas del sector de servicios han

mantenido una tasa de crecimiento positiva durante los últimos años, con un crecimiento del 5.2 % (MEIC, 2021). Además, el sector servicios es el que genera la mayor cantidad de empleos (un 57.6 %). En el ámbito nacional, las pymes que corresponden al sector servicios se encuentran distribuidas de la siguiente forma:

Tabla 5

Porcentaje de pymes del sector comercial en el año 2019

Tamaño	Porcentaje
Micro	72 %
Pequeña	23 %
Mediana	4 %
Grande	1 %

Nota. Tomado de Ministerio de Economía, Industria y Comercio, 2021.

2.1.3 Desarrolladoras de software

Las empresas desarrolladoras de software están catalogadas como empresas de servicios, según la clasificación de actividades económicas CIIU. En esta clasificación las empresas desarrolladoras de software están catalogadas con el nombre de actividades de programación informática. En el ámbito nacional se encuentran registradas un total de 219 micro y pequeñas empresas (mypes) que se dedican a este tipo de actividad, de acuerdo con registros del MEIC.

Una de las características de las pymes dedicadas a las tecnologías de información (IT) es que, en ocasiones, se encuentran limitadas en sus recursos, tanto humanos como materiales. Laporte et al. (2008) mencionan que: “Existen diferencias significativas, en términos de recursos e infraestructura disponibles, entre una PYMES que emplea de 1 a 10 personas y un departamento de Tecnología de la Información (TI) del mismo tamaño en una empresa más grande” (p. 5).

Al estar limitadas en sus recursos, las empresas desarrolladoras pueden carecer de personal capacitado para gestionar el desarrollo de software en todas

sus etapas mediante el uso de modelos estandarizados en el ámbito internacional de gestión de proyectos, calidad y gestión de la configuración. Sobre lo anterior, Larrucea *et al.* (2015, citados por Rodríguez-Dapena y Lohier, 2017) mencionan lo siguiente:

Para muchas PYMES, es un gran desafío implementar controles y estructuras para administrar adecuadamente sus procesos de software (LARRUCEA *et al.*, 2016), y la falta de formalismo en sus procesos puede tener consecuencias negativas, como perder actividades y tareas importantes, o teniendo formas limitadas de demostrar su calidad y ser reconocidos en su dominio, en consecuencia, pueden ser apartados de los proyectos (p. 35).

De esta manera, para que las mypes desarrolladoras produzcan software de calidad, se requiere que estas implementen metodologías para estandarizar sus procesos, con base en estándares internacionales. Estayno *et al.* (2009) mencionan que:

La aplicación de modelos de calidad favorece a la mejora continua, establece procesos estándares con insumos y resultados medibles, reduce costos y promueve la eficiencia. Las empresas se ven beneficiadas al poder ofrecer a sus clientes productos de mayor calidad y seguridad en el cumplimiento de los tiempos previstos (p. 2).

Existen diferentes modelos de gestión para aplicarlos al proceso de desarrollo de software. Estos estándares abarcan diversas áreas de la gestión del software, entre las que se destacan las siguientes:

- PMI: contiene una metodología para la gestión de proyectos de cualquier disciplina, incluido el desarrollo de software.

- ISO 9001 para la gestión de la calidad de cualquier producto. El ISO 9001-3 contiene la adaptación de la norma 9001 al desarrollo de software.
- IEEE: contiene un estándar para manejar el software en todo su ciclo de vida, desde una perspectiva administrativa donde se abarcan las áreas que van desde la fase de adquisición hasta la fase de operación y mantenimiento.
- Cobit: es un marco de trabajo que proporciona una serie de recomendaciones para que una organización sea eficiente.
- CMMI: modelo orientado a la mejora de procesos de desarrollo de software.

Estos modelos se aplican al proceso de ingeniería del software que posee un ciclo de vida, el cual está representado de manera diferente de acuerdo con la metodología que se utiliza. El ciclo de vida del software y sus metodologías se detallan a continuación:

2.2 Ingeniería de Software

Según Pressman (2011): “La ingeniería de software es una disciplina de la ingeniería que se interesa por todos los aspectos de la producción de software” (p. 6). El proceso de desarrollo de software se compone de varias etapas que comienzan desde que se definen sus requerimientos hasta su mantenimiento y operación.

Cada una de estas etapas varía según el proceso que se utilice, pero en general el proceso de desarrollo se compone de varias etapas comunes que se mencionan más adelante. Pressman (2011) menciona que:

La ingeniería de software busca apoyar el desarrollo de software profesional, en lugar de la programación individual. Incluye técnicas que apoyan la especificación, el diseño y la evolución del programa, ninguno

de los cuales son normalmente relevantes para el desarrollo de software personal (p. 5).

De esta manera, la ingeniería de software es una disciplina aplicada en el desarrollo de software profesional que tiene como resultado un producto final, donde en cada una de sus etapas se encuentran procesos definidos. Existen métodos y herramientas que validan los procesos que se realizan en cada una de las etapas, sin embargo, los que se utilizan varían de una organización a otra y de un proyecto a otro. El objetivo principal de esta ingeniería es que el producto final de software se entregue con calidad, dentro de la fecha y el presupuesto establecidos. Por otra parte, Bruegge y Dutoit (2002) mencionan que:

La ingeniería de software es una actividad para la solución de problemas.

Se usan los modelos para buscar una solución aceptable. Esta búsqueda es conducida por la experimentación. Los ingenieros de software no tienen recursos infinitos, y están restringidos por presupuestos y tiempos de entrega (p. 27).

Debido a lo anterior es que la ingeniería del software es un enfoque que busca el desarrollo de un producto de software mediante el uso de métodos y herramientas (metodologías) que aseguren su calidad, con los recursos establecidos. Algunas de estas metodologías se detallan más adelante.

Existe una diferencia entre el proceso y el producto de software. El proceso de software corresponde a la metodología o a la serie de actividades que se realizan para construir un producto de software y el producto de software se entrega a partir de un proceso, independientemente de la metodología que se utiliza. García et al. (2017) mencionan que:

La calidad del producto desarrollado por una organización depende, necesariamente, de la calidad del proceso que se sigue para obtenerlo [...] es posible pensar que los procesos de software deben soportar o facilitar el cumplimiento de las características de calidad que satisfacen los

estándares requeridos para los productos de software desarrollados (p. 67).

Con base en el anterior postulado existen muchos procesos diferentes en el desarrollo de software, cada proceso varía según el tipo de proyecto y de la organización. En general, Pressman (2011) propone cuatro actividades principales, las cuales se dividen en varios subprocesos:

- Especificación del software: Se define la funcionalidad del software, es decir, los requerimientos y alcances.
- Diseño e implementación del software: Es donde se especifica el modelo de arquitectura por utilizar y diseño de los componentes. Una vez definida la arquitectura y diseñado el software, se procede con su desarrollo.
- Validación del software: Se valida el software para asegurarse de que cumpla con los requerimientos y corresponde a la etapa de pruebas.
- Evolución del software: Una vez implementado el software se pueden necesitar nuevos requerimientos para cumplir con las necesidades de los clientes.

De esta manera, en el ámbito general el proceso de desarrollo de software pasa por esta serie de actividades o etapas. Como se mencionó, cada una de estas actividades tiene una serie de subactividades, cuyo nombre varía en las diferentes metodologías. Al final de cada etapa se genera un resultado con distintos entregables, generalmente documentos y código fuente. A este conjunto de etapas se le conoce como ciclo de vida del software.

2.2.1 Ciclo de vida del software

Gómez Palomo y Moradela Gil (2020) se refieren al concepto de ciclo de vida:

El ciclo de vida es algo propio de un ser vivo; nace, crece o cambia, se reproduce o diversifica y muere o desaparece. El estudio de cualquier

concepto productivo, desde una óptica del ciclo de vida, lleva implícito una evolución temporal que debe ser tomada en cuenta en todos los momentos del ciclo. Cuando un ser vivo nace, igualmente lleva implícitas cualquiera de las fases posteriores por las que va a pasar su vida (p. 33).

Con base en el siguiente postulado, todo producto que se comercialice posee un ciclo de vida, desde que surge hasta que muere. El software también pasa por un ciclo de vida con diferentes etapas, desde que se determinan los requerimientos hasta que entra en fase de operación y mantenimiento. Como se mencionó, las etapas del proceso del ciclo de vida de un software varían de una metodología a otra. Por otra parte, Beng *et al.* (2012) mencionan que:

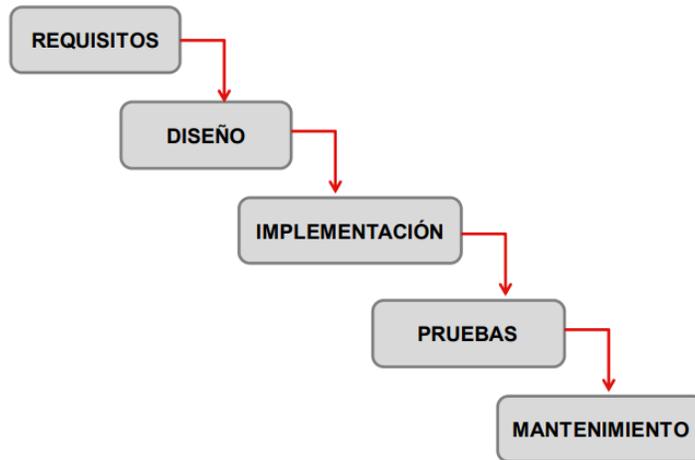
El ciclo de vida de desarrollo de software (SDLC) es un proceso de construcción o mantenimiento de sistemas de software. Por lo general, incluye varias fases, desde el análisis de desarrollo preliminar hasta el software de desarrollo posterior. prueba y evaluación (p. 1).

De esta manera, existen muchos tipos de modelos que indican cómo manejar el ciclo de vida de un software. Cada modelo varía en sus etapas y en cómo se debe manejar cada una de estas, sin embargo, en el ámbito general y sin apearse a una metodología en específico, el ciclo de vida de un software posee las siguientes etapas:

- Requerimientos.
- Diseño.
- Codificación e implementación.
- Pruebas.
- Mantenimiento y evolución.

Figura 3

Fases del ciclo de vida del software



Nota. Tomado de Ingeniería del Software: Metodologías y ciclos de vida, por Inteco, 2009.

A continuación, se detalla cada una de estas etapas en profundidad:

2.2.1.1 Requerimientos

Antes de desarrollar un proyecto de software es necesario comprender lo que desea el cliente. Esta comprensión de las necesidades del proyecto es lo que se conoce como requerimientos. La definición que proporciona la IEEE (1998) sobre la palabra requerimiento es la siguiente: “Una condición o capacidad que debe estar presente en un sistema o componentes de sistema para satisfacer un contrato, estándar, especificación u otro documento formal” (p. 62).

Un requerimiento tiene que especificarse por escrito, debe ser posible de probar o verificar, así como ser conciso, completo, consistente y no ambiguo. Es decir, que posea una sola interpretación (Chaves, 2006). Además, el requerimiento debe estar redactado en un lenguaje natural.

Los requerimientos generalmente se determinan al inicio del proyecto y estos pueden cambiar en el futuro, a esto se debe que la metodología de desarrollo que se utilice sea adaptable a los cambios. Existe una disciplina dentro de la ingeniería del software que se encarga de gestionar la etapa de la definición de los requerimientos, la cual se conoce como **ingeniería de requerimientos**. Según Pressman (2011):

La ingeniería de requerimientos proporciona el mecanismo apropiado para entender lo que desea el cliente, analizar las necesidades, evaluar la factibilidad, negociar una solución razonable, especificar la solución sin ambigüedades, validar la especificación y administrar los requerimientos a medida de que [sic] se transforman en un sistema funcional (p. 102).

De esta manera, esta disciplina busca determinar y clarificar los deseos del cliente para buscar una solución adecuada al problema que se planteó, además de determinar si el desarrollo del proyecto es factible. Debido a lo anterior es que los requerimientos deben definirse de la forma más clara posible.

Arias Chaves (2005) menciona que la principal tarea de la ingeniería de requerimientos es la:

Generación de especificaciones correctas que describan con claridad, sin ambigüedades, en forma consistente y compacta, las necesidades de los usuarios o clientes; de esta manera, se pretende minimizar los problemas relacionados por la mala gestión de los requerimientos en el desarrollo de sistemas (p. 2).

Por otra parte, existen dos tipos de requerimientos, los funcionales y los no funcionales. Los funcionales consisten en lo que el sistema debe realizar, según el tipo de software que se desarrolla. Además, varían desde requerimientos generales que indican lo que debe hacer el sistema hasta específicos que reflejan maneras locales de trabajar o los sistemas existentes de una organización (Sommerville, 2011). En contraste, sobre los requerimientos no funcionales, Chaves (2006) menciona que:

Tienen que ver con características que de una u otra forma puedan limitar el sistema, como, por ejemplo, el rendimiento (en tiempo y espacio), interfaces de usuario, fiabilidad (robustez del sistema, disponibilidad de equipo), mantenimiento, seguridad, portabilidad, estándares y etc. (p. 3).

De esta manera, un requerimiento funcional puede producir un requerimiento no funcional de forma implícita. Por ejemplo, si se solicita que un software posea un mecanismo de autenticación y autorización la funcionalidad misma es el requerimiento funcional, mientras que las características de seguridad que debe incluir ese mecanismo corresponden a requerimientos no funcionales.

Por otra parte, la ingeniería de requerimientos es importante para mejorar la capacidad de predecir cronogramas de los proyectos, estimación de tiempos y recursos necesarios. Además, disminuye los costos y retrasos del proyecto, mejora la calidad del software y evita los rechazos de los usuarios finales (Chaves, 2006).

Adicionalmente, Pressman (2011) menciona que existen siete tareas que se realizan en la ingeniería de requerimientos, las cuales son:

- **Concepción:** Es cuando se identifica la necesidad de negocio. Las personas participantes definen una idea de negocio y hacen un análisis de factibilidad.
- **Indagación:** Se pregunta al cliente y a los usuarios cuáles son los objetivos del sistema o producto. Se determina lo que se desea lograr y cómo se utiliza el sistema en la cotidianidad. Para lo anterior se utilizan diferentes técnicas como cuestionarios, entrevistas, reuniones, entre otros.
- **Elaboración:** La información obtenida durante las etapas de concepción e indagación se depura en esta etapa. En esta etapa se elabora un modelo detallado de requerimientos y se producen artefactos como los diagramas de casos de uso.
- **Negociación:** Los conflictos que pueden darse durante la etapa de requerimientos se reconcilian mediante un proceso de negociación. Se solicita que los clientes ordenen los requerimientos según la prioridad, de acuerdo con la prioridad se evalúa el costo y el riesgo.
- **Especificación:** se elabora un documento escrito mediante una plantilla estándar llamada Especificación de requerimientos que contiene todos los aspectos del software que se desarrolla.

- Validación: En esta etapa el equipo de desarrollo realiza una revisión técnica que busca encontrar inconsistencias, omisiones y errores en los requerimientos.
- Administración de requerimientos: Esta fase está diseñada para identificar, controlar y monitorear los requisitos y sus cambios en cualquier momento durante el desarrollo del proyecto.

Como conclusión, la primera etapa del ciclo de vida de desarrollo de un software comienza por la definición de los requerimientos, también llamada ingeniería de requerimientos, la cual es la etapa que antecede a la de diseño.

2.2.1.2 Diseño

Una vez definidos los requerimientos mediante las técnicas anteriores, la siguiente etapa en el proceso de desarrollo es la de diseño. Pressman (2011) detalla lo siguiente:

El diseño crea una representación o modelo del software, pero, a diferencia del modelo de los requerimientos (que se centra en describir los datos que se necesitan, la función y el comportamiento), el modelo de diseño proporciona detalles sobre arquitectura del software, estructuras de datos, interfaces y componentes que se necesitan para implementar el sistema (p. 183).

De esta manera, el diseño es la etapa donde se diseña el sistema. Es decir, se define el patrón de arquitectura, se diseñan los componentes del sistema, se definen los aspectos del despliegue, entre otros. Para lograr lo anterior, se producen diagramas que ofrezcan una representación visual a los ingenieros de software, estos diagramas se construyen utilizando lenguajes de modelado, el más conocido es Unified Modeling Language (en adelante UML).

Una vez diseñado el sistema se produce el modelo que es evaluado por los distintos participantes. Sommerville (2011) menciona que:

Un modelo es una abstracción del sistema a estudiar, y no una representación alternativa de dicho sistema. De manera ideal, una representación de un sistema debe mantener toda la información sobre la entidad a representar. Una abstracción simplifica y recoge deliberadamente las características más destacadas (p. 119).

Por otra parte, en el ámbito general los diagramas que se pueden producir durante la etapa de diseño son los siguientes de acuerdo con Sommerville (2011):

- Diagramas de actividad: muestra las actividades de un proceso del sistema.
- Diagrama de caso de uso: muestra las interacciones del sistema y el entorno que lo rodea.
- Diagrama de secuencias: muestra las interacciones dadas por los actores y los componentes del sistema.
- Diagrama de clase: Muestra las clases de objetos que hay en el sistema y sus asociaciones.
- Diagrama de estado: Muestra la forma en la que reacciona el sistema ante eventos internos y externos.

Además, Sommerville (2011) propone una categorización de modelos que se pueden producir en esta etapa. Estos tipos de modelos son:

- Modelos de contexto: Se determina cada funcionalidad del sistema y sus fronteras (cómo interactúa con su entorno). El entorno pueden ser varios sistemas automatizados externos que generan datos para el sistema.
- Modelos de interacción: Se modelan las interacciones del usuario, además de las interacciones (comunicaciones) entre el sistema por desarrollar y otros sistemas. Este modelo ayuda a identificar los requerimientos del usuario, también identifica los problemas de comunicación que puedan

presentarse. En este modelo se generan los diagramas de secuencia y casos de uso.

- Modelos estructurales: Muestran la organización de un sistema en el ámbito de módulos y componentes. En este modelo se diseña la arquitectura del sistema y se utilizan los diagramas de paquete, de componentes, clases e implementación.
- Modelos de comportamiento: Se muestra lo que debe suceder cuando un sistema responde ante un estímulo de su entorno, sean datos que lleguen al sistema o eventos que se disparen. Los diagramas que se utilizan para realizar esta representación son los diagramas de secuencias y de actividad.

De esta manera, se puede afirmar que el diseño de un software corresponde a los planos que se realizan antes de construir una casa. Estos planos permiten que el ingeniero determine dónde va cada pieza, de acuerdo con los requerimientos iniciales. De la misma forma funciona en el desarrollo de software. La etapa de diseño permite construir los planos del sistema para que el equipo de desarrollo tenga una representación visual de los componentes antes de construirlos.

Un buen modelo de diseño siempre debe implementar todos los requisitos que se establecen en la especificación de requerimientos. Además, el diseño de software debe ser legible y comprensible para los programadores y para los que lo prueban. Por último, debe mostrar el panorama completo del software y abordar el dominio de datos, comportamiento y funciones desde el punto de vista de la implementación (Pressman, 2011).

Por otra parte, de acuerdo con Durango (2014) el proceso de diseño cuenta con dos actividades fundamentales, el diseño arquitectural y el diseño detallado:

- Diseño arquitectural: Describe la arquitectura del software, es decir, cómo el software es descompuesto y organizado en módulos y sus relaciones.
- Diseño detallado: Describe el comportamiento específico y en detalle de los módulos que componen el diseño arquitectural.

Por otra parte, Hewlett-Packard desarrolló una serie de atributos de calidad de software que deben tomarse en cuenta en el momento de diseñar el software, los cuales son funcionalidad, usabilidad, confiabilidad, rendimiento y mantenibilidad (Pressman, 2011). Estos atributos de calidad deben tomarse en cuenta en el proceso de diseño y no al final del desarrollo, donde es más complicado realizar cambios.

A modo de conclusión, el diseño es una etapa trascendental en el desarrollo del software, ya que es en esta donde se realiza el diseño del sistema que se desarrolla en la etapa de codificación; sin un diseño es difícil para los desarrolladores saber en dónde colocar cada componente y la manera en la que estos interactúan. Una vez finalizada esta etapa se procede con la codificación del sistema.

2.2.1.3 Codificación

Cerrada (2006) se refiere a la etapa de codificación de la siguiente forma: Se puede decir que las etapas previas de análisis y diseño tienen como misión fundamental la de organizar y traducir los requisitos del cliente en unidades o módulos de programa que puedan ser codificados de forma independiente y sencilla. La importancia de la fase de codificación dentro del desarrollo de software es evidente si se tiene en cuenta que en ella se elabora el producto fundamental de todo el desarrollo: los programas fuente (p. 233).

De esta manera, en la etapa de codificación es donde se crea el código fuente que dará vida a la aplicación, mediante un lenguaje de programación. Todo el software está constituido por líneas de código que le indican a la computadora qué es lo que debe realizar y de qué forma debe realizarlo; esta labor la realizan los programadores. De esto la importancia de lo dicho por Cerrada (2006):

En la fase de codificación pueden intervenir un gran número de programadores y durante un tiempo muy largo. Como parte de la metodología de programación y para garantizar la adecuada homogeneidad en la codificación se deben establecer las normas y estilo de codificación (p. 234).

Con base en el anterior postulado, estas normas y estándares las define cada organización, aunque determinados lenguajes también poseen sus propios estándares de nomenclatura. Es importante que el código fuente se apegue a estándares de codificación para facilitar que otros programadores lo comprendan.

El lenguaje de programación es un lenguaje formal diseñado para realizar procesos que pueden llevar a cabo máquinas como las computadoras. Además, puede usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión o como modo de comunicación humana (Olarte Gervacio, 2018).

Por otra parte, otro aspecto para tomar en cuenta en la etapa de codificación es la **gestión de la configuración**, Sommerville (2011) menciona que:

En el desarrollo de software, los cambios ocurren todo el tiempo, de modo que la administración del cambio es absolutamente esencial. Cuando un equipo de individuos desarrolla software, hay que cerciorarse de que los miembros del equipo no interfieran con el trabajo de los demás (p. 195).

La gestión de la configuración es el proceso de gestionar los cambios del software, de manera que los desarrolladores tengan acceso a la última versión del sistema. Además, se busca que los desarrolladores realicen cambios controlados al código fuente sin tener conflictos con el código de otro desarrollador, ya que ambos pueden trabajar sobre el mismo archivo de forma paralela. Por último, mediante una gestión correcta de la configuración es más fácil revertir los cambios en caso de que sea necesario.

En la actualidad, existen diferentes herramientas para gestionar los cambios del código, que permiten controlar las diferentes versiones de este. Además, existen distintos sistemas de control de versiones disponibles en el mercado, entre ellos Git, Mercurial, Monotone, CVS y otros.

Por otra parte, el lenguaje de programación que se utilice para programar también toma importancia. Cerrada (2006) menciona que:

El lenguaje de programación es probablemente uno de los elementos más importantes de cualquier desarrollo puesto que es la herramienta de trabajo que utilizarán mayor número de personas y que además tiene una influencia decisiva en la depuración y mantenimiento de la aplicación (p. 261).

De esta manera, existen diferentes criterios para seleccionar un lenguaje de programación, entre ellos existen factores operativos y técnicos que intervienen. Entre las principales se destacan las que menciona Cerrada (2006):

- Imposición del cliente: El cliente es quien fija el lenguaje de programación que se utiliza debido a diferentes criterios.
- Tipo de aplicación: El lenguaje se escoge de acuerdo con el tipo de aplicación. Existen lenguajes que funcionan mejor que otros según el tipo de aplicación que se desarrolle.
- Experiencia previa: Se toma en cuenta la experiencia del personal y se decide que el lenguaje de programación se escoja debido a que es el dominante dentro del equipo.
- Reusabilidad: Se toman en cuenta las librerías reutilizables que ofrece el lenguaje de programación para no escribir código desde cero.

Con base en todo lo que se planteó, la etapa de codificación es la que requiere de más esfuerzo por parte del equipo de trabajo y es la que más tiempo consume. En esta etapa se produce el producto que se entrega al cliente, por lo

tanto, en el proceso de codificación y en las etapas sucesoras del desarrollo debe asegurarse la calidad tanto del proceso como del producto.

2.2.1.4 Pruebas

Según Cerrada (2006):

A lo largo de la fase de codificación se introducen de manera inadvertida múltiples errores de todo tipo e incorrecciones respecto a las especificaciones del proyecto. Todo ello debe ser detectado y corregido antes de entregar al cliente el programa acabado (p. 272).

Para lograr lo anterior, se realizan pruebas de software que ayudan a detectar los errores del código a tiempo y, de esta manera, garantizar que se entregue un producto sin defectos. Uno de los objetivos de las pruebas de software es hacer que el software no funcione de forma correcta para encontrar sus defectos y después corregirlos y garantizar su calidad.

Para lograr la calidad del software, Pressman (2011) menciona que: “La calidad del software no sólo se ve. Es el resultado de la buena administración del proyecto y de una correcta práctica de la ingeniería de software” (p. 350). Pressman (2011) propone una serie de actividades que permiten lograr la calidad del software durante todas las etapas del ciclo de vida:

1. Métodos de ingeniería de software: Se deben utilizar métodos para comprender completamente el problema y diseñar un modelo de software.
2. Técnicas de administración de proyectos: Cuando se crea un plan de proyecto adecuado, la estimación correcta de tiempos, el seguimiento de las actividades programadas y la planeación del riesgo entonces la calidad influye de manera positiva.
3. Control de calidad: Se refiere al control de calidad empleado durante el proceso de desarrollo, lo que incluye el levantamiento de requisitos, la revisión de los modelos producidos en la etapa de diseño hasta la revisión del código forma parte de este proceso.

4. Aseguramiento de la calidad: Se encarga de evaluar el proceso de control de calidad mediante funciones de auditoría y reportes.

Por otra parte, para que un producto sea de calidad Pressman (2011) menciona que el estándar ISO 9126 indica los atributos clave que debe tener un software de calidad:

- Funcionalidad: Medida en que el software satisface los requerimientos que se plantearon.
- Confiabilidad: Cantidad de tiempo que se puede utilizar sin problemas.
- Usabilidad: Grado en el que un software es fácil de utilizar, que este sea entendible, aprehensible y operable.
- Eficiencia: Grado en que el software utiliza óptimamente los recursos del sistema.
- Facilidad de mantenimiento: Es la facilidad con la que se introducen cambios al software.
- Portabilidad: Es la facilidad con la que un software se traslada de un ambiente a otro.

Las pruebas permiten al desarrollador y al cliente demostrar que el software cumple con los requerimientos (Sommerville, 2011).

El proceso de **verificación y validación** son tareas realizadas en la fase de pruebas. La verificación consiste en comprobar que el software posee todas las funcionalidades especificadas en los requerimientos, mientras que la validación se encarga de garantizar que el software cumpla con las expectativas del cliente. Tanto para la validación como para la verificación existen diferentes tipos de pruebas, entre las cuales Sommerville (2011) menciona las siguientes:

- Pruebas de desarrollo: El sistema se prueba para encontrar errores que conduzcan a defectos.

- Versiones de prueba: El sistema lo prueba un equipo separado antes de presentarse a los usuarios; el objetivo es validar los requerimientos.
- Pruebas de usuario: El sistema lo prueban usuarios reales en su propio entorno, donde brindan sus recomendaciones.

Es importante mencionar que las pruebas se pueden realizar de manera manual o automática, por lo tanto, gracias a las pruebas se asegura la calidad del producto. Este proceso debe llevarse a cabo en cada etapa del desarrollo, una vez que se codificó el sistema, la siguiente etapa corresponde a su puesta en producción, mantenimiento y evolución.

2.2.1.5 Mantenimiento y evolución

Según Sommerville (2011):

El desarrollo del software no se detiene cuando un sistema se entrega, sino que continúa a lo largo de la vida de éste. Después de distribuir un sistema, inevitablemente debe modificarse, con la finalidad de mantenerlo útil. Tanto los cambios empresariales como los de las expectativas del usuario generan nuevos requerimientos para el software existente (p. 235).

Cuando un software se pone en producción pueden necesitarse cambios en el futuro, ya sean por nuevos requerimientos o para corregir una falla no detectada, ya que un software puede utilizarse durante años y conforme transcurre el tiempo el software se va degradando.

Con base en lo anterior, Sommerville (2011) menciona que: “Evidentemente, los requerimientos de los sistemas instalados cambian conforme lo hacen el negocio y su entorno. Por consiguiente, se crean a intervalos regulares nuevas versiones de los sistemas, las cuales incorporan cambios y actualizaciones” (p. 235).

En este caso, los cambios nuevos pueden introducirse por el mismo equipo que desarrolló la aplicación o por un equipo nuevo. Si el software lo desarrolló una compañía desarrolladora, es probable que la misma empresa brinde el mantenimiento necesario, o bien que los propios desarrolladores de la empresa se encarguen de este proceso.

Ocala (2012) sugiere que existen varios tipos de mantenimiento de software, entre los cuales se destacan:

- Evolución correctiva: Corresponde al proceso de evolución donde se reparan defectos que se encontraron en el software, ya sea en periodos de emergencia o bien en reparaciones planificadas.
- Evolución adaptativa: Corresponde al mejoramiento del software para implementar nuevas funcionalidades. Durante esta etapa se siguen las etapas del proceso de desarrollo: requerimientos, diseño, codificación y pruebas.
- Evolución perfectiva: Es un método que mejora la calidad del software y su documentación para que reciba mantenimiento de una mejor manera y así reducir el tiempo, costo e impacto de los futuros cambios.
- Evolución preventiva: Se encarga de prevenir los fallos antes de que sucedan, para lo que se realizan diferentes pruebas en el momento del desarrollo.

Por consiguiente, la metodología y los estándares con los que se desarrolle el software deben ser lo suficientemente adaptables para soportar cambios en el futuro. Esto se menciona a continuación.

2.2.2 Modelos de desarrollo

Para desarrollar un software se sigue un proceso, el cual posee una serie de etapas que están organizadas de diferente forma. A la manera en la que las etapas están organizadas entre sí se le llama **flujo de proceso**. La estructura general para desarrollar un software o ciclo de vida clásico se compone de las siguientes etapas:

- Requerimientos (comunicación)
- Diseño
- Construcción
- Despliegue, mantenimiento y evolución.

Pressman (2011) menciona que:

Un flujo de proceso lineal ejecuta cada una de las cinco actividades estructurales en secuencia, comenzando por la comunicación y terminando con el despliegue [...]. Un flujo de proceso iterativo repite una o más de las actividades antes de pasar a la siguiente [...]. Un flujo de proceso evolutivo realiza las actividades en forma “circular”. A través de las cinco actividades, cada circuito lleva a una versión más completa del software y un proceso paralelo ejecuta una o más actividades en paralelo con otras (p. 28).

Con base en lo anterior, los diferentes flujos de proceso de desarrollo permiten que cada una de las etapas se realice de una manera u otra. Cada flujo de proceso posee una serie de modelos definidos por varios autores a través de los años, por ejemplo, cascada, evolutivo, progresivo y otros.

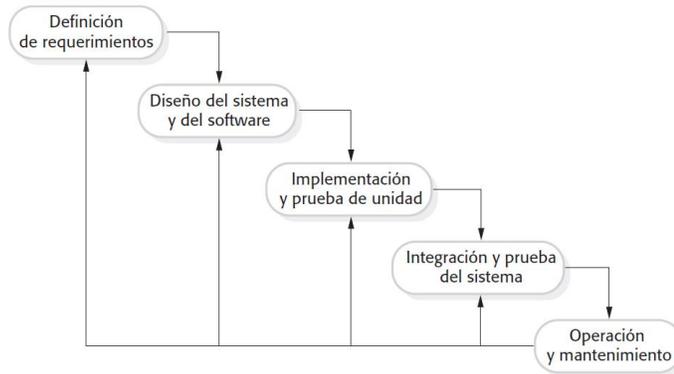
2.2.2.1 Cascada

El modelo en cascada se caracteriza porque sus pasos son secuenciales. Va hacia abajo a través de las fases de análisis de requerimientos, diseño, codificación, pruebas, despliegue y mantenimiento. Este modelo funciona bien para proyectos en los cuales el control de calidad es importante debido a su planificación y control de calidad intensivos (Alshamrani, 2015). Además, este

modelo es eficiente cuando los requerimientos están definidos y el proyecto no está susceptible de cambios constantes.

Figura 4

Ciclo de vida en cascada



Nota. Tomado de Ingeniería de Software, por Sommerville, 2011.

Otro de los objetivos del modelo en cascada es aislar cada fase de la siguiente, de manera que las fases sucesivas puedan hacerse por grupos de personas diferentes, lo que facilita la especialización. De esta forma, es posible encontrar distintos perfiles profesionales, como analista, diseñador, programador, etc. (Cerrada, 2006).

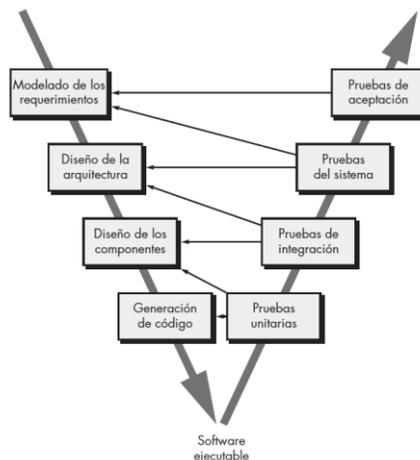
Por otra parte, este modelo posee una serie de desventajas importantes, por ejemplo, pocos proyectos son estrictamente lineales, no siempre los requerimientos se definen bien desde el inicio del proyecto. Además, el cliente debe esperar hasta fases muy avanzadas del desarrollo para brindar su retroalimentación, que puede ocasionar que el cliente se sienta disconforme y se deban realizar cambios profundos en el software, lo cual implica costos y retrasos adicionales al proyecto.

Barac (citado por Pressman, 2011) menciona que: “La naturaleza lineal del ciclo de vida clásico llega a “estados de bloqueo” en los que ciertos miembros del equipo de proyecto deben esperar a otros a fin de terminar tareas interdependientes” (p. 34).

Lo anterior significa que el equipo de trabajo tiene que invertir tiempo en espera, lo cual disminuye la productividad del equipo y, como consecuencia, aumenta los costos. Otra variante del modelo en cascada es el modelo en V, donde cada etapa del desarrollo está asociada con pruebas de verificación y validación. Pressman (2011) se refiere al modelo en V de la siguiente forma:

A medida que el equipo de software avanza hacia abajo desde el lado izquierdo de la V, los requerimientos básicos del problema mejoran hacia representaciones técnicas cada vez más detalladas del problema y de su solución. Una vez que se ha generado el código, el equipo sube por el lado derecho de la V, y en esencia ejecuta una serie de pruebas (acciones para asegurar la calidad) que validan cada uno de los modelos creados cuando el equipo fue hacia abajo por el lado izquierdo (p. 35).

Figura 5
Modelo en V



Nota. Tomado de Ingeniería del Software: Un enfoque práctico séptima edición, por Pressman, 2011.

Por otra parte, al final de cada fase se produce una serie de artefactos para que otras personas trabajen la siguiente etapa del proceso. Cerrada (2006) menciona algunos de ellos:

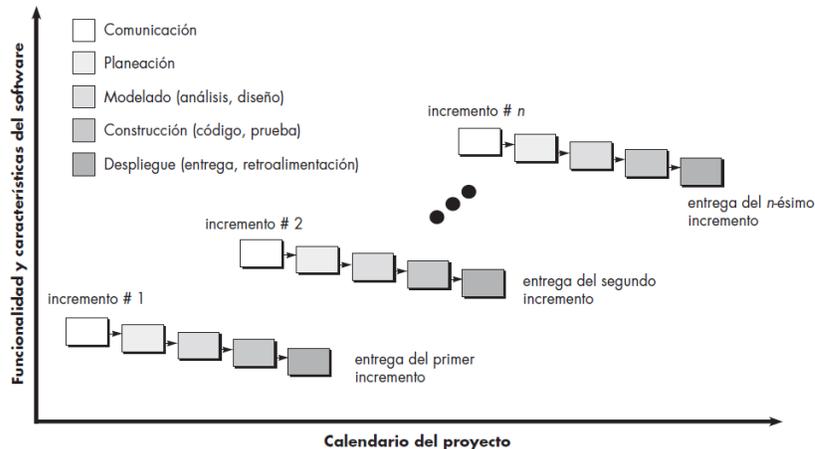
- Documento de requisitos: Se produce al final de la fase de análisis y contiene una especificación de los requerimientos. Es decir, lo que debe realizar el sistema.
- Documento de diseño del software: Se produce al final de la fase de diseño y contiene la descripción de la estructura del sistema y la especificación de sus partes y sus relaciones.
- Código fuente: Es el producto de la codificación. Contiene el código fuente del programa.
- El sistema de software ejecutable producto de la fase de despliegue.
- Documentos de cambios: contiene los cambios realizados al sistema en la fase del mantenimiento.

2.2.2.2 Incremental

El modelo progresivo se basa en la idea de diseñar una aplicación inicial básica para que sea sometida a comentarios del usuario. Lo anterior para que una vez recopilados se desarrollen más versiones de la aplicación hasta producir un sistema adecuado (Sommerville, 2011).

Figura 6

Modelo incremental



Nota. Tomado de Ingeniería del Software: Un enfoque práctico, por Pressman, 2011.

En la primera entrega de la aplicación posiblemente se entreguen los requerimientos básicos, a medida que avanzan las entregas, el software cubre los otros requisitos suplementarios. Cuando el cliente brinda la retroalimentación, se pasa por todas las etapas del desarrollo y se produce otro incremento, cada incremento está más cerca de entregar el producto final. Pressman (2011) menciona cuándo es útil utilizar este modelo:

El desarrollo incremental es útil en particular cuando no se dispone de personal para la implementación completa del proyecto en el plazo establecido por el negocio. Los primeros incrementos se desarrollan con pocos trabajadores. Si el producto básico es bien recibido, entonces se agrega más personal (si se requiere) para que labore en el siguiente incremento. Además, los incrementos se planean para administrar riesgos técnicos (p. 63).

Por otra parte, el modelo progresivo tiene algunos beneficios sobre el modelo en cascada:

1. Se facilita el desarrollo de nuevos requerimientos y se reduce el costo.
2. Se facilita la retroalimentación por parte del cliente al entregar versiones constantes de software.
3. Los clientes tienen la posibilidad de utilizar el software en etapas tempranas del desarrollo.

2.2.2.3 Evolutivo-Iterativo

Los modelos iterativos son evolutivos que iteran sobre las actividades de requerimientos, codificación y validación. Un sistema inicial se desarrolla a partir de los requisitos prioritarios o los que están mejor definidos. Esta primera versión se refina en una nueva iteración con las peticiones del cliente para producir un sistema que satisfaga sus necesidades (Cervantes Ojeda y Gómez Fuentes, 2012).

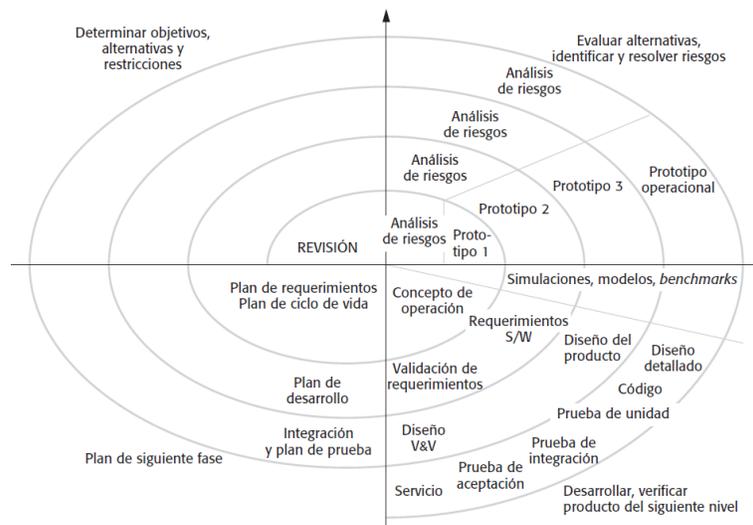
La principal diferencia con el modelo progresivo es que en el modelo iterativo todas las funcionalidades del software se desarrollan en paralelo y en cada iteración se refinan con la retroalimentación del cliente. Por otra parte, en el modelo progresivo se desarrolla una funcionalidad a la vez, una vez terminada esa funcionalidad se recibe retroalimentación por parte del cliente y se continúa con la siguiente funcionalidad. Existen dos enfoques principales para el proceso iterativo, los cuales son:

- **Prototipado:** Cerrada (2006) menciona que: “Un prototipo es un sistema auxiliar que permite probar experimentalmente ciertas soluciones parciales a las necesidades del usuario o a los requisitos del sistema” (p. 25). La idea detrás de diseñar un prototipo inicial del sistema es identificar los requerimientos del software cuando no están tan claros. Una vez listo el prototipo este puede desecharse o desarrollar el software iterativamente sobre este.
- **Espiral:** Es un modelo centrado en el riesgo que se utiliza generalmente en sistemas grandes y complejos. Inteco (2009) menciona que:

Al ser un modelo de ciclo de vida orientado a la gestión de riesgos se dice que uno de los aspectos fundamentales de su éxito radica en que el equipo que lo aplique tenga la necesaria experiencia y habilidad para detectar y catalogar correctamente riesgos (p. 33).

En el modelo en espiral cada ciclo posee cuatro actividades, donde se terminan los objetivos, se analizan los riesgos y se desarrolla y prueba un prototipo o el producto final. Por último, se vuelve a planificar la siguiente actividad. Al final de cada actividad la espiral se mueve hacia afuera, lo que quiere decir que el producto se encuentra más cerca de terminarse.

Figura 7
Modelo en espiral



Nota. Tomado de Ingeniería de Software, por Sommerville, 2011.

2.2.3 Metodologías de desarrollo ágil

En la actualidad, el desarrollo de aplicaciones está envuelto por un mundo globalizado, donde las tecnologías y requerimientos cambian constantemente. Al tener en cuenta lo anterior, el software debe desarrollarse con la rapidez y calidad necesaria para satisfacer los requerimientos cambiantes sin dejar de lado la

calidad del proceso y del producto. Sobre las metodologías ágiles Díaz (2009) afirma lo siguiente:

Las metodologías ágiles contemplan el desarrollo de software de manera integral, con un énfasis especial en la entrega de valor al cliente, en la generación de negocio y el retorno de la inversión (ROI). Sólo hay una manera efectiva de crear software que funcione, y es de manera colaborativa. La colaboración entre cliente y desarrolladores es indispensable: se debe fomentar y apoyar (p. 40).

Molina *et al.* (2018) mencionan que: “Las metodologías ágiles presentan como principal particularidad la flexibilidad, los proyectos en desarrollo son subdivididos en proyectos más pequeños, incluye una comunicación constante con el usuario, son altamente colaborativos y es mucho más adaptable a los cambios” (p. 116).

Para definir la base del desarrollo ágil, en el 2001 se reunieron 17 expertos de la industria del software para impulsar las metodologías ágiles de desarrollo. El objetivo era definir los valores y principios que permitieran desarrollar software rápidamente y adaptándose a los requerimientos cambiantes. En esta reunión se creó The Agile Alliance, una organización que promueve los conceptos que se relacionan con el desarrollo ágil de software y ayudar a las organizaciones para adoptar estos conceptos (Molina *et al.*, 2018). Además, surge un documento denominado El manifiesto ágil que reúne las ideas de la filosofía ágil.

2.2.3.1 El manifiesto ágil

Los puntos abordados en el manifiesto ágil son los siguientes, de acuerdo con Herrera y Valencia (2007):

- **Los individuos e interacciones por encima de los procesos y las herramientas:** El recurso humano se considera como el principal factor de éxito. Se reconoce que contar con equipo humano con las

habilidades técnicas necesarias, que posea habilidades blandas como el trabajo en equipo y adaptación al entorno da mayor garantía de éxito que seguir un proceso riguroso.

- **Software funcionando por encima de la documentación:** En las metodologías ágiles la documentación debe ser corta y solo se deben producir los documentos estrictamente necesarios, ya que se buscan fomentar mecanismos dinámicos de comunicación como la comunicación en persona y el trabajo grupal.
- **La colaboración del cliente por encima de la negociación del contrato:** Después de negociar el contrato se busca que el cliente forme parte del proceso de desarrollo, incluyéndolo en el equipo de trabajo, desde el inicio hasta el final del proyecto y así buscar el beneficio común entre las partes.
- **La respuesta al cambio por encima del seguimiento de un plan:** Debido a que la tecnología cambia constantemente, es normal que los requerimientos del proyecto también lo hagan. Con las metodologías ágiles se busca evitar caer en la planificación estricta, por lo que debe ser flexible para adaptarse a los nuevos cambios. Según mencionan Herrera y Valencia (2007): “Una buena estrategia es hacer planificaciones detalladas para unas pocas semanas y planificaciones mucho más abiertas para los siguientes meses” (p. 383).

Por otra parte, es importante mencionar que en general las metodologías ágiles son marcos de trabajo genéricos aplicables a cualquier disciplina en la que se pueden aplicar procesos y técnicas para el desarrollo de nuevos productos. Asimismo, introduce un ciclo de retroalimentación, cuya meta es construir prácticas que sirvan para el desarrollo de productos complejos, que en este caso son los que están en el desarrollo de software (Amézquita, s. f., p. 3).

A partir de lo anterior existen varios marcos de trabajo con base en la filosofía ágil, las cuales se mencionan a continuación.

2.2.3.2 Marcos de trabajo

- **Scrum:**

Según Molina *et al.* (2018): “Scrum es un marco de trabajo diseñado de tal forma que logra la colaboración eficaz del equipo de trabajo, emplea un conjunto de reglas y se definen roles para generar una estructura de correcto funcionamiento” (p. 117).

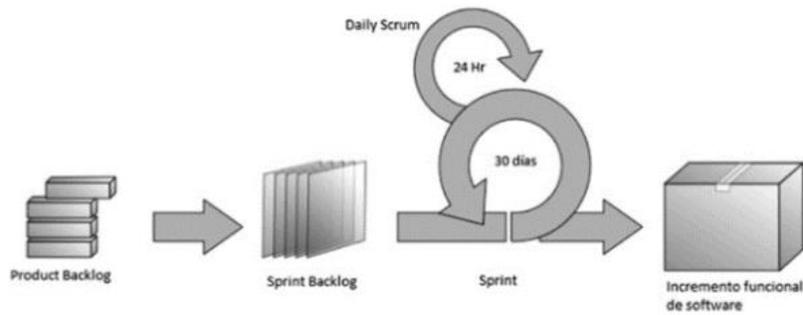
Este marco de trabajo se basa en la formación Scrum del juego rugby. Entre los roles existentes en la metodología Scrum destacan el Scrum master que es quien lidera al equipo, el product owner es el representante de quienes usarán el futuro software en sus actividades diarias. Por último, el equipo de desarrollo es quien traduce los requerimientos establecidos en un documento llamado product backlog en funcionalidades de software.

Por otra parte, dentro de Scrum existen las iteraciones o *sprints* que corresponden a una etapa de trabajo donde se crea una versión utilizable del producto y se considera como un proyecto individual (Molina et al., 2018). Adicionalmente, en el proceso de desarrollo se celebran varias reuniones llamadas ceremonias, entre las cuales Wright (2020) destaca:

- Sprint Planning: En esta reunión se determinan las actividades por completar durante el siguiente sprint.
- Daily Stand-Up: Es una reunión corta de aproximadamente 15 minutos, donde el equipo de desarrollo brinda actualizaciones sobre el trabajo realizado en el día anterior y lo que se realiza durante el día.
- Sprint Demo: El equipo muestra el trabajo que han realizado durante el sprint.
- Sprint Retrospective: En esta reunión se revisa lo que salió bien y mal durante el sprint. La información que se recopile en esta la reunión se utiliza para mejorar el siguiente sprint.

Figura 8

El ciclo Scrum



Nota. Tomado de Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software, por Molina et al., 2018.

- **Programación extrema:**

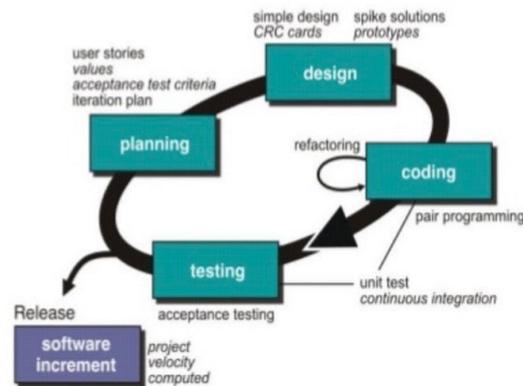
Según Canós *et al.* (s. f.): “Xp es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo” (p. 3).

Además, Sommerville (2011) menciona que: “En la programación extrema, los requerimientos se expresan como escenarios (llamados historias de usuario), que se implementan directamente como una serie de tareas. Los programadores trabajan en pares y antes de escribir el código desarrollan pruebas para cada tarea” (p. 65). Entre las prácticas principales de esta metodología Canós *et al.* (s. f.) destacan:

- Entregas pequeñas: Se producen entregas pequeñas del sistema que sean operativas, aunque no se cumpla con toda la funcionalidad, ya que constituye un valor para el negocio.
- Diseño simple: Se diseña una solución simple para que pueda cambiarse.
- Pruebas: Se escriben pruebas unitarias antes de escribirse el código y se ejecutan cada vez que se modifique el sistema.
- Programación en parejas: La programación del código se realiza en parejas para reducir la posibilidad de errores.
- 40 horas por semana: Se debe evitar trabajar horas extra para que el equipo no se desmotive.

Figura 9

El ciclo XP



Nota. Tomado de *Extreme Programming (XP)*, por Mittal, s. f.

https://www.researchgate.net/figure/Extreme-Programming-XP_fig2_356185795/

- **Lean:**

El pensamiento Lean se basa en una técnica japonesa que creó la Toyota denominada Just in Time (JIT) que busca optimizar el inventario, produciendo solo lo que se necesita y así evitar el desperdicio. Ikonen (2011) menciona que: “El pensamiento Lean enfatiza la producción de valor agregado. Los destinatarios de este valor deben ser los usuarios finales, los clientes u otras partes interesadas” (p. 19).

La metodología Lean posee una serie de principios que son aplicables al desarrollo de software, entre los cuales Poppendieck y Poppendieck (2011) destacan los siguientes:

1. Eliminar el desperdicio: Todo lo que se interponga en el camino que evite satisfacer la necesidad del cliente debe eliminarse.
2. Amplificar el aprendizaje: Aprender constantemente del proceso de desarrollo mejora el aprendizaje para futuros proyectos.

3. Decidir tan tarde como sea posible: Retrasar las decisiones es valioso porque se pueden tomar mejores decisiones cuando se basan en hechos y no en especulaciones.
4. Entregar tan rápido como sea posible: La rapidez asegura que el cliente obtenga lo que requiera de manera rápida. Cuanto más cortos sean los ciclos, más se puede aprender.
5. Empoderar al equipo: Involucrar a los desarrolladores en los detalles de decisiones técnicas es fundamental para lograr la excelencia.
6. Integridad incorporada: El software necesita de un nivel de integridad adecuado, lo cual significa que debe mantener su utilidad durante el tiempo.
7. Ver el todo: Se busca evitar optimizar todo el sistema y no una parte de este.

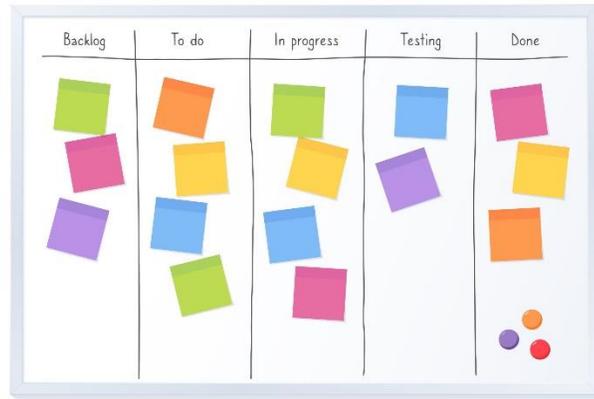
Por otra parte, dentro de la metodología Lean destaca el sistema **Kanban**. Kingberg (citado por Ikonen, 2009) menciona lo siguiente:

Kanban es básicamente un mecanismo de control de flujo para la producción Just-In Time impulsada por extracción en la que se activan las actividades de procesamiento anteriores. En general, Kanban tiene tres reglas: (1) visualizar el flujo de trabajo, (2) limitar el trabajo en curso (WIP) en cada estado del flujo de trabajo, y (3) medir el ciclo de tiempo, es decir, el tiempo promedio para completar un elemento (p. 17).

Es decir, Kanban muestra de manera visual al equipo (generalmente en una pizarra) todas las actividades que están pendientes de realizar, en progreso y terminadas.

Figura 10

Pizarra Kanban



Nota. Tomado de Infografías Kanban, por Mobiliza Academy, 2022.

<https://mobilizaacademy.com/infografias-kanban/>

De esta manera, Lean es una metodología que se utiliza para producir productos de alta calidad, lo que evita el desperdicio y permite entregar más rápido. Lean se puede utilizar, tanto para producir software como para productos en general.

2.2.4 Tipos de software

En la ingeniería del software se encuentran ingenieros en todo el mundo, quienes trabajan duro en proyectos de software en una o más categorías. En ciertos casos se elaboran sistemas nuevos, pero en muchos otros se corrigen, adaptan y mejoran aplicaciones existentes (Pressman, 2010). Pressman (2010) menciona algunas de las categorías o tipos de software (dominio de aplicación):

2.2.4.1 De Sistemas

Según Pressman (2011): “El software de sistemas es el conjunto de programas escritos para dar servicio a otros programas. Determinado software de sistemas (por ejemplo, compiladores, editores y herramientas para administrar

archivos) procesa estructuras de información complejas pero deterministas” (p. 6).

Dentro de este tipo de software se encuentran todos aquellos que realizan una gran interacción con el hardware de la computadora, como los sistemas operativos. Este tipo de software tiene la capacidad de administrar los recursos del sistema, los procesos y el hardware utilizando estructuras de datos complejas (Pressman, 2010).

2.2.4.2 Aplicación

Este tipo de aplicaciones están orientadas a resolver una necesidad específica del negocio. Este tipo de aplicaciones procesa datos comerciales o técnicos para facilitar las operaciones del negocio o la toma de decisiones administrativas y técnicas. Además, se utiliza para controlar las funciones del negocio en tiempo real, como el procesamiento de transacciones en punto de venta, control de procesos de manufactura, entre otros (Pressman, 2010).

2.2.4.3 Incrustado

Es el tipo de software que reside dentro de un producto o sistema para controlar las características para la persona usuaria final. Ejecuta funciones limitadas, ya que generalmente es el que se encuentra en dispositivos como microondas, televisores, automóviles y otros (Pressman, 2011).

2.2.4.4 Inteligencia artificial

Pressman (2011) menciona que este tipo de software:

Hace uso de algoritmos no numéricos para resolver problemas complejos que no son fáciles de tratar computacionalmente o con el análisis directo.

Las aplicaciones en esta área incluyen robótica, sistemas expertos, reconocimiento de patrones (imagen y voz), redes neurales artificiales, demostración de teoremas y juegos (p. 7).

De esta manera, el proceso de desarrollo de software pasa por una serie de etapas o actividades que varían de una metodología a otra y de un modelo a otro. Sin embargo, todo proceso de desarrollo de un software en una organización atraviesa un proceso de administración para determinar sus costos, tiempos, alcances, recursos humanos y otros. Debido a lo anterior es que la disciplina de administración de proyectos se utiliza para la gestión administrativa del proceso de desarrollo. Esta disciplina se detalla a continuación.

2.3 Administración de proyectos de software

El Project Management Institute (PMI, 2017) se refiere a la disciplina de administración de proyectos en el desarrollo de software:

Un proyecto es un esfuerzo temporal emprendido para crear un producto, servicio o resultado. Los proyectos de software, como todos los proyectos, se emprenden para lograr un objetivo específico. Además, al crear nuevos productos, los proyectos de software a menudo se emprenden para modificar un producto de software existente, para integrar un conjunto de componentes de software existentes, para ampliar las capacidades de los productos de software o para modificar la infraestructura de software de una organización (p. 4).

Cada proyecto de software atraviesa un proceso de administración, ya que la ingeniería de software profesional está sujeta siempre a restricciones de presupuesto y fecha. La mala gestión de un proyecto de software da como resultado costos adicionales que no se estimaron, incumplimiento de cronogramas o no cumplir con las expectativas del cliente (Sommerville, 2011).

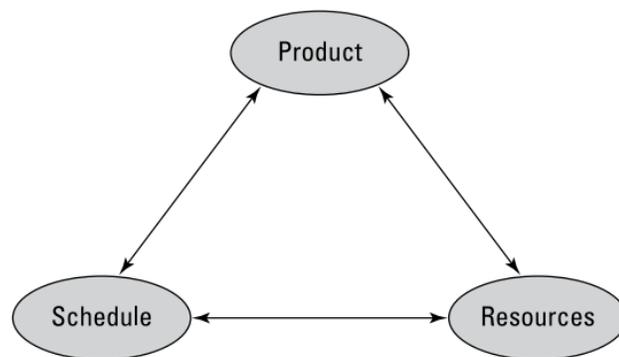
Un proyecto tiene características muy específicas que lo diferencian de otros procesos de la organización. Portny y Portny (2022) mencionan que los proyectos tienen tres componentes principales que se relacionan entre sí:

- **Alcance:** Estos son los resultados que desean o productos.
- **Recursos:** Cantidad necesaria de personas, fondos, equipos, software y otros.
- **Tiempo:** Fechas en las cuales el proyecto inicia y finaliza.

Cada uno de estos componentes tiene incidencia sobre el otro. Por ejemplo, agregar nuevos requerimientos necesita más tiempo o más recursos. Cambiar la fecha de entrega puede incidir sobre los recursos que se necesitan para cumplir con la entrega y así sucesivamente.

Figura 11

Componentes principales de un proyecto



© John Wiley & Sons, Inc.

Nota. Tomado de Project Management for Dummies, por Portny y Portny, 2022.

Para que la administración del proyecto sea exitosa se deben tener claras las metas. En general, algunas de las metas se comparten en la mayoría de los proyectos. Algunas de estas las menciona Sommerville (2011):

1. Hacer entrega del software en el tiempo establecido.
2. Mantener los costos dentro del presupuesto establecido.

3. Entregar el software, de manera que cumpla con los requerimientos establecidos por el cliente.
4. Mantener al equipo de desarrollo en un estado óptimo.

Para lograr lo anterior se deben tener en cuenta los puntos que pueden salir mal para que se eviten en el proceso de administración. Estos son mencionados por Ree (citado por Pressman, 1999):

5. El equipo de desarrollo no entiende las necesidades del cliente.
6. Los cambios no se gestionan de manera adecuada.
7. La tecnología cambió.
8. Las fechas límite son difíciles de cumplir.
9. Pérdida de patrocinio (la Gerencia no asigna los recursos necesarios al proyecto).
10. El equipo de administración de proyectos carece de las habilidades adecuadas.

La planificación, estimación de costos y tiempos de un proyecto de software es más compleja por diferentes razones. Entre ellas destacan que el software se desarrolla utilizando las habilidades cognitivas de los seres humanos (se pueden equivocar), la productividad de los desarrolladores cambia, los requerimientos algunas veces están definidos incorrectamente y la evolución continua de la tecnología puede provocar que los datos históricos y lecciones aprendidas de los proyectos sean inexactos para los nuevos proyectos (PMI, 2017).

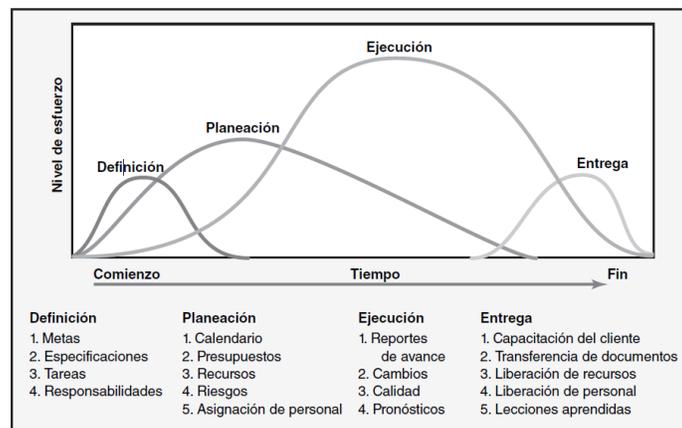
2.3.2 Ciclo de vida del proyecto

Al igual que el proceso de desarrollo de software, el proceso de administración de proyectos tiene un ciclo de vida genérico (aplicable a todas las disciplinas) que atraviesa cuatro etapas principales: definición, planeación, ejecución y entrega (Gray y Larson, 2009). Estas etapas se detallan a continuación:

1. **Definición:** Las especificaciones del proyecto se definen, se establecen los objetivos, se conforman los equipos y se asignan las responsabilidades a los equipos.
2. **Planeación:** Se desarrolla el plan del proyecto, donde se planifica la programación del proyecto, interesados, calidad esperada y el presupuesto.
3. **Ejecución:** Es donde se elabora el producto, además, se controla el proyecto para verificar el cumplimiento de costos, tiempos y especificaciones.
4. **Entrega:** Corresponde a la etapa donde se capacita al cliente, se transfieren los documentos y, finalmente, se liberan los recursos y el personal. Además, se documentan las lecciones aprendidas para futuros proyectos.

Figura 12

Ciclo de vida de un proyecto



Nota. Tomado de *Administración de proyectos*, por Gray y Larson, 2009.

2.3.3 Etapas del proyecto

2.3.3.1 Definición y alcance

Desde el comienzo del proyecto, todas las personas participantes deben tener toda la información que necesitan para realizar sus tareas asignadas, así como la motivación y el compromiso para superar cualquier desafío que puedan

encontrar a medida que avanzan con el trabajo de su proyecto (Porty y Portny, 2022). Uno de los primeros pasos para administrar un proyecto es definir lo que se desea, a esto se le conoce como alcance. Según Gray y Larson (2009): “El alcance describe lo que usted espera entregarle a su cliente cuando termine el proyecto. Su enfoque debe definir los resultados a obtener en términos específicos, tangibles y que puedan ser medidos” (p. 86).

De esta manera, el alcance o enfoque del proyecto define lo que se desea obtener como resultado de este. Generalmente, el alcance del proyecto se define junto con el cliente y el administrador de los proyectos cuando se establecen con claridad los objetivos, productos a entregar, requerimientos y otros.

Por otra parte, cuando se elabora el alcance de un proyecto es recomendable especificar lo que no formará parte del proyecto, producto o servicio. A lo anterior se le llama exclusión (o exclusiones) y se realiza para que el cliente conozca lo que no se le entrega y así evitar falsas expectativas (López y Lankenau, 2017).

Para definir el alcance del proyecto Gray y Larson (2009) proponen una lista de verificación genérica (aplicada a todas las industrias) para garantizar que la definición del proyecto y su alcance estén definidos claramente:

1. Objetivo del proyecto: Se define el objetivo general para satisfacer las necesidades del cliente.
2. Productos para entregar: Corresponde a los productos por entregar durante la vida del proyecto, en el caso de un software pueden entregarse una lista de especificaciones, códigos del software, manuales técnicos y las pruebas finales.
3. Momentos importantes: Identificación de los momentos importantes del proyecto donde se muestren los principales segmentos del trabajo en tiempos, costos y recursos.
4. Requerimientos técnicos: Corresponde a los requerimientos técnicos que tiene el producto por desarrollar.
5. Límites y exclusiones: Se especifica lo que no se realiza.

6. Revisiones con el cliente: El cliente debe aprobar los puntos anteriores.

Por otra parte, López y Lankenau (2017) mencionan que:

Una de las herramientas más importantes para una buena planeación y control de proyectos es la estructura de la división de trabajo (EDT) [...]

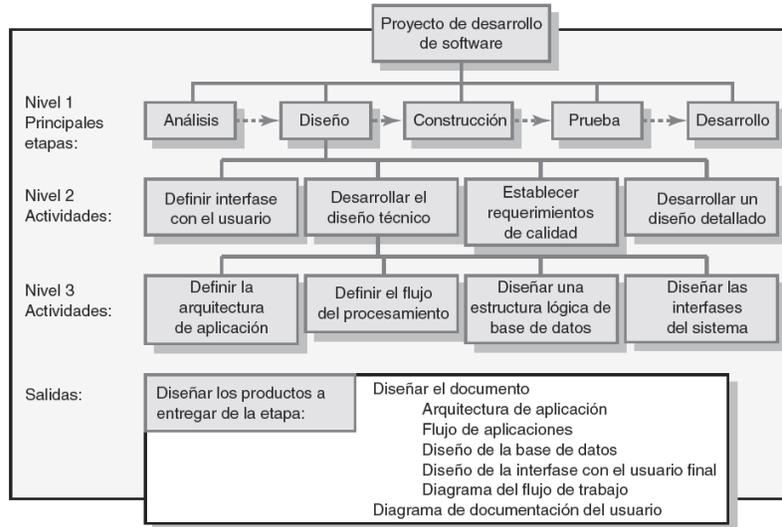
Esta herramienta gráfica o en forma de lista estructurada permite **visualizar todas las actividades implicadas en un proyecto**, de lo general a lo particular, por lo que se le considera una herramienta efectiva para describir el alcance del proyecto (p. 79).

Según Gray y Larson (2017): “La EDT define todos los elementos del proyecto en un marco jerárquico de referencias y establece su relación con el proyecto y sus aspectos” (p. 91).

La estructura jerárquica de la EDT permite la definición del costo y tiempos del proyecto. Cada aspecto de la EDT requiere de un tiempo y cálculo de costos para planear, programar, presupuestar y rastrear el proyecto. En cada jerarquía se define una serie de actividades y subactividades por realizar para que cubran el alcance del proyecto.

Figura 13

EDT para el proyecto de desarrollo de software



Nota. Tomado de *Administración de proyectos*, por Gray y Larson, 2009.

Hasta el momento, la EDT solamente muestra el desglose de las actividades del proyecto, pero no quién las realizará. Para determinar las personas participantes del proyecto se utiliza una matriz de responsabilidades. Esta matriz de responsabilidades incluye quién ejecutará la actividad, quién la coordinará y quién la informará.

Por último, otro de los aspectos por tomar en cuenta es la comunicación para mantener informados a todos los que se involucran en el proyecto. El plan de comunicación del proyecto se puede realizar de manera sencilla en una tabla que indique que información se transmitirá, cuál es el público objetivo, el medio de comunicación y cuál proveedor la brindará.

Figura 14

Plan de comunicaciones

<i>Qué información</i>	<i>Público objetivo</i>	<i>¿Cuándo?</i>	<i>Método de comunicación</i>	<i>Proveedor</i>
Reporte de los eventos importantes	Alta dirección y administrador de proyecto	Cada dos meses	Correo electrónico y copia de respaldo	Oficina de proyecto
Agendas y reportes de avance del proyecto	Personal y cliente	Cada semana	Correo electrónico y copia de respaldo	Administrador de proyecto
Reportes del estado del equipo	Administrador de proyecto y oficina de proyecto	Cada semana	Correo electrónico	Encargado del registro del equipo
Reporte de asuntos	Personal y cliente	Cada semana	Correo electrónico	Encargado del registro del equipo
Reportes de aumento	Personal y cliente	Cuando se requiera	Reunión y copia de respaldo	Administrador de proyecto
Desempeño de los proveedores externos	Personal y cliente	Cada dos meses	Reunión	Administrador de proyecto
Solicitudes aceptadas del cambio	Oficina de proyecto, alta dirección, cliente, personal y administrador de proyecto	En cualquier momento	Correo electrónico y copia de respaldo	Departamento de diseño
Decisiones de supervisión de límites	Alta dirección y administrador de proyecto	Cuando se requiera	Reporte de la reunión por correo electrónico	Grupo de supervisión u oficina de proyecto

Nota. Tomado de *Administración de proyectos*, por Gray y Larson, 2009.

2.3.3.2 Tiempo

Sobre la administración del tiempo de duración del proyecto, Portny y Portny (2022) mencionan lo siguiente:

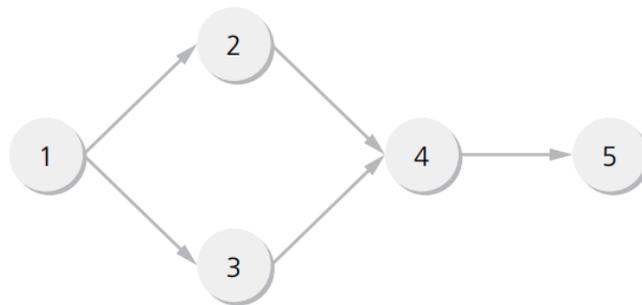
Las asignaciones de proyectos siempre tienen fechas límite. Entonces, aunque no está seguro de lo que se supone que debe lograr su nuevo proyecto, desea saber cuándo debe estar terminado [...]. La verdad es que, cuando recibe por primera vez la asignación de un proyecto, por lo general no se tiene idea de cuánto tardará en completarse. Las reacciones iniciales tienden a basarse más en el miedo. y ansiedad que, en hechos, especialmente cuando estás tratando de hacer malabarismos con múltiples responsabilidades y el proyecto suena complejo (p. 145).

Para estimar el tiempo de un proyecto existen varios mecanismos. El primero consiste en realizar el diagrama de red del proyecto con base en la EDT.

El diagrama de red permite identificar la ruta crítica del proyecto que indica la duración mínima en la que el proyecto se puede llevar a cabo (López y Lankenau, 2017).

Figura 15

Diagrama de red del proyecto



Nota. Tomado de *Administración de proyectos*, por López y Lankenau, 2017.

Cada nodo de la red corresponde a una actividad que se debe realizar y cada flecha corresponde a la siguiente tarea que debe hacerse, donde existe una relación de dependencia entre el nodo siguiente y el predecesor. El sucesor no puede hacerse antes de que el predecesor se encuentre finalizado.

Las actividades de la Figura 15 corresponden a una actividad específica y actividades predecesoras, como se muestra en la Figura 16:

Figura 16

Relación de precedencia entre actividades

Id.	Actividad	Predecesoras	Duración
1	Revisar las instrucciones de la tarea	–	1
2	Investigar en internet	1	2
3	Estudiar con ayuda del libro	1	3
4	Hacer la tarea	2, 3	2
5	Entregar la tarea	4	1

Nota. Tomado de *Administración de proyectos*, por López y Lankenau, 2017.

Una vez definido el diagrama de red se define la **ruta crítica** que permite determinar la secuencia de actividades que se deben realizar para cumplir con el tiempo total del proyecto sin atrasos. Este cálculo puede hacerse mediante las técnicas de **PERT y CPM**.

Otra manera de representar la duración de las actividades de manera visual son los diagramas de Gantt. Los diagramas de Gantt muestran una representación de las actividades, su duración y la secuencia entre ellas con respecto a una escala de tiempo. El diagrama de Gantt es una de las herramientas que más se utilizan en la administración de proyectos (López y Lankenau, 2017).

Figura 17

Diagrama de Gantt

Actividad	Pred	Dur	1	2	3	4	5	6	7	8	9	10
1 Revisar las instrucciones	–	1	■									
2 Investigar en internet	1	2		■	■							
3 Estudiar con ayuda del libro	1	3		■	■	■						
4 Hacer la tarea	2,3	2					■	■				
5 Entregar la tarea	4	1								■		

Nota. Tomado de *Administración de proyectos*, por López y Lankenau, 2017.

El diagrama de Gantt anterior muestra cada una de las actividades y una barra que se extiende a lo largo que representa la duración de la actividad. La unidad de tiempo puede representarse en días, semanas, meses y otros.

Otro aspecto importante corresponde a la asignación de los tiempos a las actividades del proyecto. La estimación del tiempo que durará cada actividad se basa en dos tipos de estimaciones, la **estimación análoga** y la **estimación paramétrica**. En la estimación análoga se toman como referencia proyectos similares y experiencias pasadas para estimar los tiempos de las actividades del proyecto actual, es decir, un juicio de expertos. Por otro lado, la **estimación paramétrica** se basa en relaciones estadísticas entre datos históricos y otras variables, con la finalidad de definir parámetros para la actividad como costo, presupuesto y duración (López y Lanckenau, 2017).

La estimación paramétrica generalmente se basa en fórmulas matemáticas para estimar la asignación de tiempos a las actividades. En el caso de los proyectos de software existen varias técnicas de estimación, algunas de ellas se mencionan por Pressman (2011):

1. Cocomo II.
2. Puntos de función.
3. PERT.

Por otra parte, en los proyectos que poseen una administración ágil la estimación de tiempos varía sobre el enfoque tradicional. Sommerville (2011) menciona lo siguiente:

Los métodos ágiles de desarrollo de software son enfoques iterativos donde el software se desarrolla y entrega a los clientes en incrementos. A diferencia de los enfoques dirigidos por un plan, la funcionalidad de dichos incrementos no se planea por anticipado, sino que se decide durante el desarrollo. La decisión acerca de qué incluir en un incremento depende del progreso y las prioridades del cliente (p. 631).

Para realizar la planificación en metodologías ágiles existen dos etapas para la planeación, las cuales se mencionan por Sommerville (2011):

1. **Planeación de la entrega:** prevé con muchos meses de antelación y decide sobre las características que deben incluirse en una entrega de un sistema.
2. **Planeación de la iteración:** la cual tiene un panorama a corto plazo y se enfoca en la planeación del siguiente incremento de un sistema.

De esta manera, al inicio del proyecto se define un plan de implementación (reléase plan) que contiene un conjunto de historias de usuario que reflejan las características que debe tener el sistema, aunque no se encuentre del todo detallado. Posteriormente, el equipo lee las historias de usuario y se clasifican de acuerdo con la cantidad de tiempo que se requiere para realizar la historia, la cantidad de tiempo está dada en términos relativos (hipotéticos) en lugar de términos absolutos (cantidad de días u horas) para evitar estimaciones erróneas. La estimación del tiempo se realiza mediante **puntos de historia**.

una vez estimadas las historias, el esfuerzo relativo se traduce en la primera estimación del esfuerzo total requerido usando la noción de velocidad (Sommerville, 2011). En Scrum la velocidad del equipo se determina por la cantidad de puntos de historia completados al final de cada iteración.

Por ejemplo, si el equipo estima que las historias de usuario que se realizan en la primera iteración tienen un total de 30 puntos, una historia de 60 puntos tardará dos iteraciones en completarse. El tiempo que dura la iteración en semanas está definido por la metodología que se utiliza.

2.3.3.3 Costo

López y Lankenau (2017) mencionan que: “La preparación del presupuesto del proyecto consiste en estimar, detallar y, posteriormente, controlar los costos que se originan con la utilización de recursos humanos y materiales” (p. 145).

Para obtener el costo de un proyecto se deben tomar en cuenta los recursos humanos que se utilizan y los materiales. Posteriormente, se realiza una

estimación de los recursos que se usan por medio de las actividades programadas del proyecto. Finalmente, se estima el costo de cada uno de los recursos tomando en cuenta costos directos (que se relacionan directamente con la actividad) e indirectos (no se relacionan con el alcance del proyecto). El costo de cada recurso se establece a lo largo del proyecto, este último paso permite conocer el **presupuesto**.

Los costos de mano de obra y materiales se clasifican como **costos directos**. Otros costos como el alquiler de las instalaciones, equipos informáticos, costos de energía e Internet se clasifican como **costos indirectos**.

En etapas tempranas del proyecto donde no se conoce del todo la factibilidad de este es necesario realizar **estimaciones**. Una estimación es una valoración aproximada que se realiza ante la falta de información real. Estas estimaciones se realizan para autorizar el presupuesto y se preparan después de que el administrador del proyecto realizó la mayor parte de la planeación del proyecto y una vez que el diseño conceptual del producto o servicio se aceptó (López y Lankenau, 2017). La precisión de estas estimaciones varía de acuerdo con el nivel de detalle y a la cantidad de información con la que se cuenta en el momento.

En el caso de los proyectos de software, el costo que tiene la mano de obra (equipo de desarrollo) se calcula multiplicando la cantidad de horas de cada recurso por el costo del recurso por hora.

$$\text{Costo} = \text{Horas recurso} * \text{Costo del recurso}$$

Para calcular las horas del recurso se utiliza la siguiente fórmula:

$$\text{Horas recurso} = \frac{\text{Número de recursos} * \text{Horas de la jornada laboral} * \text{Número de jornadas}}{\text{Número de jornadas}}$$

Por ejemplo, para ejecutar un proyecto de instalación de servidores de red en cuatro días, trabajando a turno completo (8 horas diarias) se requerirán tres técnicos en informática. Cada técnico gana ₡2.000 por hora.

Paso número uno: se requiere calcular las horas recurso:

$$\text{Horas recurso} = 3 \text{ técnicos} * 8 \text{ horas} * 4 \text{ días} = 96 \text{ horas recurso}$$

Finalmente, se calcula el costo:

$$\text{Costo} = 96 \text{ horas recurso} * 2000 = \text{₡}192.000$$

Por lo tanto, el proyecto tiene un costo de ₡192.000 solo tomando en cuenta las horas hombre.

Por otra parte, otro componente importante para calcular el costo del proyecto son los costos indirectos. Para calcular los costos indirectos se aplica un porcentaje a los gastos directos. Este porcentaje se calcula mediante la siguiente fórmula:

$$\text{Costos indirectos} = [1 - (\text{costos directos del año anterior} / \text{ingresos totales})] * 100$$

En el ejemplo anterior, si la empresa tuvo un ingreso anual de ₡1.000.000 y sus costos directos fueron ₡700.000 durante todo el año el porcentaje de costos indirectos es de 30 %. Por lo tanto, este porcentaje debe sumarse al costo anterior. De esta manera, el costo total de la instalación de servidores con costos directos e indirectos es de ₡249.600.

Otro de los costos para tomar en cuenta es la reserva para imprevistos. Según López y Lankenau (2017), las reservas para imprevistos:

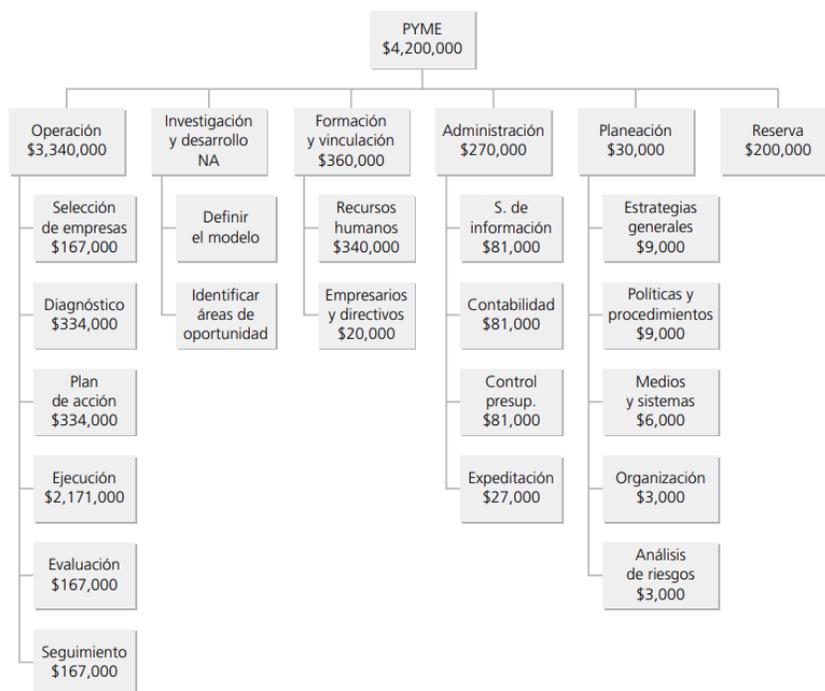
Se utilizan para prevenir desviaciones imputables al proyecto, como cambios en el costo de materiales, modificación en la productividad de la

mano de obra, errores en el diseño, retrasos por causas predecibles (como lluvia) o deficiencia en la entrega de materiales (p. 140).

De esta manera, una vez que se calculó el costo del proyecto se debe crear un presupuesto. El objetivo principal es comunicar de manera clara y ordenada los gastos del proyecto a todos los implicados, es decir, a los clientes, directivos, equipo de trabajo y a proveedores, cuando se requiera (López y Lanckenau, 2017). El presupuesto debe integrarse con las actividades definidas en la etapa de alcance. Una EDT con el presupuesto integrado se denomina estructura de división de costos (EDC).

Figura 18

Estructura de división de costos



Nota. Tomado de *Administración de proyectos*, por López y Lanckenau, 2017.

2.3.3.4 Recursos humanos

Según PMI (2017):

El equipo del proyecto está formado por personas con funciones y responsabilidades asignadas que trabajan colectivamente para lograr un objetivo de proyecto compartido. El gerente del proyecto debe invertir un esfuerzo adecuado en adquirir, administrar, motivar y empoderar al equipo del proyecto (p. 309).

Una de las estrategias para definir los miembros que participarán en el proyecto es contar con un registro de habilidades y áreas del conocimiento de cada uno de los miembros (Portny y Portny, 2022).

Tabla 6

Habilidades del equipo de trabajo

Habilidades		
No.	Nombre	Descripción
1	Daniel Bello	Programación en .NET

Para definir cuáles miembros están en el proyecto se utiliza una matriz de recursos humanos que representa las personas asignadas a cada actividad del proyecto y el esfuerzo de trabajo con el que cada persona contribuirá con cada asignación (Portny y Portny, 2022).

Tabla 7

Control de actividades y horas del proyecto

Actividad		Personal (horas hombre)		
Código EDT	Nombre de la actividad	Nombre del miembro 1	Nombre del miembro 2	Nombre del miembro 3
2.1.1	Implementación de la regla de negocio 1	32	0	24

2.1.2	Implementación de la regla de negocio 2	0	40	60
2.1.3	Implementación de la regla de negocio 3	40	0	8
Total:		72	40	92

Por otra parte, para manejar el equipo del proyecto deben tomarse en cuenta los siguientes aspectos según el PMI (2017):

1. Diversidad
2. Localización física.
3. Adquisición del equipo humano: De qué manera se contactarán para el proyecto.
4. Manejo del equipo: Corresponde a las técnicas que se utilizan para manejar al equipo y a la existencia de herramientas organizacionales.
5. Manejo del ciclo de vida: Cómo se maneja el ciclo de vida del proyecto.

Otra herramienta que se utiliza en la gestión del recurso humano es la matriz de roles y responsabilidades que muestra las personas participantes del proyecto y su rol. La matriz de responsabilidades es una herramienta sencilla de construir, pero muy poderosa porque con ella se define claramente quién hace qué en el proyecto, lo que evita confusiones u omisiones (López y Lankenau, 2017).

Figura 19

Matriz de roles y responsabilidades

EDT	E	C	I
1 Elaborar un manual		Andrés Ortigosa	
1.1 Diseño		Ana Suástegui	
1.1.1 Diseñar el contenido	Antonio Ramos		Aquivaldo Martínez (proveedor)
1.1.2 Diseñar el formato	Antonio Ramos		
1.2 Redacción		Aaron Arano	
1.2.1 Redactar el manual	Angélica Sánchez		
1.2.2 Revisar ortografía	Anselmo Suárez		
1.2.3 Revisar estilo	Anselmo Suárez		
1.3 Edición	Alma Urquidi		Aurora Téllez (asesora)

E = ejecuta, C = coordina, I = informa, es decir, proporciona información necesaria para que se ejecute la actividad.

Nota. Tomado de *Administración de proyectos*, por López y Lankenau, 2017.

2.3.3.5 Calidad

Según PMI (2017):

La Gestión de la Calidad del Proyecto incluye los procesos para incorporar la política de calidad de la organización en cuanto a la planificación, gestión y control de los requisitos de calidad del proyecto y el producto, a fin de satisfacer los objetivos de los interesados (p. 271).

De esta manera, el proyecto debe pasar por un proceso de control de calidad definido en el plan de calidad de cada proyecto. Sommerville (2011) menciona que: “La planeación de calidad es el proceso de desarrollar un plan de calidad para un proyecto. El plan de calidad debe establecer las cualidades deseadas de software y describir cómo se valorarán” (p. 653).

Humphrey (citado por Sommerville, 2011) sugiere una estructura general para un plan de calidad que cumpla con los siguientes aspectos:

- Introducción del producto: Se describe el producto y sus expectativas de calidad.
- Planes del producto: Se indican las fechas críticas de entrega, responsabilidades y planes de implementación.

- Metas de calidad: Se definen las metas y planes de calidad para el producto, esto incluye la identificación de los atributos especiales de la calidad.
- Riesgos y gestión del riesgo: Se identifican los riesgos para evitar afectar la calidad del producto.

Por otra parte, PMI (2017) menciona que un plan de gestión de la calidad debe tener en cuenta los requisitos establecidos en el plan de proyecto. Este plan debe tener en cuenta aspectos como los siguientes:

- Estándares de calidad.
- Objetivos de calidad.
- Roles y responsabilidades en cuanto a la calidad.
- Entregables y procesos sujetos a la revisión de la calidad.
- Actividades de control y gestión de calidad del proyecto.
- Herramientas de calidad que se utilizan para el proyecto.

Una vez definido el plan de calidad, este debe ejecutarse durante la ejecución del proyecto. Durante esta ejecución se producirán mediciones de control de calidad, informes de desempeño de trabajo, solicitudes de cambio y otros.

2.3.3.6 Riesgos

Según López y Lankenau (2017): “Todo proyecto implica riesgos debido a la incertidumbre y los posibles cambios internos y externos. Llevar a una buena administración de riesgos es necesario para prevenirlos, evitarlos o, al menos, disminuir sus efectos” (p. 144).

La tarea de gestionar los riesgos del proyecto es una de las más importantes para anticipar, evitar y mitigar la materialización de contratiempos que cambien las fechas definidas en el calendario o la calidad del software por entregar.

Para definir los riesgos del proyecto, Sommerville (2011) sugiere los siguientes pasos:

- Identificar el riesgo.
- Analizar el riesgo: Se valora la probabilidad y consecuencias de los riesgos que se identifican.
- Planeación del riesgo: Se planea la estrategia para enfrentar los riesgos para evitar que se materialicen o para mitigar sus efectos en caso de que se materialicen.
- Monitorización de riesgos: Se revisan constantemente los riesgos que se identifican y sus planes, además, se intenta modificarlos cuando se aprenda más sobre estos.

Para los proyectos de software, Sommerville (2011) menciona que existen tres categorías de riesgo:

- Riesgos del proyecto: Son aquellos que alteran los calendarios y los recursos del proyecto. Cuanto más se retrase el proyecto, se requiere aumentar su costo.
- Riesgos del producto: Son aquellos que afectan la calidad del producto de software, como un defecto no detectado a tiempo o la caída de un servidor.
- Riesgos empresariales: Corresponden a los riesgos que afectan a la organización, como la falta de recursos para seguir invirtiendo en el proyecto.

Todos estos riesgos se deben incluir en el plan del proyecto para hacerles frente desde etapas tempranas. En los proyectos de software la gestión de los riesgos es particularmente importante debido a la incertidumbre que enfrentan la mayoría de este tipo de proyectos, generalmente por requerimientos no aclarados, cambios de requerimientos, falla en las estimaciones y otros (Sommerville, 2011).

2.3.3.7 Control

Una vez iniciado el proyecto se requiere evaluar y controlar el avance de este. Según Gray y Larson (2009):

El control hace que la gente se haga responsable, evita que los pequeños problemas se multipliquen y se hagan grandes; además, mantiene el enfoque [...]. El control es una de las áreas más descuidadas de la administración de proyectos. Por desgracia, no es raro encontrar resistencia para controlar los procesos (p. 387).

Para lograr el control se utilizan sistemas de supervisión para recolectar datos como el avance actual del proyecto como costo actual, fecha de terminación, problemas que se requieran abordar, excesos de costos y otros. Esta supervisión se puede realizar mediante programas informáticos.

Una vez recopilados los datos se realiza un informe que según Gray y Larson (2009) posee el siguiente formato:

- Progreso desde el último informe.
- Estado actual del proyecto.
 1. Programa
 2. Costo
 3. Alcance
- Tendencias acumulativas.
- Problemas y asuntos desde el último informe.
 1. Acciones y solución de problemas anteriores.
 2. Nuevas variaciones y problemas que se identifican.
- Acciones correctivas planeadas.

Por otra parte, López y Lankenau (2017) mencionan que para validar el alcance se utilizan los criterios de aceptación que se definieron junto con los involucrados en el proceso de planeación. Con base en esos criterios se formaliza la aceptación de los entregables del proyecto.

En el caso de las metodologías ágiles, al final de cada iteración el equipo de desarrollo se reúne con el cliente para hacer una demostración del trabajo realizado. Una vez terminada la demostración, se documenta la retroalimentación del cliente para que se incluya en la siguiente iteración. Estas reuniones permiten controlar el avance del proyecto desde la perspectiva del cliente.

Para representar el avance del proyecto de manera visual López y Lankenau (2017) proponen dar a conocer los siguientes datos durante el informe de control:

- Cantidad de dinero gastado en mano de obra, materiales y recursos.
- El porcentaje de las horas de mano de obra de diferentes áreas (desarrolladores, analistas, probadores y otros).
- Comparación entre costos reales y proyectados.
- Porcentaje de avance: se calcula dividiendo el tiempo transcurrido entre el total de tiempo programado.

Este proceso de control debe hacerse hasta que se finalice el proyecto para evitar incurrir en tiempos y gastos adicionales.

2.3.3.8 Cierre

Gray y Larson (2009) mencionan que:

Eventualmente, cada proyecto llega a su fin. En algunos proyectos el final puede no ser tan claro como se podría esperar. Aunque la declaración del alcance puede definir una terminación clara de un proyecto, la conclusión real puede o no corresponder (p. 437).

Para determinar cuándo un proyecto ha finalizado Gray y Larson (2009) mencionan una serie de condiciones que determinan el cierre de un proyecto:

1. Condiciones normales: En este tipo de condición el proyecto finaliza con la terminación del producto sin mayores contratiempos.
2. Prematuras: Algunos proyectos pueden finalizar antes de la fecha esperada debido a la eliminación de una de sus partes. Antes de

finalizar un proyecto prematuramente se deben analizar los riesgos que implica tomar esta decisión.

3. Perpetuas: Corresponden a los proyectos que se llenaron de retrasos y costos adicionales y que están lejos de cumplir con la fecha de finalización estimada. El retraso se debe a los nuevos requerimientos que se agregan al proyecto u otros factores.
4. Proyecto fracasado: Es cuando el proyecto falla por diversas razones.
5. Cambios de prioridad: Cuando las prioridades de la organización cambian y se deben cerrar algunos proyectos.

Cuando el proyecto se cierra la entrega se formaliza con el cliente y el patrocinador. El administrador del proyecto junto con los involucrados realiza las siguientes actividades para formalizar el cierre (López y Lankenau, 2017):

6. Verificación del trabajo completado: Se revisa la información de las etapas previas para verificar que el trabajo se completó. Se verifica el cumplimiento de los objetivos y los criterios de aceptación de cada requerimiento.
7. Implementación: Se verifica que el producto se transfiera al área de producción y operaciones.
8. Lecciones aprendidas: Se recolectan las lecciones aprendidas durante todas las etapas del proyecto y se archivan para su futuro uso.

Al final del proceso de cierre, se liberan las responsabilidades mutuas y se da por concluido el proyecto de común acuerdo entre las partes (López y Lankenau, 2017).

De esta manera, el proceso de cierre da por concluido el proyecto. A modo de conclusión, todo proyecto pasa por una serie de etapas: alcance, costo, tiempo, recursos, control-calidad y cierre. Los proyectos se administran de acuerdo con una metodología o estándar que varía según el tipo de proyecto,

desde las más tradicionales (como PMBOK) hasta metodologías ágiles (como Scrum).

2.3.4 Estándares

En la disciplina de administración de proyectos existen diferentes metodologías y estándares que permiten gestionar el proyecto en todas sus etapas. La mayoría de estos estándares solo especifica el conjunto de tareas que se deben realizar sin especificar cómo. Algunos solo se aplican para proyectos de software y otros para proyectos de cualquier disciplina, algunos en sus versiones más recientes tienen integración con metodologías ágiles. A continuación, se mencionan los estándares que más se utilizan en la administración de proyectos y cada una de sus etapas:

2.3.4.1 El estándar para la dirección de proyectos de PMI y la guía PMBOK

Según Gray y Larson (2009):

El Project Management Institute (Instituto de Administración de Proyectos, PMI, por sus siglas en inglés) se fundó en 1969 como una sociedad internacional para administradores de proyecto [...]. El PMI ofrece certificación como Project Management Professional (Profesional en administración de proyectos, o PMP, por sus siglas en inglés) (p. 4).

Por otra parte, el estándar del PMI se basa en el Estándar para la Dirección de Proyectos del Instituto Nacional de Normalización de los Estados Unidos (ANSI). El estándar de PMI está escrito en una guía denominada Project Management Book of Knowledge (PMBOK).

La última versión del PMBOK es la versión 7 que se lanzó en el año 2021. Esta última versión se enfoca en **brindar valor** a las personas interesadas al seguir una serie de principios, en lugar de las tradicionales áreas de conocimiento de las versiones anteriores. Al respecto, PMI (2021) menciona que: “Esta edición

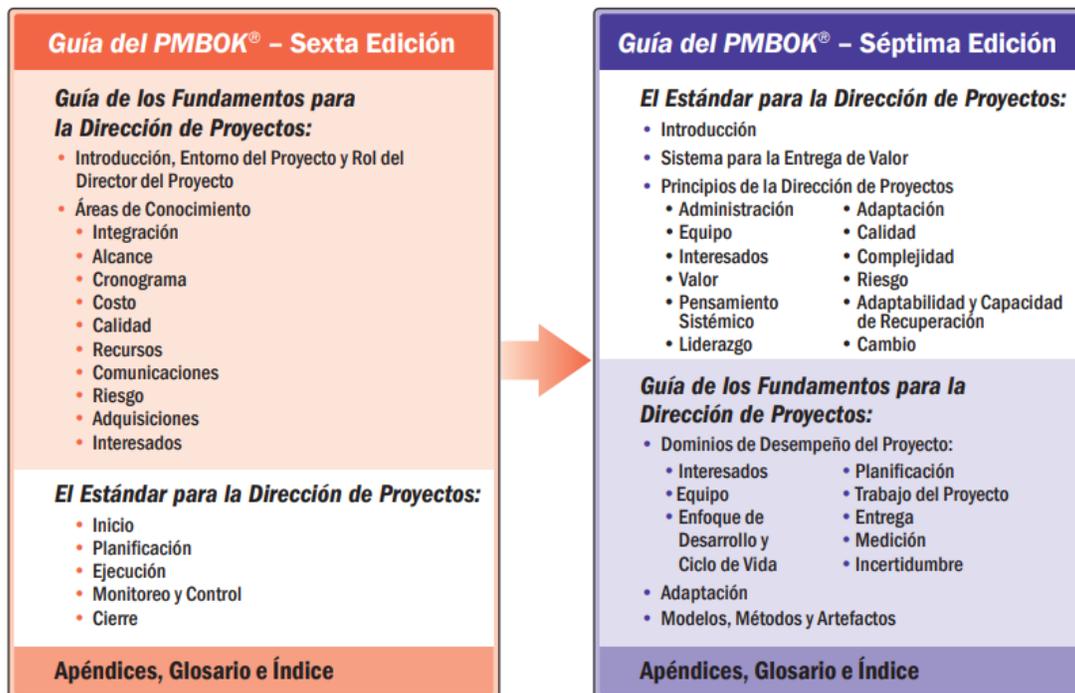
cambia a un estándar basado en principios con el fin de apoyar la dirección eficaz de los proyectos, y centrarse más en los resultados previstos que en los entregables” (p. 10).

Esta serie de principios busca brindar un enfoque adaptativo (adaptarse a los cambios) a los proyectos en lugar de un enfoque predictivo (predecir los cambios desde el inicio).

Esta nueva versión de la guía PMBOK hace hincapié en que los proyectos no solo producen salidas, sino que lo que es más importante, permiten que esas salidas impulsen resultados que, en última instancia, aportan valor a la organización y a sus interesados (PMI, 2021). Además, en conjunción con los principios establecidos en esta guía, se encuentran ocho dominios de desempeño. Según PMI (2021): “Un dominio de desempeño es un grupo de actividades relacionadas que son fundamentales para la consecución efectiva de los resultados de los proyectos” (p. 12).

Figura 20

Comparación entre la guía PMBOK sexta y séptima edición



Nota. Tomado de Guía de los Fundamentos para la Dirección de Proyectos y el Estándar para la Dirección de Proyectos, por Project Management Institute, 2021.

Por otra parte, los nuevos principios que establece esta guía son los siguientes, según el PMI (2021):

- **Valor:** Beneficio obtenido por la organización (interesados internos y externos) al desarrollar el proyecto. Además, para dar valor a la entidad los proyectos deben concordar con los objetivos estratégicos del negocio.
- **Administración:** La administración abarca responsabilidades, tanto internas como externas a la organización e incluye integridad, cuidado, confiabilidad y cumplimiento (PMI, 2021).
- **Equipo:** Este principio se enfoca en los miembros del equipo que trabajan colaborativamente para alcanzar los objetivos del proyecto.
- **Interesados:** Son personas que se benefician con la ejecución del proyecto. Se debe involucrar a las personas interesadas durante el proyecto para garantizar su éxito y la satisfacción del cliente.
- **Pensamiento sistémico:** El pensamiento sistémico implica tener una visión holística sobre cómo las partes del proyecto interactúan y con sistemas externos (PMI, 2021).
- **Liderazgo:** Ejercer un liderazgo efectivo para promover el éxito del proyecto. El liderazgo puede provenir de cualquier miembro del equipo.
- **Adaptación:** El éxito del proyecto se basa en adaptarse al contexto único de este para determinar los métodos más apropiados para producir los resultados que desean (PMI, 2021).
- **Calidad:** Implica que las necesidades y los requisitos del proyecto o producto se cumplan con base en criterios de aceptación.
- **Complejidad:** Implica manejar adecuadamente la complejidad del proyecto producida en el momento de gestionar el alcance, las comunicaciones, los interesados, el riesgo y la innovación tecnológica (PMI, 2021).
- **Riesgo:** El riesgo debe evaluarse y mitigarse para minimizar los impactos negativos que afecten al proyecto.

- **Adaptabilidad y resiliencia:** Los proyectos deben adaptarse y recuperarse de las situaciones imprevistas y cambiantes.
- **Cambios:** Preparar a los afectados para la adopción y el mantenimiento de comportamientos y procesos nuevos para la transición del estado actual al estado futuro previsto que creó los resultados del proyecto (PMI, 2021).

Por otra parte, los dominios establecidos en la guía PMBOK séptima edición son los siguientes:

- **Interesados:** Se encarga del involucramiento de las personas interesadas del proyecto a lo largo de este. Las personas interesadas corresponden a los beneficiarios que brindan apoyo durante el proyecto.
- **Equipo:** Se encarga de las actividades y funciones asociadas con las personas responsables de producir los entregables del proyecto que hacen realidad los resultados de negocio (PMI, 2021).
- **Enfoques de desarrollo y ciclos de vida:** Se encarga del manejo del enfoque de desarrollo y las fases del ciclo de vida del proyecto. Cada proyecto puede tener uno o varios enfoques de desarrollo y ciclos de vida.
- **Planificación:** Aborda las actividades y funciones asociadas con la organización y coordinación iniciales, continuas y en evolución, necesarias para la entrega de los entregables y resultados del proyecto (PMI, 2021).
- **Trabajo del proyecto:** El trabajo de proyecto está asociado con el establecimiento de los procesos y la realización del trabajo para permitir que el equipo de proyecto cumpla con los entregables y resultados que se esperan (PMI, 2021).
- **Entrega:** La entrega del proyecto se enfoca en cumplir con los requisitos, el alcance y las expectativas de calidad para producir los

entregables esperados que impulsarán los resultados previstos (PMI, 2021).

- **Medición:** La medición involucra evaluar el desempeño del proyecto e implementar respuestas apropiadas para mantener un desempeño óptimo (PMI, 2021).
- **Incertidumbre:** Se ocupa de las actividades y funciones asociadas con el riesgo y la incertidumbre (PMI, 2021).

2.3.4.2 CMMI

Según Chrissis *et al.* (2011) los modelos CMMI (Capability Maturity Model Integration) son: “Colecciones de las mejores prácticas que ayudan a las organizaciones a mejorar sus procesos. Estos modelos se desarrollan por equipos de las disciplinas de la industria, gobierno y en Instituto de Desarrollo de Software (SEI)” (p. 15).

El modelo aplicado al desarrollo de software se conoce como CMMI-DEV. Este modelo provee una serie de guías para desarrollar productos y servicios. Además, provee una serie de prácticas que cubren la administración de proyectos, manejo de procesos y otros procesos de apoyo que se utilizan en el desarrollo y mantenimiento (Chrissis *et al.*, 2011).

Esta metodología posee un conjunto de áreas de proceso que son un conjunto de prácticas que se relacionan en cada área. Existen 22 áreas de proceso y únicamente se mencionan las relacionadas con la gestión de proyectos:

- Gestión de requisitos.
- Planificación de los proyectos.
- Monitoreo y control del proyecto.
- Gestión integrada del proyecto.
- Gestión de riesgos.
- Gestión de acuerdos con proveedores.
- Gestión cuantitativa del proyecto.

Otro aspecto para tomar en cuenta es que cada área del proceso tiene metas específicas que se deben cumplir para satisfacer el área respectiva. Además, existen metas llamadas genéricas porque la misma declaración de objetivos se aplica a múltiples áreas de proceso (Chrissis *et al.*, 2011, p. 23).

Por otra parte, CMMI posee niveles de madurez que consisten en prácticas específicas que se relacionan para un conjunto de áreas de proceso que mejoran el desempeño de la organización (Chrissis *et al.*, 2011, p. 23). Los niveles de madurez se mencionan a continuación:

- **Inicial:** Los procesos que suelen ser caóticos y, por lo general, no respaldados por la organización.
- **Manejado:** Los procesos por implementar de CMMI se planifican de acuerdo con la política y con personas capacitadas que tienen los recursos necesarios para producir salidas, involucrar y monitorear los procesos.
- **Definido:** Son procesos que se han integrado a la organización y se incluyen en las políticas de estas.
- **Analizado cuantitativamente:** La organización establece objetivos cuantitativos para la calidad y el desempeño del proceso. Esos objetivos se basan en las necesidades del cliente, usuarios y de la empresa.
- **Optimizado:** Los procesos se mejoran continuamente con base en la medición de los objetivos cuantitativos. Constantemente, se revisan para mejorar el rendimiento de los procesos que se basa en los objetivos de la organización.

Figura 21

Niveles de madurez de CMMI



Nota. Tomado de Niveles de Madurez.

https://www.researchgate.net/figure/lustracion-de-los-Niveles-de-Madurez-CMMI-Recuperado-de_fig1_339206762/

A continuación, únicamente para las **áreas del proceso de planificación y monitoreo** se presenta una tabla que muestra cada una de las áreas del proceso con actividades.

Tabla 8*Proceso de planificación del proyecto según CMMI*

Actividad	Subactividad	Prácticas por realizar
Establecer las estimaciones	Estimar el alcance	<ul style="list-style-type: none">• Desarrollar la EDT con sus principales tareas• Descomponer la EDT en subactividades para facilitar la estimación de los tiempos y responsabilidades• Identificar productos y componentes que deben adquirirse de forma externa• Identificar los productos que deben reutilizarse
	Establecer estimaciones del trabajo	<ul style="list-style-type: none">• Determinar el tamaño y complejidad de cada función• Determinar enfoque técnico (características de la arquitectura, funcionalidades y atributos de calidad)
	Determinar las fases del ciclo de vida	<ul style="list-style-type: none">• Determinar el ciclo de vida del proyecto de acuerdo con la metodología

	Estimar el costo y el esfuerzo	<ul style="list-style-type: none"> • Recolectar datos históricos • Determinar las necesidades de infraestructura requerida. • Estimar el esfuerzo mediante datos históricos y modelos.
Desarrollar el plan del proyecto	Determinar el presupuesto y calendario	<ul style="list-style-type: none"> • Determinar los hitos • Determinar las dependencias entre actividades • Establecer y mantener el calendario y presupuesto.
	Identificar los riesgos	<ul style="list-style-type: none"> • Identificar los riesgos • Documentar los riesgos • Revisar los riesgos
	Planificar el manejo de datos	<ul style="list-style-type: none"> • Requerimientos y procedimientos para asegurar la privacidad y seguridad de los datos. • Establecer mecanismos de acceso a los datos.
	Planificar los recursos del proyecto	<ul style="list-style-type: none"> • Determinar los requerimientos de las comunicaciones • Determinar las necesidades del personal • Determinar las necesidades de equipo y componentes requeridos.

	Planificar las habilidades y conocimiento	<ul style="list-style-type: none"> • Identificar el nivel de conocimiento requerido para llevar a cabo el proyecto. • Acceder a los registros de conocimiento y habilidades disponibles. • Seleccionar mecanismos para entrenar al personal
	Plantificar el involucramiento de los patrocinadores	<ul style="list-style-type: none"> • Determinar la lista de patrocinadores, sus relaciones e importancia.
	Establecer el plan de proyecto	<ul style="list-style-type: none"> • Puntos anteriores.

2.3.4.3 ISO

Otro estándar para la administración de proyectos es la norma 21500 de las normas ISO (Organización Internacional de Normalización). Esta norma provee una orientación sobre conceptos y procesos de gestión de proyectos que son importantes y que tienen impacto en su desempeño (ISO, 2012).

Esta norma al igual que PMBOK establece que todo proyecto se divide en grupos de procesos y grupos de materias. Los grupos de procesos establecidos por la norma son los siguientes:

1. **Inicio:** Estos son los procesos que se realizan al iniciar una fase o un proyecto para definir los objetivos de la etapa o fase y para que el administrador autorice su inicio.
2. **Planificación:** Estos son los procesos que se realizan para planificar el proyecto.
3. **Implementación:** Corresponde a los procesos realizados para ejecutar el proyecto.
4. **Control:** Procesos para monitorear, medir y controlar el rendimiento del proyecto de acuerdo con su plan. Se realizan acciones correctivas y preventivas para alcanzar los objetivos del proyecto.
5. **Cierre:** Estos son los procesos que se realizan para terminar una etapa del proyecto o todo el proyecto. Se establecen las lecciones aprendidas.

Por otra parte, los grupos de materias cuyas etapas se mencionaron anteriormente son las siguientes:

6. **Integración:** Actividades que se relacionan para identificar, definir, combinar, unificar y controlar las actividades y procesos del proyecto (PMI, 2012).
7. **Partes interesadas:** Incluye los procesos para identificar y manejar el patrocinio del proyecto, clientes e interesados.
8. **Alcance:** Incluye los procesos requeridos para identificar y definir el trabajo y los entregables y solo el trabajo y los entregables requeridos.

9. **Recursos:** Incluye los procesos para identificar y adquirir recursos adecuados para el proyecto como personas, equipo, materiales, infraestructura y herramientas.
10. **Tiempo:** Incluye las actividades para calendarizar las actividades del proyecto y su monitoreo a lo largo de este.
11. **Costo:** Incluye los procesos requeridos para desarrollar el presupuesto y monitorear los costos.
12. **Riesgo:** Procesos para identificar los riesgos y amenazas del proyecto.
13. **Calidad:** Procesos para planificar y establecer el control de calidad del proyecto.
14. **Adquisiciones:** Procesos para adquirir productos o servicios y gestionar la relación con los proveedores.
15. **Comunicación:** Incluye los procesos para distribuir la información relevante al proyecto.

Por otra parte, para cada materia dentro del grupo hay una serie de entradas y salidas que se gestionan, generalmente documentos físicos, las cuales se detallan en la Tabla 9:

Tabla 9

Entradas y salidas para las actividades de la administración de proyectos según la normativa ISO 21500.

Actividad	Entradas	Salidas
Desarrollo del acta de constitución	<ul style="list-style-type: none"> • Enunciado de alcance • Contrato • Caso de negocio 	<ul style="list-style-type: none"> • Acta de constitución
Desarrollo del plan de proyecto	<ul style="list-style-type: none"> • Acta de constitución • Lecciones aprendidas de los proyectos anteriores • Caso de negocio 	<ul style="list-style-type: none"> • Plan del proyecto
Control del proyecto	<ul style="list-style-type: none"> • Plan del proyecto • Datos de progreso • Mediciones de calidad • Registro de riesgos • Registros de problemas 	<ul style="list-style-type: none"> • Solicitudes de cambio • Reportes de progreso • Informes de finalización del proyecto
Cierre	<ul style="list-style-type: none"> • Reportes de progreso • Documentación contractual • Informes de finalización del proyecto 	<ul style="list-style-type: none"> • Procedimientos completados • Reporte de cierre del proyecto • Liberación de recursos
Patrocinadores	<ul style="list-style-type: none"> • Registro de patrocinadores • Plan del proyecto 	<ul style="list-style-type: none"> • Solicitudes de cambio
Alcance	<ul style="list-style-type: none"> • Acta de constitución 	<ul style="list-style-type: none"> • Enunciado de alcance

	<ul style="list-style-type: none"> • Aprobaciones 	<ul style="list-style-type: none"> • Requerimientos
Creación de la EDT	<ul style="list-style-type: none"> • Plan del proyecto • Requerimientos • Aprobaciones 	<ul style="list-style-type: none"> • EDT • Diccionario de la EDT
Definición de las actividades	<ul style="list-style-type: none"> • EDT • Diccionario de la EDT • Plan del proyecto • Aprobaciones 	<ul style="list-style-type: none"> • Lista de actividades
Establecer el equipo de trabajo	<ul style="list-style-type: none"> • Requerimientos de recursos • Organigrama • Disponibilidad de recursos • Plan del proyecto • Descripción de los roles 	<ul style="list-style-type: none"> • Asignaciones de personal • Contratos de personal
Estimar la duración de las actividades	<ul style="list-style-type: none"> • Lista de actividades • Requerimientos de recursos • Datos históricos • Aprobaciones 	<ul style="list-style-type: none"> • Estimación de la duración de las actividades
Calendarización del proyecto	<ul style="list-style-type: none"> • Estimación de la duración de las actividades • Lista de actividades 	<ul style="list-style-type: none"> • Calendario

Estimación de costos	<ul style="list-style-type: none"> • Registro de riesgos • Aprobaciones • EDT • Lista de actividades • Plan del proyecto • Aprobaciones 	<ul style="list-style-type: none"> • Estimaciones de costos
Elaboración del presupuesto	<ul style="list-style-type: none"> • EDT • Estimaciones de costos • Cronograma • Plan del proyecto • Aprobaciones 	<ul style="list-style-type: none"> • Presupuesto
Control de costos	<ul style="list-style-type: none"> • Reportes de progreso • Plan del proyecto • Presupuesto 	<ul style="list-style-type: none"> • Costos reales • Costos pronosticados • Solicitudes de cambio • Acciones correctivas
Identificación de riesgos	<ul style="list-style-type: none"> • Plan del proyecto 	<ul style="list-style-type: none"> • Registro de riesgos
Evaluación de riesgos	<ul style="list-style-type: none"> • Registro de riesgos • Plan del proyecto 	
Manejo de riesgos	<ul style="list-style-type: none"> • Registro de riesgos • Plan del proyecto 	<ul style="list-style-type: none"> • Respuestas al riesgo

Planificación de la calidad	<ul style="list-style-type: none"> • Plan del proyecto • Requerimientos de calidad • Política de calidad • Aprobaciones 	<ul style="list-style-type: none"> • Plan de calidad
Control de la calidad	<ul style="list-style-type: none"> • Datos de progreso • Entregables • Plan de calidad 	<ul style="list-style-type: none"> • Mediciones de la calidad • Verificación de entregables • Reportes de inspección • Solicitudes de cambio • Acciones correctivas

Nota. *Elaboración propia a partir del estándar ISO 21500.*

2.3.4.4 IEEE

El estándar elaborado por la IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) relacionado con la administración de proyectos se denomina Estándar para Planes de Gestión de Proyectos de Software 1058. La IEEE (1998) menciona lo siguiente sobre el alcance de este estándar:

Este estándar especifica el formato y el contenido de los **Planes de Proyecto** (SPMP). Esta norma no especifica las técnicas exactas a ser utilizado en el desarrollo de un SPMP, ni proporciona ejemplos de SPMP. Cada organización que utilice este estándar debe desarrollar un conjunto de prácticas y procedimientos para proporcionar una guía detallada para la preparación y actualización de los SPMP basados en este estándar (p. 3).

De esta manera, este estándar al igual que los otros estándares, no especifica cómo se deben llevar a cabo las actividades, sino que es una guía de referencia sobre qué es lo que se debe realizar y en cuál orden. Este estándar se divide en cláusulas, cada cláusula contiene una descripción general y una especificación detallada del estándar, incluidos los componentes requeridos que deben incluirse y los componentes opcionales que pueden incluirse en planes de gestión de proyectos de software (SPMP) con base en este estándar (IEEE, 1998).

En el ámbito general, este estándar menciona el contenido que debe incluir un plan de proyecto. Este contenido se menciona a continuación:

Tabla 10

Formato de un plan de proyecto de software según la norma IEEE 1058

Cláusula	Subcláusula
----------	-------------

Visión general del proyecto	<ul style="list-style-type: none"> • Resumen del proyecto • Propósito, alcance y objetivos • Suposiciones y restricciones • Entregables • Calendario y resumen del presupuesto • Historial de cambios
Organización del proyecto	<ul style="list-style-type: none"> • Interfaces externas • Estructura interna • Roles y responsabilidades
Procesos de gestión	<ul style="list-style-type: none"> • Plan de inicio • Plan de estimaciones • Plan de adquisición de recursos • Plan del personal • Plan de trabajo • Actividades • Asignación de tiempos • Asignación de recursos • Asignación de presupuesto • Plan de control de requerimientos • Plan de manejo del calendario • Plan del control de calidad • Plan de colección de métricas • Plan de manejo de riesgos • Plan de cierre del proyecto
Procesos técnicos	<ul style="list-style-type: none"> • Modelo de proceso • Métodos, técnicas y herramientas • Plan de infraestructura • Plan de aceptación

Planes de soporte	<ul style="list-style-type: none">• Plan de gestión de la configuración• Plan de verificación y validación• Plan de documentación• Plan de aseguramiento de la calidad• Revisiones y auditorías
-------------------	---

Nota. Elaboración propia a partir del estándar IEEE 1058.

De esta manera, los diferentes estándares para la administración de proyectos son genéricos para varias disciplinas. La mayoría de estos estándares únicamente especifican la lista de tareas por realizar sin especificar cómo deben hacerse. Cada empresa puede utilizar el estándar que más se adapte a sus necesidades.

CAPÍTULO III:
MARCO METODOLÓGICO

3 Marco metodológico

3.1 Enfoque de investigación

El tipo de enfoque de la presente investigación es **cuantitativo**, ya que según Sampieri et al. (2014) el enfoque cuantitativo: “Utiliza la recolección de datos para probar hipótesis con base en la medición numérica y el análisis estadístico, con el fin de establecer pautas de comportamiento y probar teorías” (p. 40).

Por otra parte, Barrantes (1999) menciona que en la investigación cuantitativa: “El investigador es un elemento externo al objeto que se investiga. En este proceso utiliza las técnicas estadísticas en el análisis de datos y generaliza los resultados. Parte de constructos hipotéticos para explicar ciertos fenómenos” (p. 71).

Con base en la afirmación anterior, este estudio es **cuantitativo** debido a que para elaborar una guía de administración de proyectos de software para este tipo de empresas es necesario realizar una recolección y análisis de datos sobre la forma en la que gestionan sus proyectos. Este análisis estadístico permite generalizar los resultados a la población (todas las compañías mypes dedicadas al desarrollo de software) para elaborar una guía genérica para este tipo de organizaciones.

3.2 Tipo de investigación

Sampieri *et al.* (2014) mencionan que en los estudios descriptivos: “Se busca especificar las propiedades, las características y los perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis” (p. 92).

Torres *et al.* (2010) menciona que en los estudios **descriptivos**: “Se muestran, narran, reseñan o identifican hechos, situaciones, rasgos, características de un objeto de estudio, o se diseñan productos, modelos, prototipos, guías, etcétera, pero no se dan explicaciones o razones de las situaciones, los hechos, los fenómenos, etcétera” (p. 113).

Por otra parte, Sampieri *et al.* (2014) mencionan que la investigación **exploratoria**: “Se realiza cuando el objetivo es examinar un tema o problema de investigación poco estudiado, del cual se tienen muchas dudas o no se ha abordado antes” (p. 97).

Con base en las afirmaciones anteriores la presente investigación es descriptiva, ya que se describe el proceso de gestión de proyectos de software en las mypes del país y con base en esta descripción se busca diseñar una guía personalizada de administración de proyectos para este tipo de empresas. Por otra parte, la presente investigación también posee un enfoque **exploratorio** debido a que existen pocos estudios sobre la gestión de proyectos de software para mypes, tanto en el ámbito nacional como en el ámbito internacional.

3.2.1 Fuentes de investigación

3.2.1.1 Fuentes primarias

Según Torres *et al.* (2010):

Las fuentes primarias son todas aquellas de las cuales se obtiene información directa, es decir, de donde se origina la información. Es también conocida como información de primera mano o desde el lugar de los hechos. Estas fuentes son las personas, las organizaciones, los acontecimientos, el ambiente natural, etcétera (p. 191).

De esta manera, en la presente investigación se utiliza como fuentes primarias a los administradores de los proyectos (o colaboradores que realicen una función relacionada con esta disciplina) de las empresas de la muestra que se seleccionó. Se aplica un instrumento de recolección de datos, con preguntas principalmente cuantitativas que permitan conocer la manera en la que se gestionan los proyectos de software en la organización.

3.2.1.2 Fuentes secundarias

Torres *et al.*, (2010) mencionan que:

Las fuentes de información secundaria son todas aquellas que ofrecen información sobre el tema que se va a investigar, pero que no son la fuente original de los hechos o las situaciones, sino que sólo los referencian. Las principales fuentes secundarias para la obtención de la información son los libros, las revistas, los documentos escritos (en general, todo medio impreso), los documentales, los noticieros y los medios de información (p. 192).

De esta manera, en la presente investigación las fuentes de información secundarias corresponden a libros, revistas indexadas, artículos científicos, tesis, artículos de Internet y otros que se relacionan con el desarrollo de software y la administración de proyectos que ayuden a sustentar la teoría de la investigación.

3.3 Técnicas de recolección de datos

En cuanto a los instrumentos de recopilación, se pretende utilizar un **cuestionario** con preguntas abiertas y cerradas al personal de las empresas mypes, con el objetivo de determinar el nivel de conocimiento sobre las prácticas y procesos de gestión de proyectos que utiliza la organización y la capacitación que ha recibido el personal en estos temas. Bernal *et al.* (2010) mencionan que: “La encuesta se fundamenta en un cuestionario o conjunto de preguntas que se preparan con el propósito de obtener información de las personas” (p. 210).

Por otra parte, se presenta una **revisión bibliográfica** de fuentes secundarias para contrastar lo que establecen los estándares de administración de proyectos consultados y lo que realizan las empresas en su quehacer diario.

3.4 Consideraciones éticas

En la aplicación del cuestionario se garantiza la confidencialidad de las respuestas al no solicitar ningún tipo de información personal. De esta forma, se garantiza el anonimato de las personas participantes del estudio.

3.5 Población

Según Jany (1994 citado por Torres *et al.*, 2010), población es: “La totalidad de elementos o individuos que tienen ciertas características similares y sobre las cuales se desea hacer inferencia” (p. 48), o bien unidad de análisis.

La población del presente proyecto corresponde a las **empresas desarrolladoras de software** con personería jurídica que se encuentren dentro de la categoría micro y pequeña empresa (mypes). Como se mencionó en el Capítulo I, la CCSS establece que una organización entra en la categoría de microempresa si la misma posee de uno a cinco trabajadores y pequeña si posee de seis a 30 trabajadores.

Según establece el MEIC, existen **219 empresas** que se encuentran dentro de la categoría de micro y pequeña empresa y que se dedican a actividades de programación informática. De ese conglomerado existen compañías con personería jurídica y física dentro de todo el territorio nacional.

Tabla 11

Empresas mypes por región socioeconómica

Región	Cantidad de empresas
Central Oriental	163
Central Occidental	37
Brunca	6
Chorotega	4
Huetar Atlántica	2
Huetar Norte	5

Pacífico Central	2
Total	219

Es importante mencionar que de las 219 compañías que poseen la actividad de programación informática un total de 68 empresas no tiene información actualizada de contacto. Por lo tanto, se presume que ya no existen, a pesar de que siguen activas en el MEIC.

De esas 151 empresas restantes se encontró que el mayor conglomerado se encuentra en la Gran Área Metropolitana, en los cantones de **Curridabat, Escazú y Montes de Oca** donde se encuentran ubicadas 38 compañías. La Tabla 13 muestra el desglose de las empresas por cantón.

Tabla 12

Empresas mypes en los tres cantones con mayor concentración

Cantón	Cantidad de empresas
Curridabat	18
Escazú	7
Montes de Oca	13
Total	38

Con base en lo anterior la población corresponde a **38 compañías**.

3.6 Criterios de inclusión

Los criterios para que una empresa se considere dentro de la población de estudio son los siguientes:

- La empresa debe estar catalogada como mypes.
- La empresa debe ser micro o pequeña (máximo 30 trabajadores).
- Debe dedicarse a labores de programación informática.
- Constituida bajo personería jurídica.

- Encontrarse entre los tres cantones con mayor cantidad de empresas en la GAM.

3.7 Criterios de exclusión

- Empresas que se ubiquen fuera del país.
- Empresas de tamaño mediano o grande (ya que generalmente tienen su proceso definido, documentado y validado).
- Empresas con cédula física.

3.8 Muestra

Barrantes (1999) se refiere al concepto de muestra:

Pocas veces podrá medirse a toda la población, por eso tiene que trabajar con base en muestras que debe ser el reflejo fiel de la población. Por eso las muestras deben ser representativas. Las muestras pueden dividirse en dos grandes grupos: las probabilísticas y las no probabilísticas. En las muestras probabilísticas todos los elementos de la población tienen la misma posibilidad de ser seleccionados y las no probabilísticas no dependen de la probabilidad, si no a otras causas relacionadas con el investigador o el estadígrafo (p. 135).

Debido a lo anterior, el muestreo que se utiliza en la investigación corresponde al **no probabilístico por conveniencia**, debido a que las empresas que se estudian se seleccionaron a conveniencia. Con lo anterior se busca tener fácil acceso a la información ya que algunas compañías en el momento de hacer la selección tienen un estado inactivo o no poseen ningún medio de contacto.

Sampieri *et al.* (2014) mencionan que: “En las muestras no probabilísticas, la elección de los elementos no depende de la probabilidad, sino de causas relacionadas con las características de la investigación o los propósitos del

investigador” (p. 417). Por otra parte, para calcular el tamaño de la muestra se utilizó la siguiente fórmula matemática:

Figura 22

Fórmula para calcular el tamaño de la muestra

$$\text{Tamaño de la muestra} = \frac{\frac{z^2 \times p(1-p)}{e^2}}{1 + \left(\frac{z^2 \times p(1-p)}{e^2 N}\right)}$$

Nota. Tomado de *Calculadora del tamaño de muestra* (Fotografía), por SurveyMonkey, s. f. <https://es.surveymonkey.com/mp/sample-size-calculator/>

La Tabla 5 muestra el significado de las variables de la fórmula y el procedimiento aplicado para sustituir los valores:

Tabla 13

Variables que se utilizan en el cálculo de la muestra

Variable	Descripción	Valor de la variable
Z	Puntuación z de acuerdo con el nivel de confianza deseado.	Se desea un nivel de confianza del 90 %, por lo que la puntuación z es 1.65 .
P	Tamaño de la población	38 empresas
E	Margen de error	0.10
N	P * e * z	38 * 0.10 * 1.65

Por esto, la muestra que formará parte del estudio es de **25 compañías**.

3.9 Matriz metodológica

Tabla 14

Matriz metodológica

Objetivo específico	Variables	Conceptualización	Dimensión	Indicadores y métrica	Instrumentos
Conocer la forma en la que se gestionan los proyectos de <i>software</i> en las mypes desarrolladoras de <i>software</i> de los tres cantones que concentran la mayor cantidad de empresas en la Gran Área Metropolitana mediante consultas a las partes interesadas.	Gestión de proyectos	Conjunto de metodologías para planificar y dirigir los procesos de un proyecto.	Metodologías	<ul style="list-style-type: none"> - Nombre y descripción de las etapas - Entradas y salidas de las etapas. 	Recopilación de datos bibliográficos (libros, tesis, Internet y otros).
			Etapas		
	mypes que desarrollan <i>software</i> .	Empresas que no superan los 30 trabajadores.	Tamaño	<ul style="list-style-type: none"> - Cantidad de colaboradores (<5, 6-15, > 15) 	Cuestionario.

			Tipo de <i>software</i>	<ul style="list-style-type: none"> - Tipo de <i>software</i> que desarrolla (de sistemas, aplicación, incrustado o inteligencia artificial). 	
			Proceso de administración de proyectos en las empresas.	<ul style="list-style-type: none"> - Metodologías de administración de proyectos (PMI, CMMI, ISO, IEEE, combinación de metodologías, otras metodologías). - Manejo del alcance, costo, tiempo y recursos del proyecto. - Software que se utiliza para 	

				administrar el proyecto.	
			Capacidad técnica	<ul style="list-style-type: none"> - Puesto que ocupa en la empresa (desarrollador, administrador de los proyectos, líder técnico y otros). - Conocimiento en metodologías de administración de proyectos (alto/medio/bajo) 	
			Metodologías de desarrollo tradicionales y ágiles	<ul style="list-style-type: none"> - Metodologías de desarrollo que se utilizan (cascada, evolutivo, incremental, iterativo o metodologías ágiles). 	

Identificar los estándares que más se utilizan por las empresas durante su proceso de administración de proyectos de <i>software</i> mediante la recopilación de datos.	Estándares de administración de proyectos	Conjunto de metodologías existentes para administrar proyectos.	Metodologías y estándares existentes.	<ul style="list-style-type: none"> - Cantidad de etapas - Orden de las etapas - Entradas y salidas de las etapas 	Recopilación de datos bibliográficos (libros, tesis, Internet y otros).
	Mypes que desarrollan <i>software</i> .	Empresas que no superan los 30 trabajadores.	Proceso de administración de proyectos en las empresas.	<ul style="list-style-type: none"> - Metodologías de administración de proyectos (PMI, CMMI, ISO, IEEE, combinación de metodologías, otras metodologías). - Manejo del alcance, costo, tiempo y recursos del proyecto. - Software que se utiliza para administrar el proyecto. 	Cuestionario.

Determinar el estándar de administración de proyectos más adecuado para este tipo de empresas mediante el análisis de la información que se recopiló.	Estándares de administración de proyectos	Conjunto de metodologías existentes para administrar proyectos.	Metodologías y estándares existentes.	<ul style="list-style-type: none"> - Nombre de las etapas - Entradas y salidas de las etapas 	Recopilación de datos bibliográficos (libros, tesis, Internet y otros)
	Metodologías de administración de proyectos existentes en las empresas	Metodologías de administración de proyectos que utilizan las compañías para administrar sus proyectos de <i>software</i> .	Proceso de administración de proyectos en las empresas.	<ul style="list-style-type: none"> - Metodología de administración de proyectos que se utiliza (PMI, CMMI, ISO, IEEE, combinación de metodologías). - Software que se utiliza para administrar el proyecto. 	Cuestionario.
Diseñar la metodología de administración de proyectos de <i>software</i> que requiere este tipo de organizaciones,	Estándares de administración de proyectos	Conjunto de metodologías existentes para administrar proyectos.	Metodologías y estándares existentes.	<ul style="list-style-type: none"> - Nombre de las etapas - Entradas y salidas de las etapas 	Recopilación de datos bibliográficos (libros, tesis, Internet y otros).

<p>mediante los estándares internacionales consultados.</p>	<p>Metodologías de administración de proyectos existentes en las empresas</p>	<p>Metodologías de administración de proyectos que utilizan las compañías para administrar sus proyectos de <i>software</i>.</p>	<p>Proceso de administración de proyectos en las empresas.</p>	<ul style="list-style-type: none"> - Metodología de administración de proyectos que se utiliza (PMI, CMMI, ISO, IEEE, combinación de metodologías). - Manejo del alcance, costo, tiempo y recursos del proyecto. - Software que se utiliza para administrar el proyecto. 	<p>Cuestionario</p>
---	---	--	--	---	----------------------------

CAPÍTULO IV:
ANÁLISIS SITUACIONAL

4 Análisis situacional

4.1 Consideraciones sobre el instrumento aplicado

En el presente capítulo se presentan los principales hallazgos de la investigación al aplicar el instrumento de recolección de datos. Este cuestionario posee preguntas abiertas y cerradas que abarcan diferentes áreas de proceso de la gestión de proyectos, entre ellas:

- Generalidades de las empresas.
- Alcance.
- Administración del tiempo.
- Costo y recursos.
- Riesgos.
- Calidad.
- Cierre.

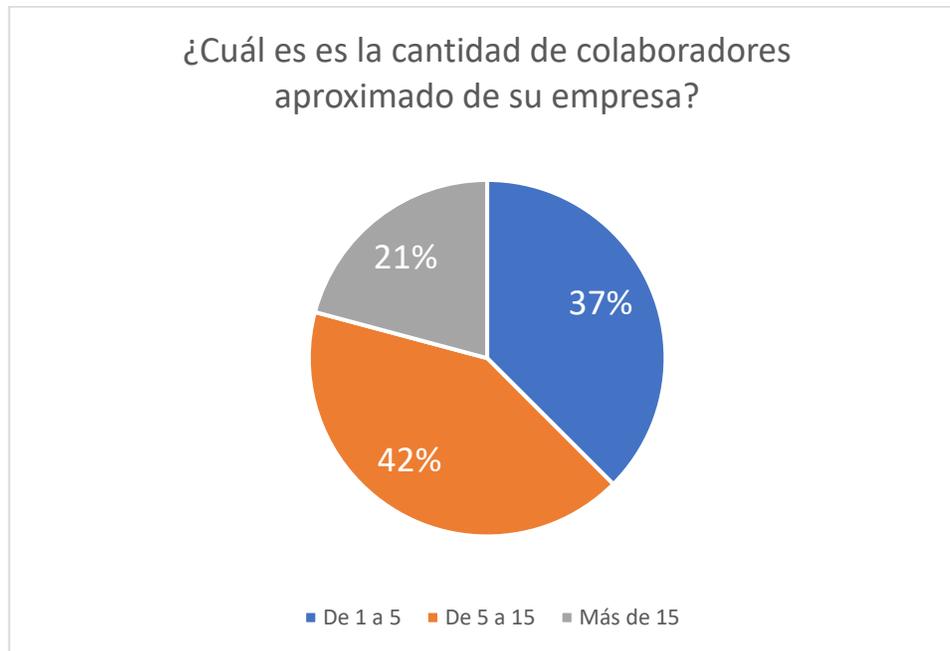
Un total de 38 empresas fueron contactadas vía teléfono y correo electrónico donde se les envió el cuestionario. De ese total de compañías, 24 respondieron el cuestionario, para casi un 100 % del tamaño de la muestra calculada en el marco metodológico que corresponde a 25 empresas.

4.2 Generalidades de las empresas analizadas

Uno de los aspectos consultados corresponde a la cantidad de colaboradores que poseen las empresas. El Gráfico 1 muestra que un 37 % tiene de 1 a 5 colaboradores, un 42 % tiene de 5 a 15 colaboradores y un 21 % posee más de 15 colaboradores, las compañías de 5 a 15 colaboradores son las que predominaron en el estudio. De lo anterior, se observa que la mayoría de las empresas se ubica dentro de la categoría de **pequeña empresa**, debido a que la CCSS establece que una compañía entra en esta categoría si posee de 6 a 30 empleados.

Gráfico 1

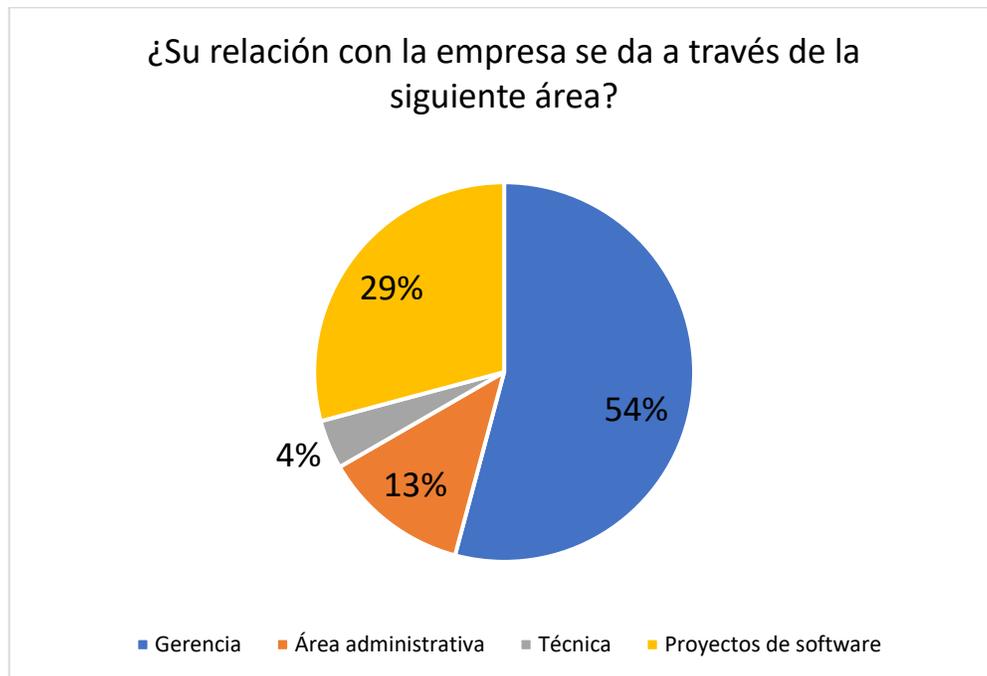
Tamaño aproximado de las empresas consultadas



Por otra parte, el 54 % de las personas encuestadas tiene una relación gerencial con la organización. Además, un 29 % se relaciona con la administración de proyectos, 13 % con el Área Administrativa y únicamente un 4 % con el área técnica. De esta manera, la mayoría de los encuestados posee una relación gerencial por lo que se puede asumir que tienen conocimiento amplio del negocio.

Gráfico 2

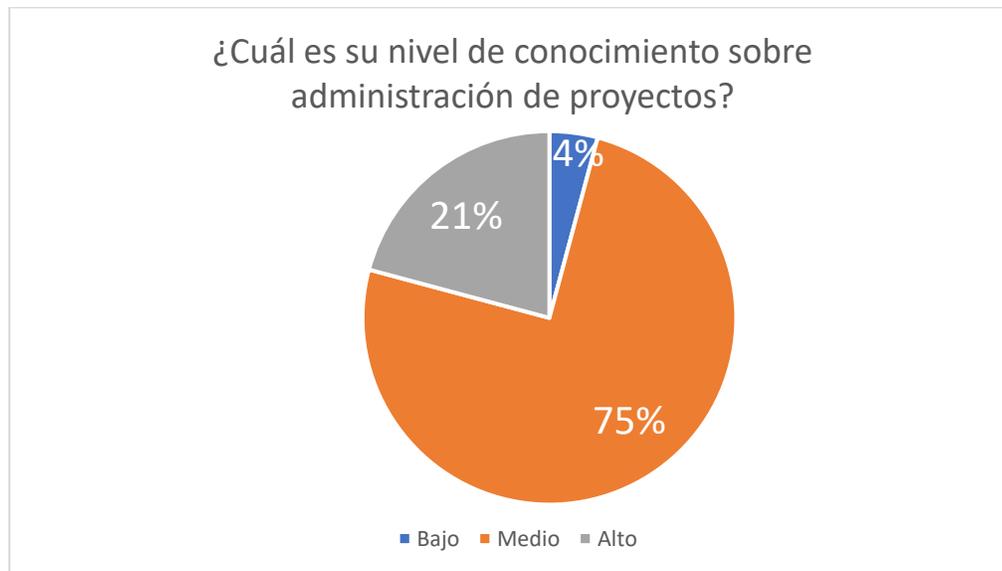
Relación laboral de las personas que participaron de la encuesta



Respecto al nivel de conocimiento en administración de proyectos de las personas que participaron del estudio, se encontró que la mayoría tiene un conocimiento medio sobre el tema con un 75 %, un 21 % tiene un conocimiento alto y un 4 % tiene un conocimiento bajo, como se muestra en el Gráfico 3.

Gráfico 3

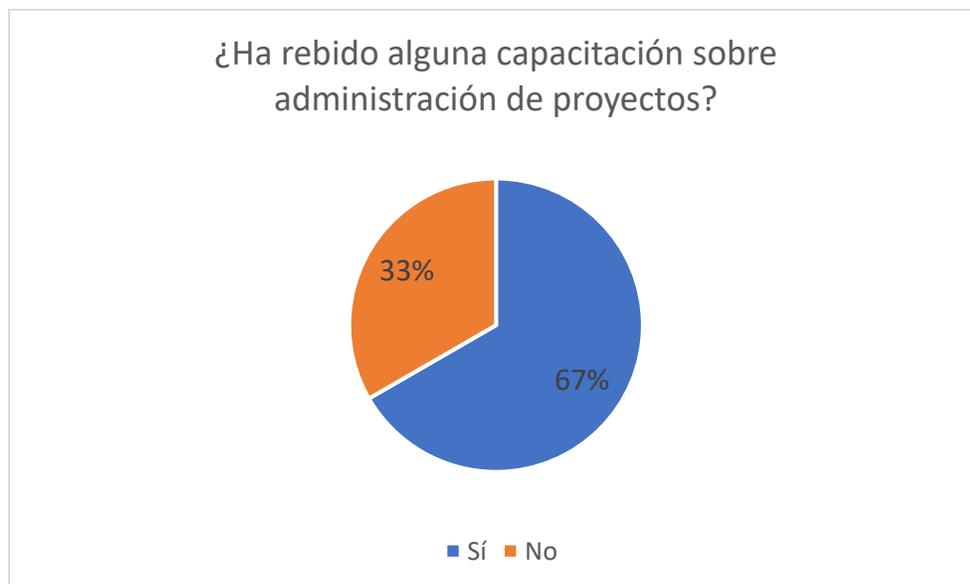
Nivel de conocimiento sobre administración de proyectos



Cuando se consultó si habían recibido alguna capacitación en administración de proyectos, un 67 % respondió afirmativamente, mientras que un 33 % respondió negativamente. De lo anterior se observa que un porcentaje importante de la muestra consultada posee conocimiento sobre este tema.

Gráfico 4

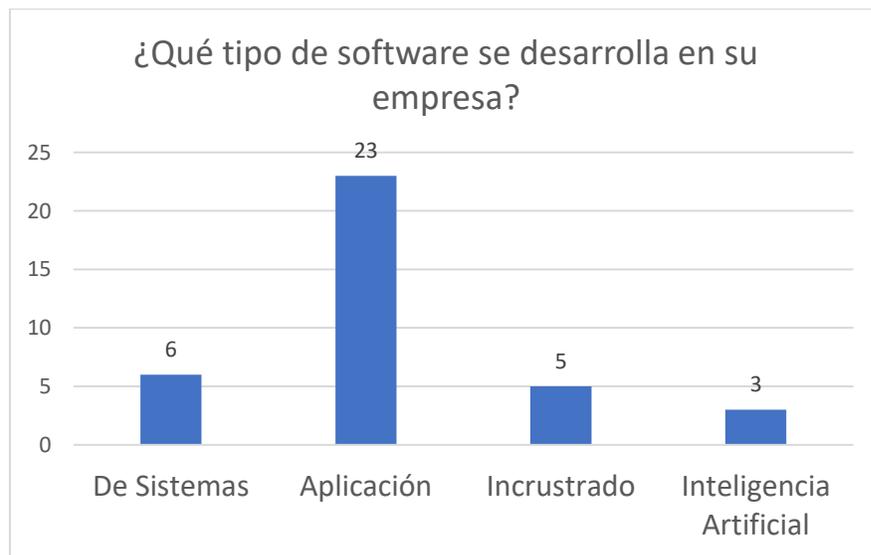
Capacitaciones sobre administración de proyectos



Por otra parte, el tipo de software que desarrollan las empresas que formaron parte del estudio es principalmente de aplicación para un 62 % de las respuestas. Otros tipos de software como el de sistemas, incrustado e inteligencia artificial, aunque es desarrollado por las compañías no predominaron en el estudio.

Gráfico 5

Tipo de software desarrollado por las empresas



Finalmente, cuando se consultó si la empresa utilizaba algún software para gestionar sus proyectos, los datos recolectados revelan que casi un 30 % de las empresas no utilizan algún software para gestionar sus proyectos, esta es la tendencia. Adicionalmente, de las compañías que sí utilizan un software, el que más se utiliza es Jira que corresponde al 25 % de las ocurrencias.

Tabla 15

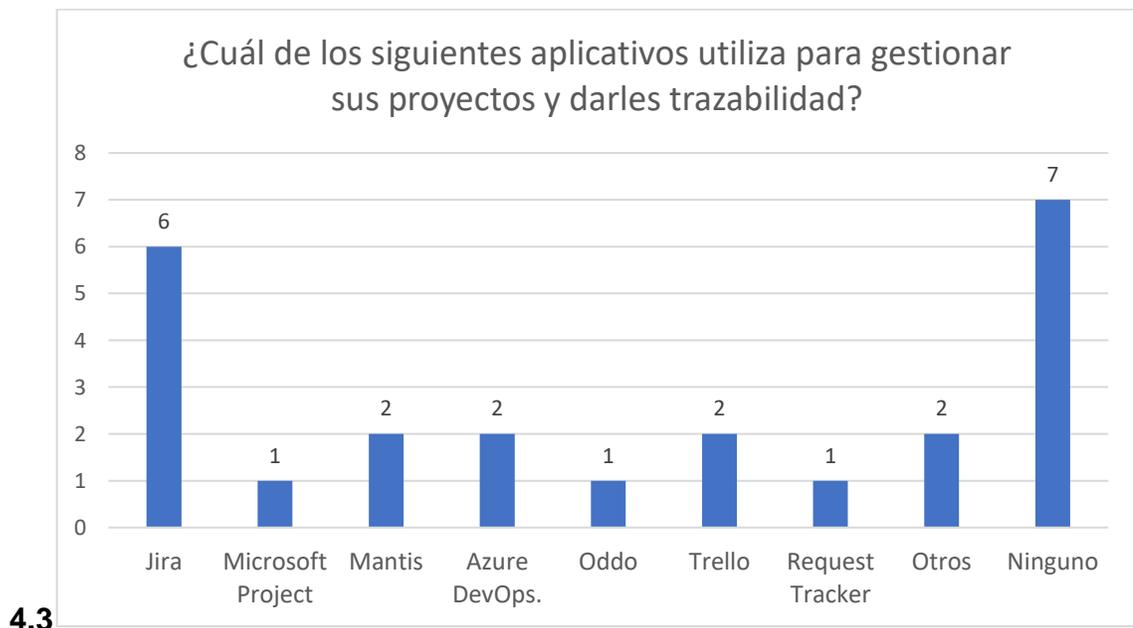
Software que se utiliza por las empresas para administrar sus proyectos

Software que se utiliza	Frecuencia absoluta
Jira	6
Microsoft Project	1
Mantis	2
Azure DevOps.	2
Oddo	1
Trello	2
Request Tracker	1
Otros	2
Ninguno	7
Total	24

Por otra parte, el Gráfico 6 muestra de manera visual la información expuesta en la Tabla 15:

Gráfico 6

Aplicativos para gestionar los proyectos



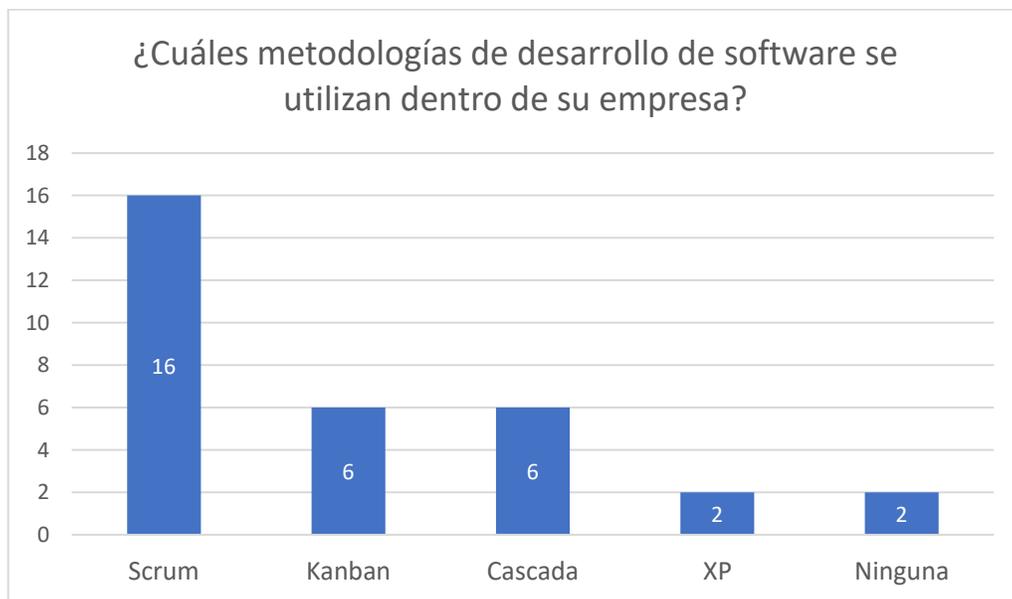
4.3

4.4 Metodologías que se utilizan

Respecto a las metodologías que se utilizan y flujos de proceso para desarrollar software, se encontró que un 66 % de las empresas consultadas utiliza metodología Scrum como metodología de trabajo. Esta es la que más predomina, lo que concuerda con la tendencia actual de implementación de metodologías ágiles en el mercado de desarrollo de software. Kanban y Cascada son otras metodologías y flujos de procesos que se utilizan con un 25 % de las respuestas obtenidas. Aquí se observa que la metodología cascada tradicional se utiliza por las compañías con un 18.75 % a pesar del auge de las metodologías ágiles.

Gráfico 7

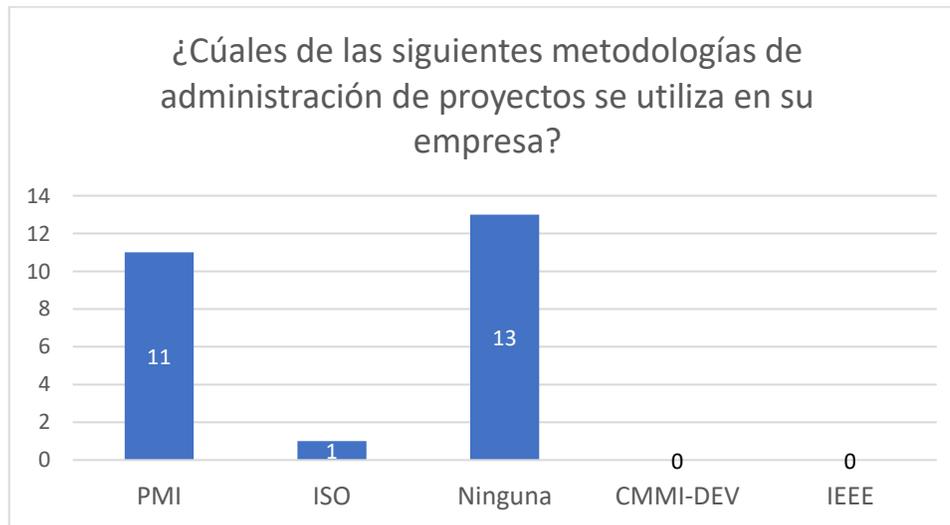
Metodologías de desarrollo en las compañías



Adicionalmente, respecto a las metodologías para gestionar los proyectos de software en las empresas, se encontró que un 54 % no utiliza metodologías para administrar sus proyectos, mientras que un 45 % usa PMI. La metodología menos utilizada es la ISO con un 8 %. De lo anterior se observa que la tendencia es que casi la mitad de todas las empresas consultadas no utilizan metodologías de administración de proyectos.

Gráfico 8

Metodologías de administración de proyectos que utilizan las empresas



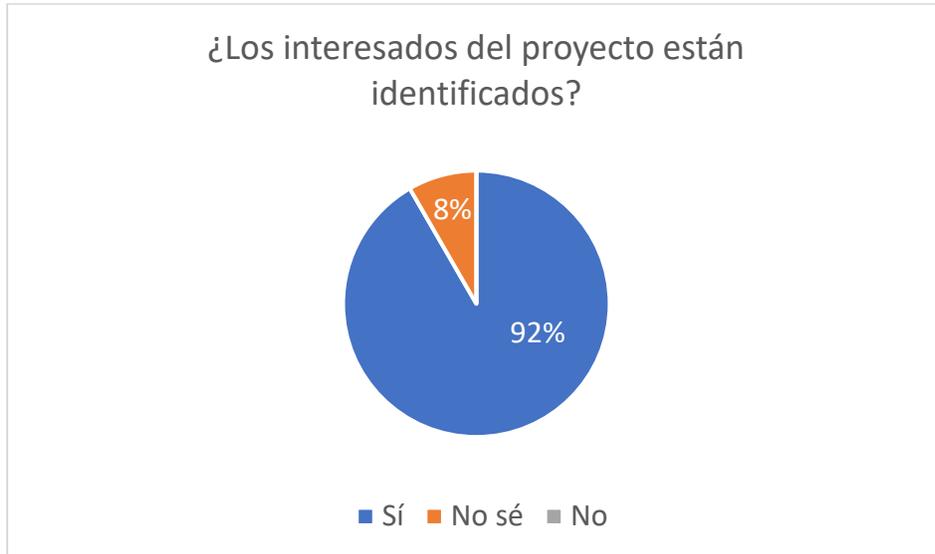
4.5 Alcance

El alcance de un proyecto corresponde a la determinación del resultado y de los objetivos que se deben cumplir para que el proyecto beneficie a los interesados. Por otro lado, los interesados corresponden a todos aquellos beneficiarios del proyecto, sean organizaciones, departamentos o usuarios externos.

En el momento de definir el alcance del proyecto, cuando se consultó sobre si los interesados del proyecto estaban identificados, un 92 % respondió afirmativamente, mientras que un 8 % no los identifica. Lo anterior muestra que la mayoría de las compañías define los interesados de sus proyectos.

Gráfico 9

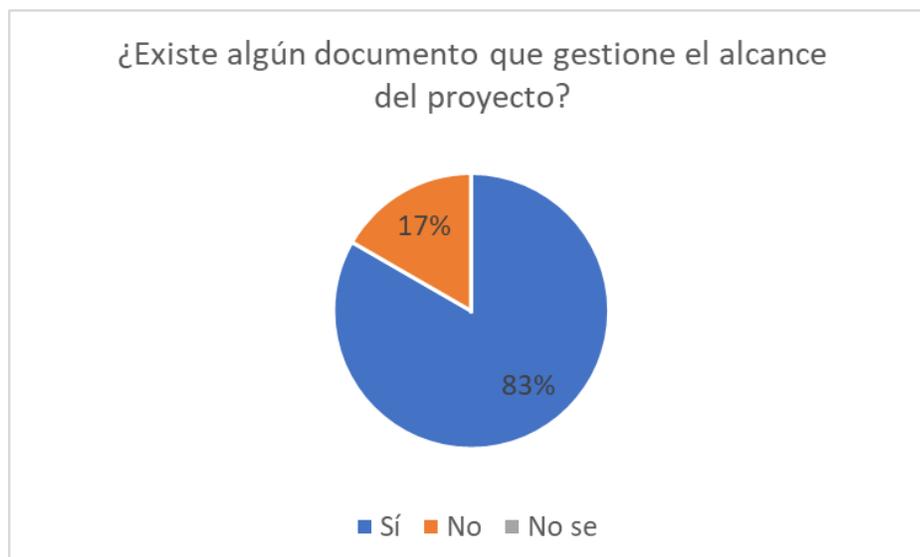
Interesados del proyecto



Por otra parte, se consultó a las empresas si existía algún documento que gestionara el alcance del proyecto para conocer si se tienen claro los resultados que se esperan de ese y los objetivos que deben cumplirse. Se encontró que un 83 % cuenta con un documento que gestione el alcance de los proyectos mientras que un 17 % no lo tiene.

Gráfico 10

Documentación del alcance del proyecto



En el momento de definir el alcance de los proyectos las empresas toman en cuenta varios criterios, los requerimientos, fechas de entrega, costos y entregables son los que más predominan. Otros criterios como las exclusiones y restricciones no se toman en cuenta por la mayoría de las compañías. La Tabla 16 muestra de manera detallada estos criterios y su frecuencia absoluta.

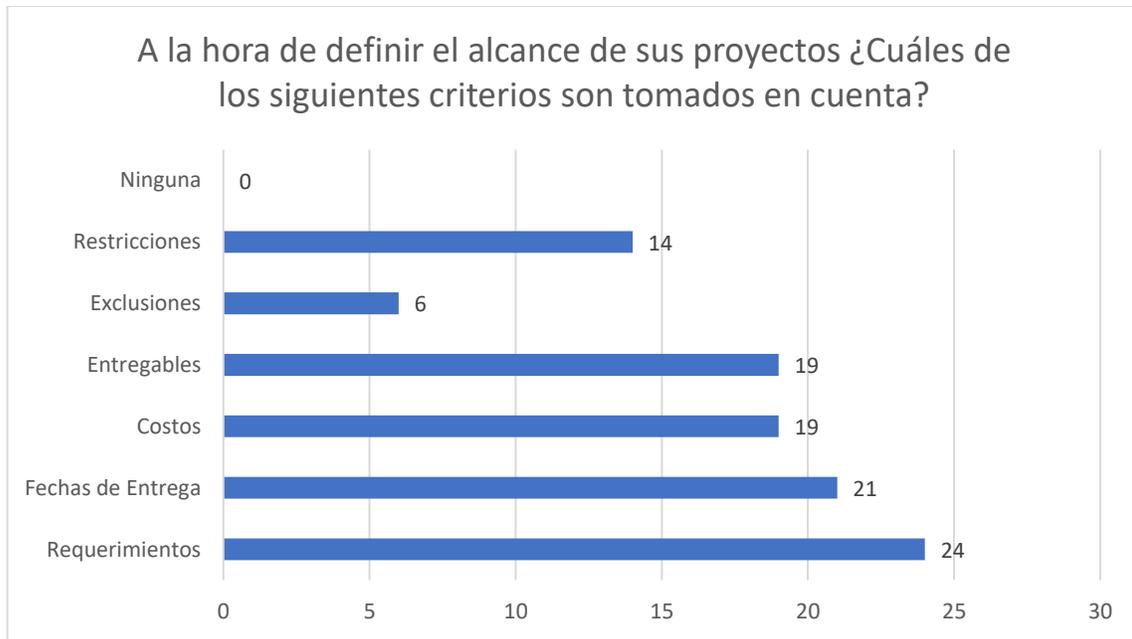
Tabla 16

Criterios que se utilizan en el momento de definir el alcance del proyecto

Criterios	Frecuencia absoluta
Requerimientos	24
Fechas de entrega	21
Costos	19
Entregables	19
Exclusiones	6
Restricciones	14
Ninguna	0
Total	103

Gráfico 11

Criterios que se utilizan en el momento de definir el alcance del proyecto



Por otra parte, cuando se consultó si los clientes se mantenían informados sobre los avances del proyecto. La mayoría respondió de forma afirmativa (96 %), mientras que únicamente una empresa respondió de forma negativa para un (4 %).

Gráfico 12

Avance entre los clientes del proyecto

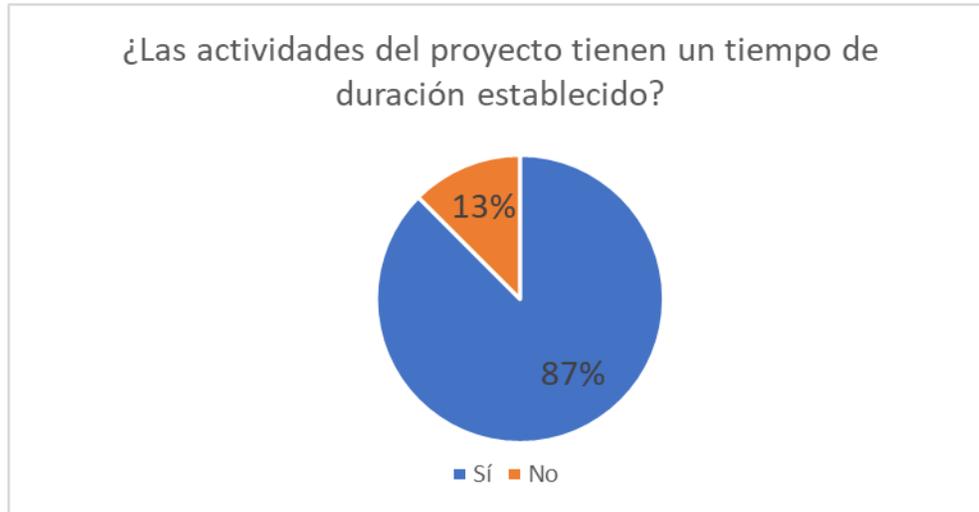


4.6 Tiempo

Cada proyecto tiene una serie de actividades que deben completarse, en ocasiones, deben cumplir con un tiempo determinado. Cuando se consultó sobre la duración de las actividades del proyecto, el 87 % de las empresas establece un tiempo determinado para la duración de sus actividades y un 13 % respondió de manera negativa. Lo anterior da a conocer que la mayoría de las compañías establece un tiempo para las actividades de sus proyectos.

Gráfico 13

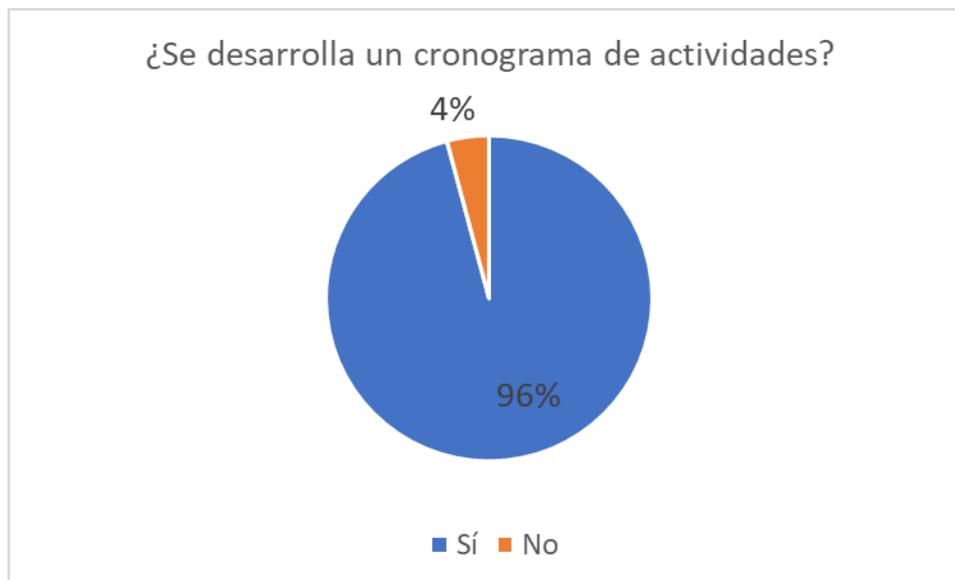
Duración de las actividades del proyecto



Por otra parte, el 96 % de las empresas encuestadas elabora un cronograma de actividades y un 4 % no lo realiza. Un cronograma permite al cliente estimar cuándo está listo el proyecto, de ahí su importancia.

Gráfico 14

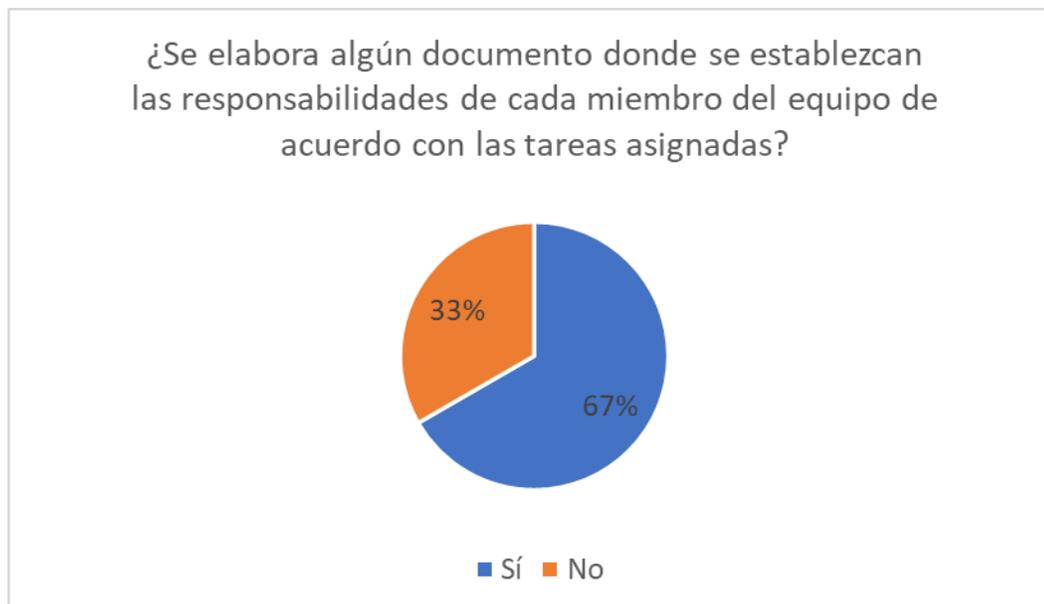
Cronograma de actividades del proyecto



Cuando se consultó si se elabora un documento que establezca las responsabilidades del equipo de trabajo de acuerdo con las tareas asignadas un 67 % respondió de forma positiva, mientras que un 33 % respondió de forma negativa.

Gráfico 15

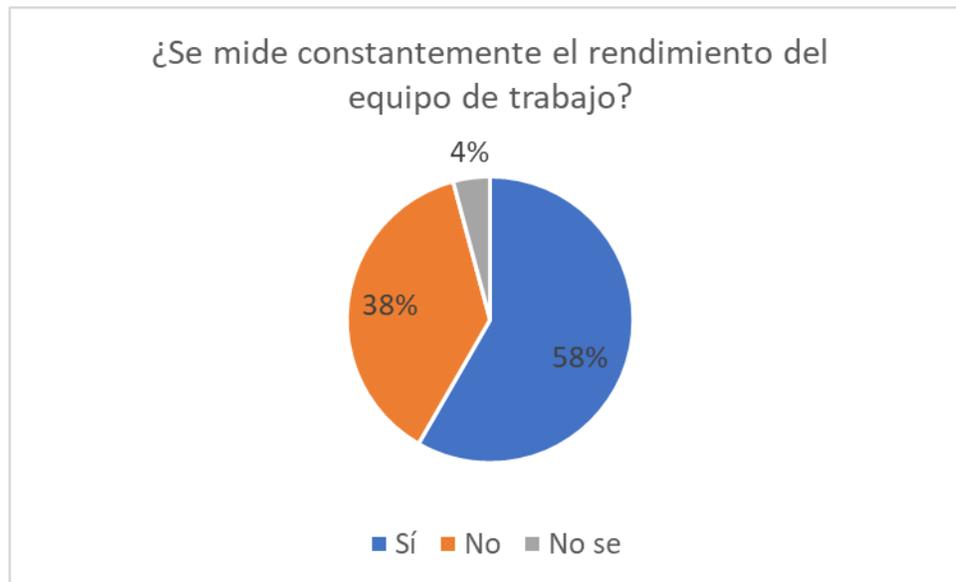
Responsabilidades de los miembros del equipo



Además, se determinó que un 58 % mide el rendimiento del equipo de trabajo mientras que un 38 % no lo mide. De lo anterior, se observa que casi cuatro de cada diez empresas no realizan este proceso.

Gráfico 16

Rendimiento de los miembros del equipo



Finalmente, sobre la forma en la que se estima el tiempo de duración de los proyectos se encontraron los siguientes criterios, los cuales se detallan en la Tabla 17.

Tabla 17

Técnicas de estimación del tiempo de las actividades de los proyectos

Criterios	Frecuencia absoluta
Puntos de Historia	10
Diagrama de PERT	1
Ninguna	3
Juicio de Experto	1
Total	15

Cabe mencionar que la técnica de estimación predominante son los **puntos de historia**. Esta técnica se relaciona con la metodología Scrum.

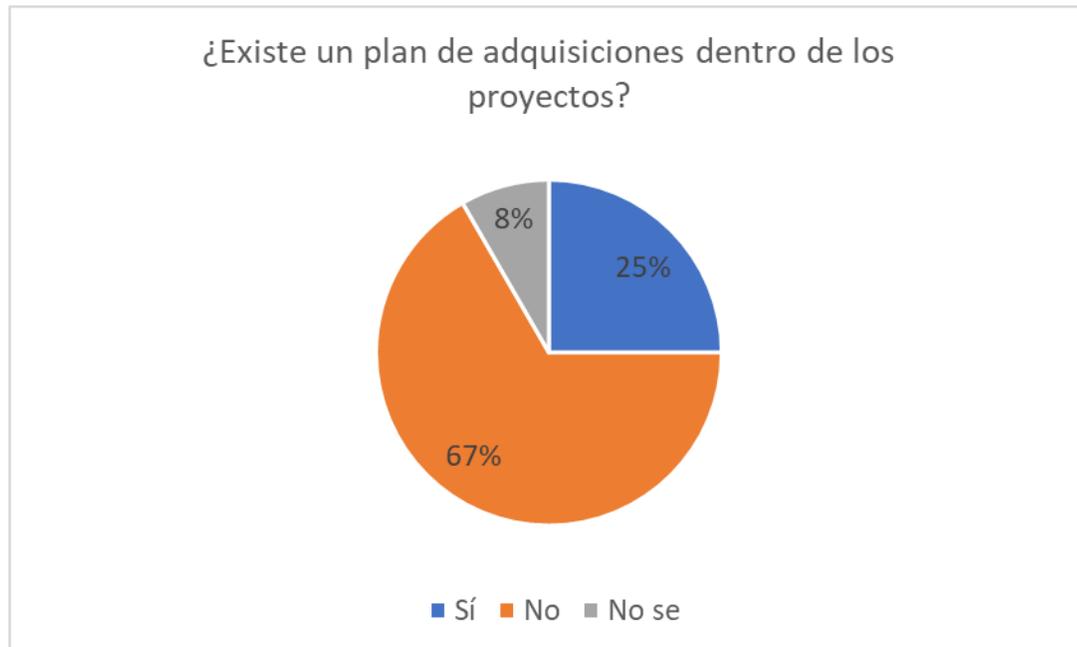
4.7 Costo y recursos

Todo proyecto tiene un costo que debe ser asumido por el cliente. Los costos de un proyecto de software incluyen el costo de los recursos humanos (hora hombre) y las adquisiciones de materiales físicos y digitales que debe realizar la empresa.

Sobre las adquisiciones del proyecto, se encontró que un 67 % de las empresas no realiza un plan de adquisiciones para los proyectos. Un 25 % lo lleva a cabo mientras que un 8 % desconoce si se realiza.

Gráfico 17

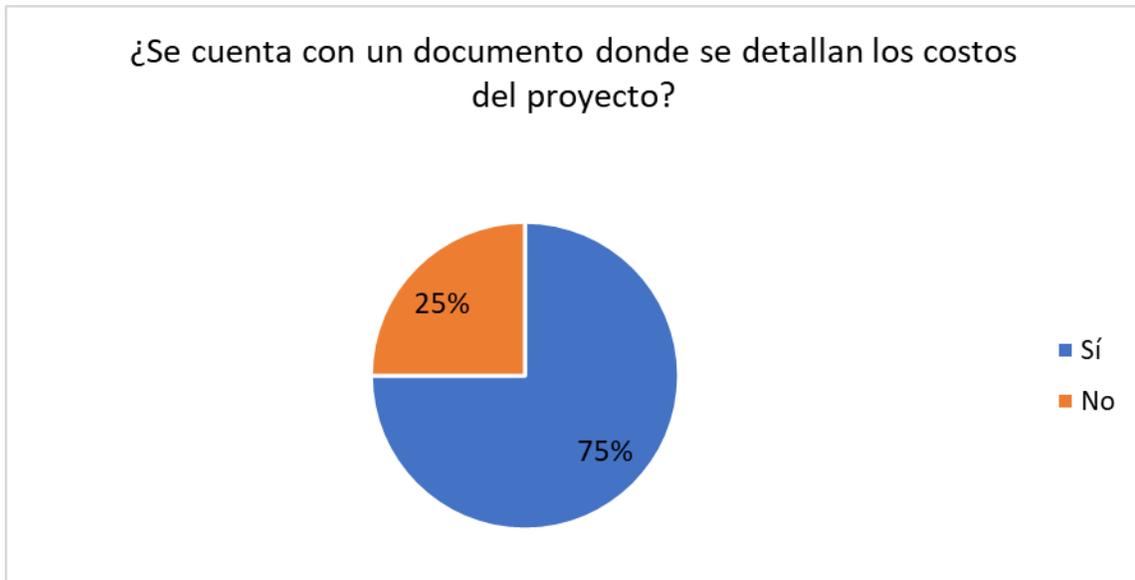
Existencia del plan de adquisiciones



Por otra parte, cuando se consultó si las empresas cuentan con un documento que detalle los costos del proyecto, un 75 % respondió afirmativamente, mientras que un 25 % respondió de forma negativa. En términos absolutos, casi 3 de cada 10 compañías no documentan los costos de su proyecto, por lo que se desconoce la manera en la que estas realizan la monetización.

Gráfico 18

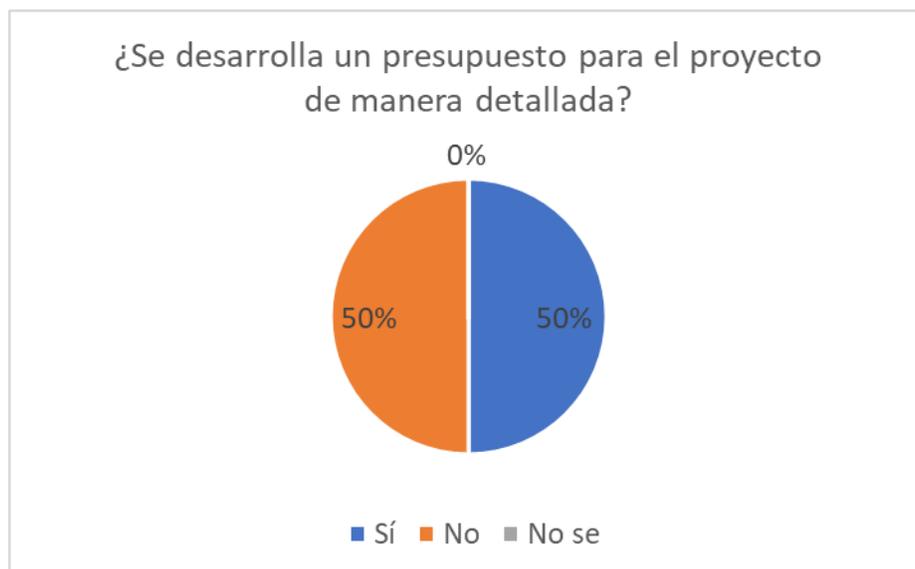
Existencia del plan de costos



Además, un 50 % elabora un presupuesto detallado para el proyecto, mientras que de forma similar un 50 % no lo elabora. Se observa como la mitad de las empresas no elabora presupuestos para sus proyectos.

Gráfico 19

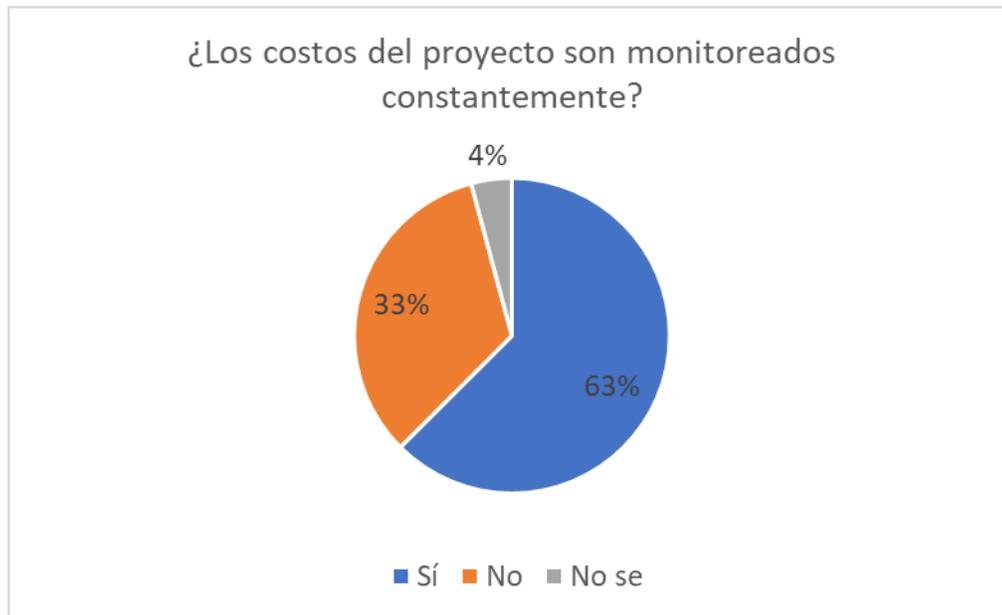
Elaboración del presupuesto



Finalmente, sobre el monitoreo de los costos del proyecto se encontró que la mayoría de las empresas realiza un monitoreo constante de sus costos (63 %), mientras que una parte considerable no los monitorea (33 %).

Gráfico 20

Monitoreo de los costos



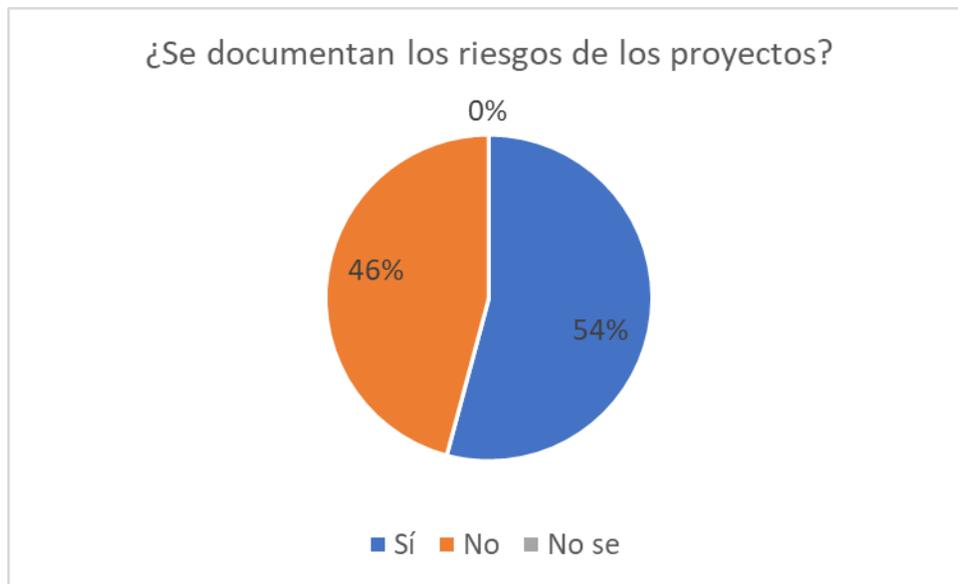
4.8 Riesgos

Otro aspecto relevante en la administración de proyectos de software es la gestión de los riesgos. La mayoría de los proyectos tiene una serie de riesgos que deben gestionarse para evitar que cuando se materialicen causen pérdidas a la empresa.

Al consultar a las empresas si documentaban sus riesgos, la mitad afirmó que se documentan, mientras que la otra mitad no los documenta. De lo anterior se observa que la documentación de los riesgos no es una prioridad que cuatro de cada diez compañías tomen en cuenta en su proceso de administración de proyectos.

Gráfico 21

Documentación de los riesgos



Se encontró que la mayoría de las empresas (54 %) les da seguimiento a los riesgos del proyecto, contra un 46 % que no lo hace. Lo anterior muestra que, aunque las compañías no documentan los riesgos, se les da seguimiento.

Gráfico 22

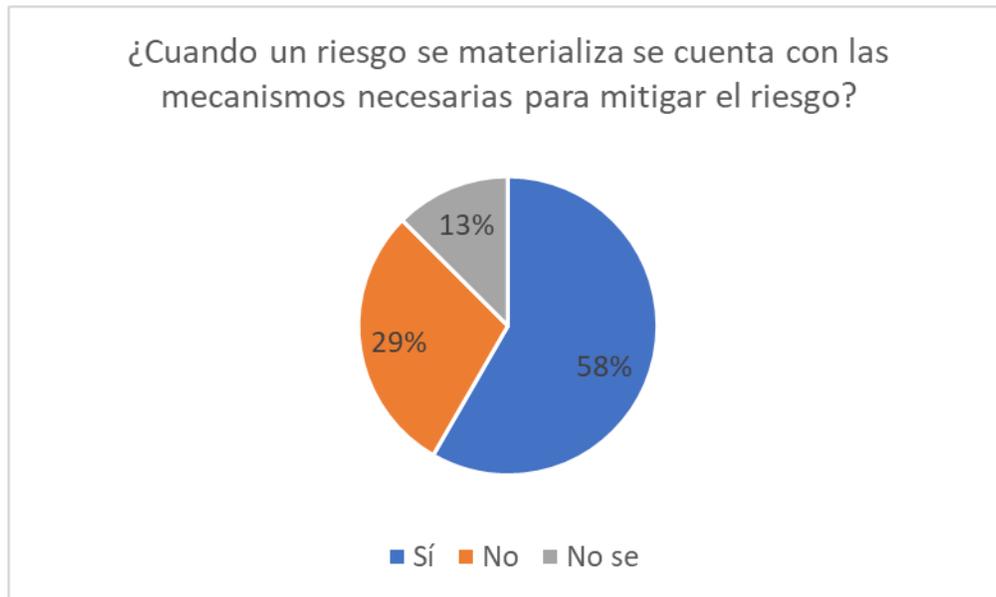
Monitoreo de los riesgos



Por otra parte, al consultar si al materializarse un riesgo se contaba con los mecanismos necesarios para mitigarlo, un 58 % respondió de manera afirmativa, mientras que un 29 % indica no mitigar los riesgos y un 13 % no sabe. Con lo anterior se observa que se encuentra una parte de las empresas que sí mitigan sus riesgos, pero también existe una cantidad considerable que no lo hace.

Gráfico 23

Mitigación del riesgo

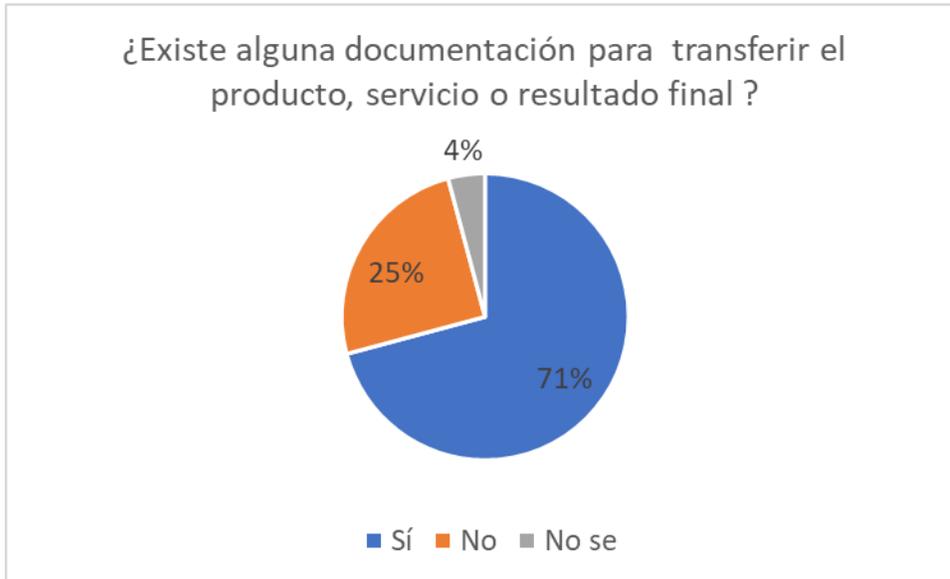


4.9 Cierre

El proceso de cierre corresponde a la etapa donde se entrega el producto final, se liberan los recursos que se utilizan y se finiquita el contrato en los casos en los que aplique. Cuando se consultó sobre el proceso de cierre de los proyectos en las empresas, el 71 % respondió que sí existe documentación para transferir el producto, servicio o resultado, mientras que el 25 % no realiza ese tipo de documentación y un 4 % indica no saber.

Gráfico 24

Documentación del cierre del proyecto



Otro aspecto importante en el momento de cerrar los proyectos son las lecciones aprendidas. Las lecciones aprendidas son experiencias negativas o positivas que sucedieron durante el proyecto. Estas experiencias deben documentarse para futuros proyectos. En el caso de las empresas consultadas una mayoría no las documenta con un 58 %, mientras que un 42 % si realiza este proceso al finalizar el proyecto.

Gráfico 25

Documentación de las lecciones aprendidas



4.10 Calidad

Respecto a la gestión de calidad del proyecto, se determinó que la mayoría de las empresas (58 %) cuenta con un procedimiento que abarque este tema, mientras que un 42 % no cuenta con este procedimiento. De lo anterior se observa que existe un porcentaje importante que no establece procedimientos de calidad para los productos, a pesar de que la mayoría sí los establece.

Gráfico 26

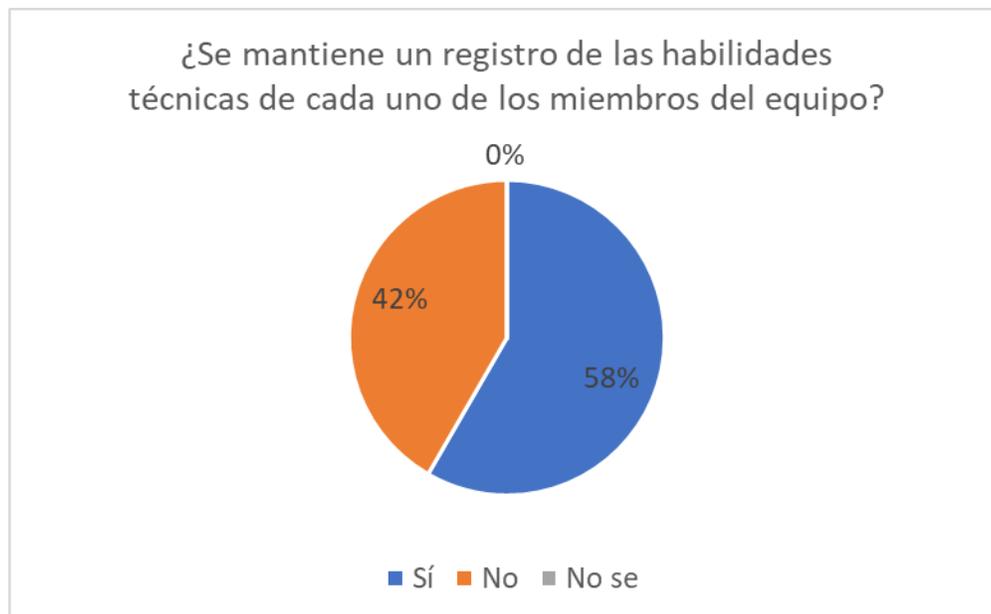
Procedimiento de calidad para los productos



Por otra parte, cuando se consultó si las empresas mantenían un registro de las habilidades técnicas de los miembros del equipo, la mayoría respondió de manera afirmativa (58 %), mientras que un porcentaje importante respondió negativamente (42 %).

Gráfico 27

Registro de habilidades técnicas de los miembros del equipo



La mayoría de las empresas prueba constantemente la calidad de los productos que desarrolla (75 %), mientras que un 25 % afirma que la calidad de los productos no se prueba.

Gráfico 28

Evaluación de calidad de los productos



Finalmente, cuando se consultó sobre un informe de calidad al final del proyecto la mayoría de las empresas afirmó que no lo realiza para un 83 %. Solamente el 17 % de la muestra encuestada lo lleva a cabo. Este último elemento es importante que se desarrolle debido a que el cliente debe tener la certeza de que el producto pasó por un proceso de calidad riguroso.

Gráfico 29

Informe de calidad del proyecto



CAPÍTULO V:
CONCLUSIONES Y RECOMENDACIONES

5 Conclusiones y recomendaciones

En este capítulo se presentan algunas conclusiones y recomendaciones tomando como referencia la información expuesta en el análisis situacional, marco metodológico y teórico. Las conclusiones y recomendaciones están orientadas a responder a los objetivos de la investigación.

5.1 Objetivo específico n.º 1: Conocer la forma en la que se gestionan los proyectos en las mypes del país que desarrollan proyectos de software mediante consultas a las partes interesadas

5.1.1 Conclusiones

- Sobre la gestión de proyectos en este tipo de empresas se concluye que la mayoría de ellas gestiona el alcance, interesados, costo y tiempo, las cuales constituyen áreas de gestión importantes en esta disciplina. Sin embargo, se encontró un alto porcentaje de compañías que no gestiona los riesgos, adquisiciones y cierre de los proyectos. Lo anterior puede ocurrir debido a que las empresas pueden carecer de las herramientas adecuadas para realizar estos procesos. Otra posibilidad es que las empresas consideran que estos procesos no son necesarios porque nunca los han realizado y les ha funcionado bien de esa manera.
- Aunque el rendimiento del equipo de trabajo se mide por la mayoría de las empresas, existe un gran porcentaje que no lo mide (38 %), a pesar de que un alto porcentaje utiliza metodologías ágiles de desarrollo (66 %) que se enfocan en aumentar la productividad del equipo de trabajo para que se entreguen productos con mayor eficiencia. Las compañías que lo miden pueden saber cómo se comporta el equipo para conocer las áreas que deben mejorarse. Por otra parte, las empresas que no miden pueden carecer del conocimiento necesario para establecer planes de mejora. El consultor Peter Drucker mencionó que lo que no se mide no se controla,

lo que no se controla no se mejora y lo que no se mejora se deteriora. En ambos casos cuando el líder de proyecto conoce cómo se desempeña el equipo puede tomar decisiones, como disminuir la carga de trabajo, realizar actividades de recreación para fomentar la unión del equipo o invertir mayores recursos en capacitaciones.

- La mayoría de las empresas prueba la calidad de sus productos, al menos la mitad no realiza este proceso y la mayoría no lleva a cabo informes de calidad de los productos que se entregan a los clientes finales. Las compañías que no realizan control de calidad de los productos que entregan pueden correr el riesgo de que estos contengan errores que conduzcan a defectos y estos a fallos que provoquen efectos negativos. Por otra parte, al no desarrollar informes de calidad los clientes no tienen la certeza sobre el proceso de calidad por el que pasó el producto y, de esta manera, el cliente no tendría certeza si el producto se desarrolló siguiendo estándares de calidad.
- Sobre el cierre de los proyectos, aunque la mayoría de las empresas cuenta con los procedimientos para entregar el proyecto o servicio al cliente, la mayoría no documenta las lecciones aprendidas. Estas últimas son importantes para futuros proyectos o etapas de este debido a que se pueden reutilizar las experiencias en otras etapas o en proyectos nuevos. Por otra parte, las compañías que no cuentan con procedimientos para entregar el producto final cuando el cliente finiquite el contrato con la empresa y esta última requiere hacer modificaciones al sistema pueden carecer de la información necesaria para realizar estos cambios.

5.1.2 Recomendaciones

- Las empresas que no gestionan áreas como alcance, interesados, costo y tiempo deben evaluar si establecer una estrategia para gestionar estas áreas aportaría valor a su proceso o complicaría su quehacer diario. Para gestionar estas áreas se recomienda que la empresa tome como referencia los métodos, técnicas y herramientas establecidas por los estándares de administración de proyectos que más se ajusten a sus necesidades y poco a poco implementarlas con los recursos que estén disponibles en el momento. Si lo implementado no da resultados la compañía tiene la libertad de eliminarlo.
- Debido a que el factor humano es importante para los proyectos, las empresas que utilizan Scrum (la mayoría) pueden sacar provecho de las herramientas que tiene este marco de trabajo como el Gráfico de Quemado, Team Velocity o simplemente la cantidad de unidades de trabajo realizadas en un tiempo determinado. La mayoría de aplicativos de gestión de proyectos implementan estas funcionalidades por lo que se recomienda revisar si estos las incluyen. Las compañías que implementan un enfoque de desarrollo más tradicional pueden realizar reuniones periódicas para consultar su progreso y si hay algún bloqueo que impida completar el trabajo asignado.
- Sobre la calidad de los productos se sugiere que las empresas implementen procesos de calidad, al menos básicos. En la medida de lo posible una persona del equipo de desarrollo debe tomar la función de probador (tester) para que pruebe los tiquetes desarrollados por los programadores y verifique que el trabajo realizado sea el correcto de acuerdo con el requerimiento (pruebas de verificación), después presentar los cambios al cliente y validar que el trabajo realizado satisface sus necesidades (pruebas de validación). Otro tipo de

pruebas más especializadas pueden hacerse, pero quedan sujetas al conocimiento que posean los miembros del equipo.

- Respecto al cierre de los proyectos se recomienda que las empresas valoren la posibilidad de entregar un documento donde se le informe al cliente que se aceptan los entregables y realizar la entrega de toda la documentación al cliente para que este lo opere con éxito. Por otra parte, a lo interno de la empresa se pueden analizar y documentar cuales fueron las lecciones aprendidas para utilizarlas en futuros proyectos.

5.2 Objetivo específico n.º 2: Identificar los estándares que más se utilizan por las empresas durante su proceso de administración de proyectos de software mediante la recopilación de datos

5.2.1 Conclusiones

- Sobre los estándares que utilizan las empresas se concluye que el estándar de PMI es uno de los predominantes. Sin embargo, es importante destacar que la mitad de la muestra consultada no utiliza ningún estándar para administrar sus proyectos. Al no contar con metodologías para administrar este tipo de proyectos, la empresa administra con base en el juicio de experto. Al utilizar el juicio de experto la compañía usa su propio criterio para administrar sus proyectos en lugar de utilizar un estándar. La consecuencia es que la empresa puede carecer de los métodos, técnicas y herramientas necesarias para administrar de manera adecuada sus proyectos, por lo que eventualmente incurriría en errores de administración. A pesar de que la mayoría de las empresas no utiliza estándares de gestión de proyectos, un alto porcentaje posee colaboradores que tienen un conocimiento medio sobre esta disciplina. Al conocer sobre

administración de proyectos las compañías pueden adoptar un estándar de gestión de proyectos.

5.2.2 Recomendaciones

- Se recomienda que la empresa evalúe si el juicio de experto es la forma más adecuada de gestionar los proyectos. Si el cliente y el equipo se sienten satisfechos con el proceso que utiliza la compañía entonces se debe mejorar continuamente para asegurar que se siga llevando a cabo con calidad. De igual manera, las empresas que sí utilizan estándares de administración de proyectos deben evaluar los métodos, técnicas y herramientas que se proponen en el próximo capítulo y adaptar aquellos dominios de desempeño que deben optimizar o que desean implementar.

5.3 Objetivo específico n.º 3: Determinar el estándar de administración de proyectos más adecuado para este tipo de empresas mediante el análisis de la información que se recopiló

5.3.1 Conclusiones

- Debido a que la mayoría de las empresas implementan Scrum (66 %) se concluye que el estándar que más puede adecuarse a este tipo de compañías es la Guía de los Fundamentos para la Dirección de Proyectos y el Estándar para la Dirección de Proyectos, séptima edición. Este estándar incluye un enfoque adaptativo (ágil) para administrar proyectos y se enfoca en dar valor a los clientes a través de diferentes áreas de dominio que se explicaron en el Capítulo II. Además, este estándar posee una extensión dedicada exclusivamente a los proyectos de software denominada Extensión de software a la guía PMBOK. Las

empresas que no implementan Scrum, de igual manera, pueden implementar este estándar debido a que es genérico, es decir, no importa la metodología de desarrollo que se utilice, se adapta a cualquier empresa.

5.3.2 Recomendaciones

- Se sugiere que las empresas evalúen si adoptar un estándar de administración de proyectos aportaría valor a la organización debido a que el objetivo de un estándar es sugerir una serie de buenas prácticas para gestionar proyectos. Sin embargo, algunas de estas compañías pueden tener limitaciones de recursos para implementar un estándar en concreto ya que se requiere invertir recursos humanos y económicos. La empresa debe evaluar el costo-beneficio de adoptar un estándar en particular y si su implementación aportaría valor en lugar de complicar su quehacer diario.

CAPÍTULO VI:
PROPUESTA

**GUÍA PARA LA ADMINISTRACIÓN DE PROYECTOS
PARA LAS MYPES DESARROLLADORAS DE
SOFTWARE DE LA GRAN ÁREA METROPOLITANA DE
COSTA RICA BASADA EN EL PMBOK SÉTIMA EDICIÓN**

Elaborado por Daniel Bello Ramírez
Fecha: mayo de 2023
Versión: 1

6 Propuesta

6.1 Objetivo

El objetivo de la presente propuesta es proporcionar una guía de administración de proyectos que incluye una serie de procedimientos, artefactos y herramientas que las empresas mypes pueden considerar cuando se administran proyectos de software utilizando las mejores prácticas. Esta propuesta se basa principalmente en la Guía de los Fundamentos para la Dirección de Proyectos PMBOK séptima edición, donde se brinda una serie de buenas prácticas para el proceso de administración de proyectos de cualquier disciplina.

6.2 Alcance

Esta propuesta presenta algunos de los procesos sugeridos en el PMBOK para administrar proyectos de software utilizando mejores prácticas. Lo establecido en el PMBOK es un modelo de referencia, su implementación total no es obligatoria debido a que no es una metodología. El estándar identifica los procesos que se consideran buenas prácticas en la mayoría de los proyectos, la mayoría de las veces. El estándar también identifica las entradas y salidas que generalmente se asocian con esos procesos, no exige llevar a cabo ningún proceso o práctica particular (PMI, 2017).

Cada proyecto y empresa es diferente. Ningún proyecto es igual a otro, por lo que la propuesta aquí presentada incluye una serie de recomendaciones que pueden implementarse de manera **opcional** para cualquier tipo de proyecto de software tomando como referencia la guía PMBOK y las metodologías ágiles que estas empresas utilizan de acuerdo con la información presentada en el análisis situacional.

Por otra parte, con la presente propuesta se pretende que las empresas adopten una serie de buenas prácticas en su proceso de administración de proyectos. Cada empresa debe analizar si la propuesta presentada en este capítulo agrega valor a su proceso y adoptar solo lo que se considere necesario, debido a que en general este tipo de compañías poseen limitaciones de recursos.

6.3 Introducción

El estándar para la dirección de proyectos de la guía PMBOK incluye un conjunto de buenas prácticas para la dirección de proyectos de cualquier industria, independientemente del enfoque de desarrollo de producto que se utiliza, predictivo, adaptativo o híbrido (combinación de ambas).

Sobre las buenas prácticas según menciona PMI (2017):

Significa que existe consenso general acerca de que la aplicación de conocimientos, habilidades, herramientas y técnicas a los procesos de dirección de proyectos puede aumentar la posibilidad de éxito de una amplia variedad de proyectos para entregar los resultados y los valores del negocio esperados (p. 41).

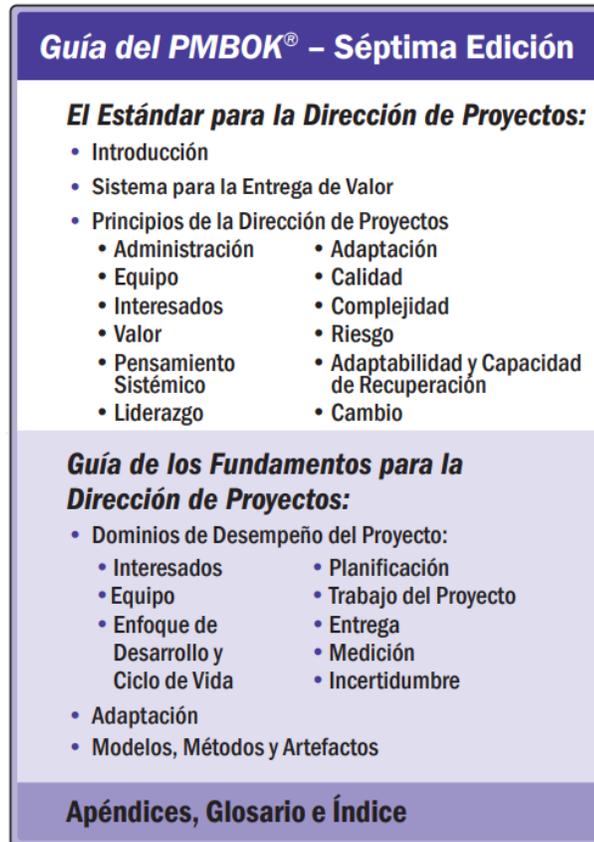
Por otra parte, es importante mencionar que la séptima edición de la guía PMBOK a diferencia de ediciones anteriores se enfoca principalmente en la **entrega de valor**. PMI (2021) menciona que:

Un enfoque de los sistemas hacia la entrega de valor cambia la perspectiva de dirigir portafolios, programas y proyectos, para centrarse en la cadena de valor que vincula esas y otras capacidades empresariales para avanzar en la estrategia organizativa, el valor y los objetivos del negocio (p. 11).

Para aportar valor a las organizaciones mediante la administración de proyectos, la guía PMBOK presenta una serie de principios que se deben cumplir para lograr este objetivo y la consecución de estos principios **guiará** los **dominios de desempeño** del proyecto. La presente guía contiene una serie de prácticas para seguir para cada dominio de desempeño del PMBOK aterrizadas a una pequeña empresa que administra proyectos de software con el objetivo de entregar **valor**.

Figura 23

Estructura de la guía PMBOK – Séptima Edición



Nota. Tomado de *Guía de los Fundamentos para la Dirección de proyectos y el Estándar para la Dirección de proyectos*, por Project Management Institute, 2021.

6.4 Entrega de valor en los proyectos de software

Todo proyecto de cualquier disciplina es un esfuerzo temporal que se realiza para crear un servicio, producto o resultado que debe entregar valor a los interesados. Al respecto PMI (2010) menciona que:

El valor de un proyecto se define por el valor que un proyecto crea para sus partes interesadas. El valor del proyecto podría estar representado por una sola o cualquier combinación de eficiencia, eficacia técnica y la

satisfacción de las partes interesadas de un proyecto con énfasis en los clientes y accionistas (s. p).

Es importante mencionar que la entrega de valor en los proyectos puede afectarse por el entorno **interno** (en la organización) y **externo** (fuera de la entidad) de la organización que las mypes deben tomar en cuenta. Algunos ejemplos de factores internos que pueden tener incidencia sobre los proyectos son los siguientes según la guía PMBOK:

- Seguridad: procedimientos de seguridad existentes o inexistentes en la organización que pueden afectar el proyecto.
- Infraestructura: instalaciones que existen en la empresa para el desarrollo de proyectos.
- Software informático: software informático para desarrollar los proyectos.
- Disponibilidad de recursos: disponibilidad de recursos para adquirir activos para la organización.
- Capacidad de los empleados: corresponde a las habilidades y conocimiento de los empleados que llevarán a cabo el proyecto.

Algunos factores externos que pueden afectar la ejecución y los resultados del proyecto son los siguientes:

- Condiciones del mercado: condiciones principalmente económicas dentro de un momento debido a que pueden influir con la ejecución del proyecto.
- Entorno regulatorio: leyes, regulaciones que deben tomarse en cuenta en el proyecto.

Para que un proyecto de software entregue valor a los interesados existen diferentes maneras de lograrlo. La más importante es hacer que el producto que se entrega como resultado del proyecto contribuya con satisfacer la necesidad del cliente. Lo anterior se logra con compromiso y centrándose en **conocer su verdadera necesidad** para resolver su problema, debido a que todo proyecto busca satisfacer una necesidad tomando en cuenta los factores internos y

externos involucrados. Los miembros del equipo del proyecto deben tener consciencia de la misión del cliente para trabajar todos en conjunto para entregar valor.

Para conocer la necesidad de un cliente el equipo de trabajo debe trabajar en conjunto para obtener un mejor entendimiento del problema y obtener retroalimentación cada vez que un entregable del proyecto se entregue.

El factor humano es fundamental en el momento de llevar a cabo un proyecto. Las personas colaboradoras de la empresa también deben ser una prioridad y la compañía debe invertir en ellas cuando sea posible, atrayendo a los mejores talentos y manteniéndolos motivados. El equipo debe ser dirigido por una persona que sea diligente y tolerante para que el proyecto se adapte a los cambios constantes que suceden a menudo en este tipo de proyectos y también de comunicarse efectivamente con el cliente.

Para que el proyecto se lleve a cabo de una manera exitosa la guía PMBOK propone una serie de principios que deben tomarse en cuenta en todo proyecto. Estos principios se explicaron de manera breve en el marco teórico.

6.5 Dominios de desempeño del proyecto y su implementación

Los dominios de desempeño que menciona la guía PMBOK séptima edición corresponden a una serie de actividades fundamentales en los proyectos. Estos dominios de desempeño deben trabajar en conjunto para obtener los resultados que desean. Los dominios de desempeño del PMBOK y su implementación (procesos y herramientas por utilizar) en una mypes se mencionan a continuación en formato de guía.

La guía que se presenta a continuación presenta una serie de procesos que deben seguir las empresas para implementar un proceso de gestión de proyectos que se basa en el PMBOK. Además, la guía presenta herramientas o artefactos que las compañías pueden utilizar para cada dominio de desempeño. Es importante mencionar que la mayoría de artefactos presentados están presentes en el PMBOK, sin embargo, existen otros que son específicos al

desarrollo de software y no aparecen en este. Por otra parte, se presenta una serie de criterios de verificación para que las empresas verifiquen que cumplen con cada dominio de desempeño. La mayoría de los procesos presentados en la guía deben hacerse por el administrador de los proyectos de la empresa o quien tenga un rol similar.

6.5.1 Interesados

Los interesados son grupo u organización que puede afectar, verse afectado o percibirse a sí mismo como afectado por una decisión, actividad o resultado de un proyecto, programa o portafolio (PMI, 2021, p.8)

Un proyecto puede tener un pequeño grupo de interesados o, potencialmente, millones de interesados. Puede haber diferentes interesados en diferentes fases del proyecto y la influencia, el poder o los intereses de estos pueden cambiar a medida que se desarrolla el proyecto (PMI, 2021). Para gestionar los interesados del proyecto siga los siguientes pasos:

Procedimientos por realizar:

1. **Identificar a los interesados:** Identifique a cada persona interesada a la que afecta el proyecto. El interesado principal es el cliente, pero existen otros que deben tomarse en cuenta como inversionistas, proveedores y empleados.
2. **Analizar a los interesados:** Analice los principales requerimientos, impacto e influencias que tienen cada una de las personas interesadas. Estos datos pueden variar a lo largo del proyecto.
3. **Priorizar a los interesados:** Priorice a las personas interesadas para que el equipo de proyecto se involucre con los interesados que posean mayor poder, interés e influencia sobre el proyecto. Para priorizar a los interesados utilice la herramienta matriz poder-influencia que se explica a continuación.

4. **Involucrar a los interesados:** Involucre a las personas interesadas durante todo el proyecto. Los interesados con mayor poder e influencia deben invitarse a participar de las diferentes reuniones que haga el equipo de proyecto, especialmente en las etapas iniciales. Esto se debe a que la mayoría de las empresas consultadas utiliza metodología Scrum. Las personas interesadas pueden invitarse a las diferentes ceremonias que se celebran en Scrum, por ejemplo:

- Planificación del sprint.
- Revisión del sprint.
- Retrospectiva del sprint.
- Reuniones diarias (daily stand-up)

El plan de comunicaciones es un artefacto que se utiliza para planificar las comunicaciones entre los interesados del proyecto. Este artefacto se explica más adelante.

Artefactos/herramientas por utilizar: Los artefactos y herramientas para gestionar a los interesados son los siguientes:

- **Plan de involucramiento de los interesados:** Documente a cada una de las personas interesadas del proyecto, una vez identificados, analizados y priorizados. Para priorizarlos tome como referencia el artefacto matriz poder/influencia.

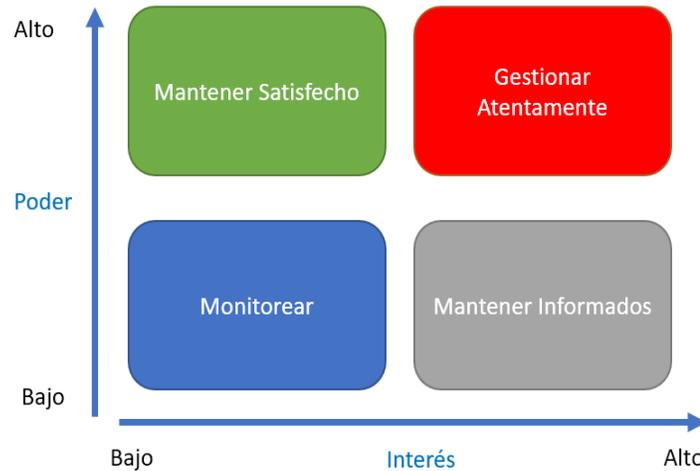
Registro de interesados					
Código	Nombre	Puesto	Información de contacto	Tipo de interesado (interno o externo)	Prioridad

- **Matriz poder influencia:** La matriz poder-influencia permite determinar el poder y la influencia de cada persona interesada para determinar su

prioridad en el involucramiento. Elabore una matriz poder-influencia, de acuerdo con el siguiente ejemplo:

Figura 24

Matriz poder-influencia



La matriz anterior puede interpretarse de la siguiente manera:

- Si la persona interesada tiene alto poder y alto interés (cuadrante rojo) la empresa debe mantenerlos involucrados activamente en todas las etapas del proyecto de manera tal que se mantengan satisfechos.
- Si la persona interesada tiene mucho poder, pero bajo interés (cuadrante verde) se deben mantener satisfechos con los avances del proyecto y mantenerlos informados en todo momento para que este no afecte negativamente al proyecto. Es decir, que limite los recursos que se le asignan al proyecto por el poco interés que tiene esta persona.
- Si la persona interesada tiene bajo poder y bajo interés (cuadrante azul) solamente se debe monitorear ocasionalmente, debido a que existe poca probabilidad de que la persona interesada afecte de forma negativa al proyecto debido a su bajo poder y poco interés.
- Si la persona interesada tiene poco poder, pero alto interés (cuadrante gris) la retroalimentación que proporciona en el proyecto puede ser muy valiosa, sin embargo, debido a su poco poder, casi no influye sobre las

decisiones que se toman en el proyecto. En este caso es mejor mantenerlos informados.

- **Plan de comunicaciones:** Esta herramienta permite definir cómo se realiza la comunicación entre los interesados del proyecto, tanto internos como externos. En esta herramienta documente las comunicaciones que se llevan a cabo en el proyecto y hacia quién están dirigidas. En esta herramienta puede colocar los siguientes criterios:
 - **Receptor:** Hacia quién está dirigida la comunicación.
 - **Medio de comunicación:** Canal que se utiliza para manejar la comunicación: en persona *e-mail*, telepresencia y otros.
 - **Propósito:** Por qué se requiere comunicar a la persona interesada.
 - **Frecuencia:** Diaria, semanal, quincenal y otros.
 - **Emisor:** Miembro del equipo que se encarga de emitir la comunicación a su destinatario.
 - **Fecha de inicio:** Fecha de inicio en la que se empieza a comunicar.

Plan de comunicaciones					
Receptor	Medio de comunicación	Propósito	Frecuencia	Emisor	Fecha de inicio

Verificación de resultados:

- Si existe un número significativo de modificaciones a los requerimientos y el alcance puede significar que los interesados no están involucrados o alineados con los objetivos del proyecto.
- Los interesados principales brindan apoyo y se encuentran satisfechos.
- Los interesados que se oponen al proyecto o alguno de sus entregables no afectan negativamente la consecución del proyecto.

6.5.2 Equipo

El equipo de proyecto constituye una de las áreas más importantes en la gestión de proyectos debido a que todos los proyectos de software son desarrollados por seres humanos.

El administrador de los proyectos es el líder principal que debe comunicar efectivamente a todos los miembros del equipo la visión del proyecto e involucrarlos activamente en las reuniones de planificación con los interesados.

Procedimientos por llevar a cabo:

En el caso de las empresas mypes, el equipo generalmente es el mismo para todos los proyectos. Sin embargo, se pueden definir las habilidades, roles y responsabilidades de los miembros del equipo de trabajo para cada proyecto mediante los siguientes artefactos.

Artefactos/herramientas por utilizar:

1. **Acta de constitución de equipo:** este artefacto permite documentar a cada uno de los miembros del equipo, sus habilidades, alcance (lo que el equipo realizará) y los recursos con los que cuenta. Los datos que puede incluir el acta de constitución de equipo son los siguientes:
 - **Propósito del equipo:** Indica cuál es el propósito de la constitución del equipo, para qué están reunidos.
 - **Duración:** Corresponde a la duración en la que el equipo está reunido. Además, indica el tiempo que el equipo invertirá semanal o mensualmente en llevar a cabo sus tareas.
 - **Alcance:** El alcance del equipo. Cuál es el trabajo que deben realizar.
 - **Recursos de apoyo:** recursos de apoyo que se utilizan como software, salas de reunión, materiales y otros.
 - **Resultado final esperado:** Resultado final que se espera del equipo.
 - **Miembros:** Indique los miembros del equipo.

- **Principales habilidades:** Indique las principales habilidades del equipo de trabajo.

Utilice la siguiente plantilla como apoyo:

Acta de constitución de equipo				
Propósito del equipo	Duración	Alcance	Recursos de apoyo	Resultado final esperado
Miembros	Principales habilidades			

- 2. Matriz de roles y responsabilidades:** La matriz de roles y responsabilidades (en adelante RACI) es una herramienta que se utiliza para definir los roles y responsabilidades de cada miembro del equipo de trabajo. Para utilizar la matriz debe definir previamente las actividades del proyecto y quien las realizará. Esta matriz se explica a continuación:

Actividades	Administrador de los proyectos	Cliente	Desarrolladores
Fases iniciales del proyecto			
Caso de negocio o enunciado de alcance	C	R	
Fase de planificación	R /A		C
Requerimientos	R /A		C
Cronogramas	R /A		C
Presupuesto	R /A		C
Adquisiciones	R /A		C
Riesgos	R /A		C
Calidad	R /A		C
Fase de ejecución			
Desarrollo de los entregables	I	C	R
Creación de informes	R/A		C
Fase de control			
Control de cambios (presupuesto, tiempos, riesgos y otros)	R/I		C

Fase de cierre	
Lecciones aprendidas	R/I
Liberación de recursos	R/I
Finiquito	R/I

R= responsables, A = autoridad, C= consultores, I = informadores

La matriz anterior ejemplifica una serie de actividades que realiza cada miembro del equipo. Existen diferentes categorías de responsabilidad que cada miembro del equipo posee, las cuales son:

- **Responsables:** Estos son los responsables de realizar la tarea asignada.
- **Autoridad:** Corresponde a los miembros del equipo que tienen un nivel de autoridad para asignar tareas y tomar decisiones. Esta labor la realiza tanto el administrador de los proyectos como el cliente.
- **Consultores:** Tienen conocimiento amplio del contexto y pueden emitir opiniones respecto a un tema en particular.
- **Informadores:** Corresponden a los que se encargan de informar a los interesados sobre el trabajo que se realizó. Los responsables comunican a los informadores cuando una tarea se termina y estos informan a los interesados.

Verificación de resultados:

1. Los miembros del equipo conocen la visión y objetivos del proyecto.
2. Los miembros del equipo confían y colaboran entre sí.
3. El equipo se adapta a situaciones cambiantes.
4. El cliente conoce quién trabaja en el proyecto y sus habilidades.

6.5.3 Enfoque de desarrollo y ciclo de vida

“Este dominio de desempeño implica establecer el enfoque de desarrollo, la cadencia de entrega y el ciclo de vida del proyecto necesarios para optimizar los resultados del mismo” (PMI, 2021, p.32)

Para los proyectos de desarrollo existen diferentes enfoques de desarrollo y ciclo de vida. Cada enfoque de desarrollo posee una cadencia (frecuencia) para entregar los entregables del proyecto. Esta cadencia puede ser de varios tipos entre ellos:

- Entrega única: La entrega se realiza al final del proyecto y solo tiene un único entregable como ocurre en los proyectos de construcción, automovilística, aeronáutica y otros.
- Entrega múltiple: Las entregas pueden ser múltiples durante el proyecto.
- Entregas periódicas: Son similares al punto anterior, pero estas tienen un cronograma que indica la frecuencia con la que se transferirán los entregables. Este último se aplica en los proyectos de software con un flujo de proceso iterativo donde se realiza una entrega de software funcional en una determinada frecuencia de tiempo.

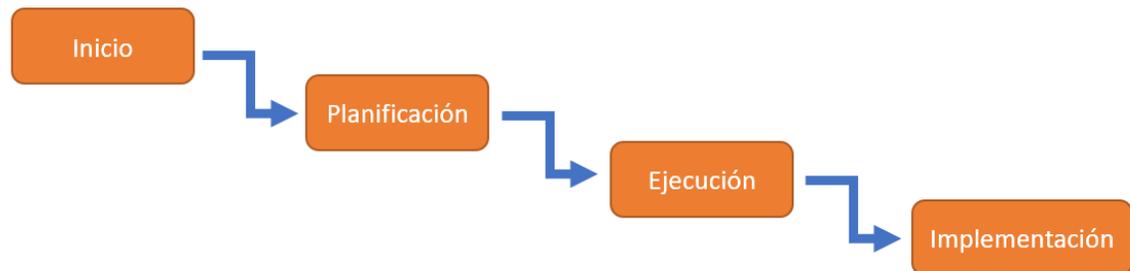
La guía PMBOK también menciona que existen dos enfoques principales de desarrollo, el enfoque **predictivo y adaptativo**. En el enfoque predictivo los requisitos se definen y analizan al comienzo del proyecto. Como los requisitos se conocen bien la planificación se sigue al pie de la letra durante el proyecto. Además, se le conoce como **enfoque en cascada**.

En los enfoques predictivos existe un documento de especificación de requerimientos donde se detallan **todos los requerimientos** del sistema antes de comenzar a desarrollarlo. El equipo de desarrollo sigue este documento al pie

de la letra en el desarrollo del sistema y, generalmente, se realiza una única entrega al final.

Figura 25

Ciclo de vida en enfoques predictivos

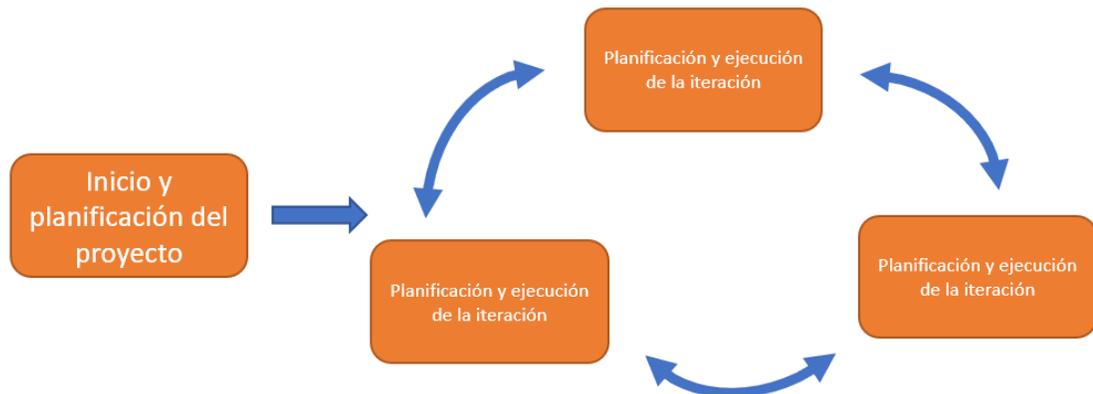


Por otra parte, el enfoque adaptativo se utiliza cuando los requerimientos tienen un alto nivel de incertidumbre y es probable que sufran cambios a lo largo del proyecto. Al inicio del proyecto solo se establecen a **alto nivel** y se refinan conforme el cliente proporcione retroalimentación sobre el trabajo realizado en cada iteración o etapa del proyecto.

En cada iteración el equipo únicamente desarrolla una serie de requerimientos priorizados por el cliente, una vez desarrollados se muestran al cliente y con base en su retroalimentación se refina el requerimiento hasta que este quede completado. En la siguiente iteración se selecciona otro conjunto de requerimientos priorizados y se repite el proceso una y otra vez hasta que el sistema quede terminado completamente y el cliente quede satisfecho.

Figura 26

Ciclo de vida en enfoques adaptativos ágiles



Procedimientos por realizar:

Las empresas generalmente poseen una metodología y enfoque de desarrollo definidos que se replica en todos los proyectos que gestionan. No existe un artefacto específico para determinar cuál enfoque de desarrollo utilizar. Sin embargo, a través de la definición de alcance y los requerimientos del cliente la empresa puede identificar cuál es el enfoque de desarrollo más adecuado para el proyecto, es decir, juicio de experto.

En el caso de las empresas analizadas, la mayoría utiliza enfoques adaptativos. Algunos de los artefactos que pueden determinar el enfoque de desarrollo por utilizar se explican más adelante en la presente guía. Por otra parte, más adelante en el dominio de desempeño de la adaptación se menciona cómo adaptar los enfoques y metodologías de la empresa en función de cada proyecto.

Artefactos/herramientas por utilizar: Los siguientes artefactos pueden ser de utilidad para determinar el enfoque de desarrollo del proyecto.

- Declaración de alcance.
- Documento de especificación de requerimientos.
- Pila de producto.

- Otras conversaciones realizadas con el cliente que ayuden a identificar el enfoque de desarrollo más adecuado para el proyecto.
- Enfoque de desarrollo que se utiliza.

Verificación de resultados:

1. Cada fase del enfoque de desarrollo escogido presenta una serie de entregables que posee criterios de salida apropiados.
2. Al final de cada fase el cliente proporciona su retroalimentación sobre el trabajo realizado en los entregables que se presentaron.

6.5.4 Planificación

“La planificación organiza, elabora y coordina el trabajo del proyecto a lo largo de la totalidad del mismo” (PMI, 2021, p.51).

La planificación de un proyecto se realiza según el enfoque de desarrollo que se utiliza. En los enfoques predictivos existe una etapa dedicada únicamente a la planificación inicial. Conforme avance el proyecto, el plan se ejecuta y pocas veces existen variaciones al plan debido a que el alcance y los requerimientos están definidos y detallados. En general, se elabora un plan para la dirección del proyecto que sirve de guía en el momento de su ejecución.

Usualmente, el momento de planificar un proyecto se toma en cuenta lo siguiente:

- Alcance.
- Tiempo.
- Costo.
- Calidad.
- Riesgos.
- Comunicaciones.

- Adquisiciones.

Por otra parte, las empresas que utilizan enfoques adaptativos realizan una planificación inicial, pero no tan detallada como en los enfoques predictivos debido a que se requiere que el cliente vea el producto funcionando sobre planificación extensiva. Esto no quiere decir que no se planifique. Como en ocasiones en este tipo de enfoque no se conocen de antemano todos los requerimientos se crea una pila de producto con todos los requerimientos a alto nivel. En cada iteración se conversan con el cliente los requerimientos de la pila de producto que desarrollan y las dudas que existan se aclaran para refinar el requerimiento. Los requerimientos en este tipo de enfoque se denominan **historias de usuario**.

6.5.4.1 Alcance

Procedimientos por llevar a cabo:

1. **Identificación de los objetivos:** Identifique los objetivos del proyecto que deben cumplirse con el desarrollo del sistema. Estos objetivos deben proporcionarse por cliente el debido a que él es el que conoce el negocio en profundidad.
2. **Definición de los entregables:** Identifique los entregables por desarrollar y la frecuencia con la que se suministran estos entregables.
3. **Identificación del costo estimado del proyecto:** Indique el costo estimado del proyecto con base en sus actividades principales. Las técnicas para el cálculo de los costos se mencionan más adelante.
4. **Elaboración del cronograma:** Elabore un cronograma que indique las posibles fechas de entrega para cada entregable. Las técnicas para la definición de las actividades del proyecto y su estimación en tiempo se mencionan más adelante.
5. **Identificación de las exclusiones del proyecto:** Identifique todos aquellos aspectos que no se incluirán dentro del proyecto y que la empresa no desarrollará ni implementará.

6. **Definición de los requerimientos:** Recopile los requerimientos del sistema por desarrollar. Si utiliza Scrum (enfoque predictivo) trabaje con el cliente en la definición de las historias de usuario y colóquelas en la pila del producto. Además, si usa un enfoque más predictivo donde se conocen los requerimientos de antemano elabore un documento de especificación de requerimientos.

Artefactos/herramientas por utilizar:

1. **Declaración de alcance:** Documento que especifica cuál es el alcance del proyecto. Incluya la información que se sugiere seguidamente:
- **Objetivos:** Se describen los objetivos del proyecto que deben cumplirse con el desarrollo del sistema. Estos objetivos deben proporcionarse por el cliente debido a que él conoce el negocio en profundidad.
 - **Entregables principales:** Indicar qué es lo que se entrega al cliente y la frecuencia con la que se suministran estos entregables.
 - **Cronograma:** Adjuntar un cronograma que indique las posibles fechas de entrega para cada entregable.
 - **Costo estimado:** Indicar el costo que tiene el proyecto.
 - **Exclusiones:** Indica todos aquellos aspectos que no se incluirán dentro del proyecto y que la empresa no desarrollará ni implementará.
 - **Restricciones:** Se indican las restricciones que pueden afectar el desarrollo del proyecto.

Utilice la siguiente plantilla como referencia:

Definición de alcance					
Objetivos	Entregables principales	Cronograma	Costo estimado	Exclusiones	Restricciones

2. Documento de especificación de requerimientos: Los requerimientos funcionales y no funcionales del software también se pueden recopilar y analizar como parte de la definición de alcance. Puede elaborar un documento de especificación de requerimientos donde se documenten los mismos. Incluya la información que se sugiere seguidamente:

- **Resumen del alcance:** Se detalla de manera resumida el alcance del sistema.
- **Objetivos:** Se detallan los objetivos que se lograrán con el desarrollo del sistema.
- **Usuarios:** Usuarios que usarán el sistema y sus interacciones.
- **Restricciones:** Se indican las restricciones que pueden afectar el desarrollo del sistema.
- **Requerimientos funcionales:** Se enumeran los requerimientos funcionales que tiene el sistema de manera detallada y sin ambigüedades. Si es posible colocar los criterios de aceptación.
- **Requerimientos no funcionales:** Se enumeran los requerimientos no funcionales que tiene el sistema de manera detallada y sin ambigüedades.
- **Arquitectura del sistema:** Se debe indicar el patrón de arquitectura a partir del cual se construye el sistema. Adjuntar diagramas que muestren los componentes del sistema, relaciones y dependencias, además del flujo de los diferentes procesos.

Utilice la siguiente plantilla como referencia:

Especificación de requerimientos	
Resumen del alcance:	
Objetivos:	
Usuarios:	
Restricciones:	

Requerimientos funcionales:	
Requerimientos no funcionales:	
Arquitectura del sistema:	

3. **Pila de producto:** Si utiliza metodología Scrum, elabore la pila del producto con sus historias de usuario. La pila de producto, en general, tiene los siguientes apartados:

- **Código de la historia de usuario:** Código único que sirve para identificar a la historia de usuario.
- **Criterios de aceptación:** Lo que tiene que cumplir la historia de usuario para darse como completada.
- **Prioridad:** Indica la prioridad: baja, media y alta.
- **Dependencias:** Se indica si la historia tiene una dependencia por lo que está bloqueada.
- **Iteración:** Se indica el sprint en el que se realiza la iteración.
- **Estado:** Estado de la historia: Por hacer, en progreso y terminada.

Utilice la siguiente plantilla como referencia:

Pila del producto	
Código de la historia de usuario:	
Criterios de aceptación	
Prioridad:	
Restricciones:	
Dependencias:	
Iteración:	
Estado:	

6.5.4.2 Tiempo

Para determinar el tiempo de duración del proyecto y crear un cronograma se debe llevar a cabo una lista de tareas o actividades para el proyecto con base en

los requerimientos recolectados. Una vez determinadas estas actividades se realiza la estimación de cada una. La guía PMBOK menciona que existen dos tipos de estimaciones, la **absoluta y relativa**. La absoluta corresponde a una unidad de medida exacta como dada en días, horas, meses y otros. La relativa corresponde a una unidad de medida dada en términos relativos. El esfuerzo relativo se traduce en la primera estimación del esfuerzo total requerido usando la noción de velocidad (Sommerville, 2011). Un ejemplo de unidades de medida relativa son los **puntos de historia** que se utilizan en varias metodologías ágiles.

En general, en el momento de realizar un cronograma de trabajo se colocan medidas absolutas en lugar de medidas relativas. Es decir, para cada actividad se colocan los días, meses u horas que durará cada actividad del proyecto.

Procedimientos por realizar:

- 1. Determine las actividades del proyecto:** Las actividades del proyecto se determinan a través de los requerimientos. Existe una herramienta que permite descomponer las actividades en partes más pequeñas denominada estructura de desglose del trabajo (EDT):
 - Si los requerimientos se determinaron a través de historias de usuario (épicas), generalmente estas deben descomponerse en tareas o historias de usuario más pequeñas. **Las historias de usuario también son actividades del proyecto y pueden colocarse en la EDT.**
 - Indique si las actividades tienen dependencias unas con otras, es decir, si una depende de otra antes de poder iniciarse.
- 2. Determine la duración de cada actividad:** Para determinar la duración de las actividades existen varias técnicas de estimación. Para efectos de la presente guía se explica la técnica PERT. La técnica PERT consiste en una fórmula que proporciona la duración aproximada de la actividad tomando en cuenta variables como el tiempo probable, optimista y pesimista. Este tiempo puede darse en días o semanas.

$$\text{Duración} = \frac{o + 4m + p}{6}$$

M = escenario más probable, es decir, el tiempo probable que se durará haciendo la tarea con base en el juicio de experto.

O = tiempo optimista tomando en cuenta que no se presentan bloqueos, dependencias y atrasos para realizar la tarea.

P = tiempo pesimista tomando en cuenta que en la actividad presenta bloqueos, dependencias y atrasos para realizar las tareas.

Para cada actividad del proyecto determine el tiempo de duración más probable, el tiempo optimista y el tiempo pesimista. Reemplace los valores en la fórmula y obtenga su duración. La duración obtenida en la fórmula representa el tiempo de la actividad.

- 3. Elabore un diagrama de Gantt:** Utilice una herramienta informática que le permita elaborar un diagrama de Gantt. Este diagrama permite observar el tiempo de duración de las actividades a través del tiempo y muestran la fecha de finalización del proyecto. Algunas de estas herramientas pueden mostrar la **ruta crítica y los tiempos de holgura** de las actividades. La ruta crítica corresponde a la secuencia de actividades que deben hacerse en el tiempo establecido para no sufrir algún retraso que modifique la fecha de entrega propuesta. Por otro lado, los tiempos de holgura son actividades a las que se les puede retrasar su inicio o alargar su duración debido a que, al no depender de otras actividades, el tiempo del proyecto no se vea afectado. En general, la ruta crítica y los tiempos de holgura se determinan mediante un diagrama de red con la técnica CPM, sin embargo, como se mencionó muchos sistemas para gestionar proyectos ya calculan ambos elementos de manera automática.

Artefactos/herramientas por utilizar:

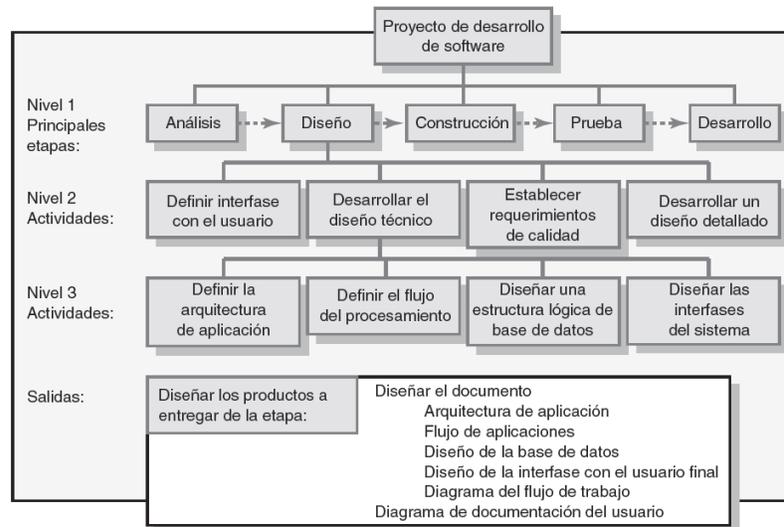
Los artefactos mencionados a continuación pueden elaborarse con herramientas informáticas para mayor facilidad. Muchas de estas herramientas

son de uso gratuito. Algunas herramientas que se pueden utilizar se explican al final de la presente guía.

1. Diagrama de EDT:

Figura 27

EDT para proyectos de desarrollo de software

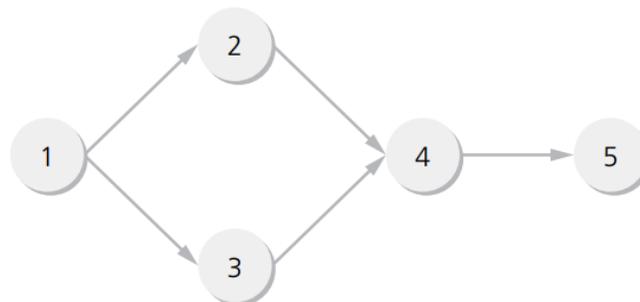


Nota. Tomado de *Administración de proyectos*, por Gray y Larson, 2009.

2. Diagrama de red:

Figura 28

Diagrama de red del proyecto



Nota. Tomado de *Administración de proyectos*, por López y Lanckenau, 2017.

3. Diagrama de Gantt:

Figura 29

Diagrama de Gantt

Actividad	Pred	Dur	1	2	3	4	5	6	7	8	9	10
1 Revisar las instrucciones	–	1	■									
2 Investigar en internet	1	2		■	■							
3 Estudiar con ayuda del libro	1	3		■	■	■						
4 Hacer la tarea	2,3	2					■	■				
5 Entregar la tarea	4	1							■			

Nota. Tomado de Administración de proyectos, por López y Lanckenau, 2017.

6.5.4.3 Presupuesto

La elaboración del presupuesto en los proyectos de software tiene un componente principal: el costo de la mano de obra y los costos indirectos.

Procedimientos por realizar:

- 1. Determine el costo por hora por cada recurso:** Las horas recurso se calculan multiplicando el número de recursos (total de personas) por las horas de la jornada laboral (generalmente 8 horas) por el número de jornadas (generalmente cinco días a la semana).

$$\text{Horas recurso} = \text{Número de recursos} * \text{Horas de la jornada laboral} * \text{Número de jornadas}$$

- 2. Determine el costo total de la mano de obra:** Para calcular el costo de la mano de obra tome el tiempo total que invertirá el recurso realizando cada tarea de la EDT y multiplíquelo por el costo por hora (generalmente se toman en cuenta las horas debido a que es más sencillo realizar los cálculos).

$$\text{Costo} = \text{Horas recurso} * \text{Costo del recurso}$$

- 3. Determine los costos indirectos:** Los costos indirectos son aquellos en que la empresa debe incurrir para ejecutar el proyecto más allá de la mano

de obra, por ejemplo, el alquiler de la oficina, la secretaria, servicios públicos como la luz, agua, Internet y otros. Utilice la siguiente fórmula para calcular los costos indirectos:

$$\text{Costos indirectos} = [1 - (\text{costos directos del año anterior} / \text{ingresos totales})] * 100$$

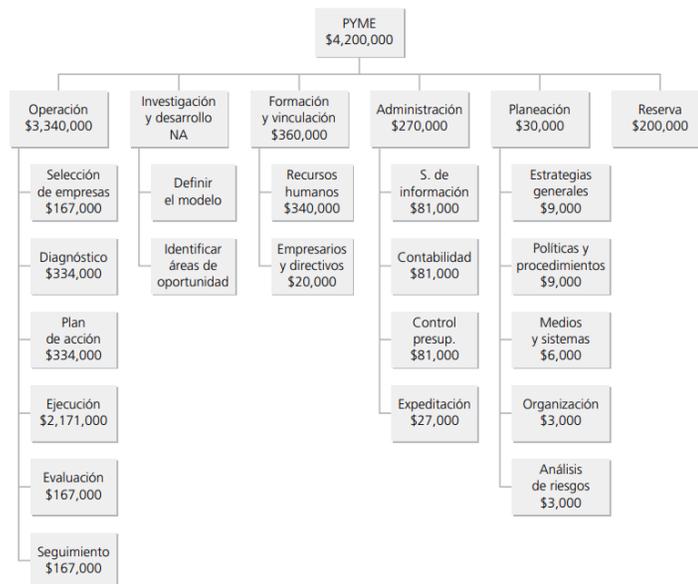
La fórmula anterior toma en cuenta los costos directos del año anterior dividido entre los ingresos totales que se generan por la empresa para producir el porcentaje de costos indirectos que se debe incluir en cada actividad. Por otra parte, debido a que las empresas obtienen ganancia para su capital, se debe indicar un porcentaje de ganancia que la empresa obtendrá por sus servicios.

Artefactos/herramientas por utilizar:

1. **Estructura de desglose de recursos:** Este artefacto es igual a la EDT, con la diferencia de que se coloca el costo de cada actividad.

Figura 30

EDT con el costo de cada una de las actividades



Nota. Tomado de *Administración de proyectos*, por López y Lanckenau, 2017.

6.5.4.4 Recursos físicos y Adquisiciones

Los recursos físicos corresponden a todo aquello que es tangible y que la empresa deba utilizar o adquirir para la ejecución del proyecto. En los proyectos de software las empresas generalmente cuentan con instalaciones propias que generan costos indirectos, mas no son adquisiciones propias del proyecto. En ciertos casos, se requiere que se realicen adquisiciones de productos o servicios para un proyecto en específico, por ejemplo, adquisición de licencias y librerías de software, servidores físicos o en la nube para poner en producción el aplicativo.

Procedimientos por realizar:

1. Determine las adquisiciones que se deben realizar para el proyecto:

Identifique los siguientes elementos para cada producto o servicio:

- **Producto o servicio:** Nombre del producto o servicio por contratar.
- **Cantidad:** Cantidad a adquirir.
- **Costo:** Costo del producto o servicio por adquirir.
- **Duración del contrato:** Duración del contrato en semanas, meses o años.
- **Inicio de contrato:** Fecha de inicio del contrato.
- **Fin del contrato:** Fin del contrato.

Artefactos/herramientas por utilizar:

1. **Plan de adquisiciones:** De acuerdo con la información anterior elabore un plan de adquisiciones para el proyecto, utilizando como referencia el siguiente formato:

Plan de adquisiciones					
Producto/servicio	Cantidad	Costo	Duración del contrato	Inicio del contrato	Fin del contrato

--	--	--	--	--	--

Verificación de resultados:

1. El alcance, costo, tiempo, adquisiciones y otros se documenta apropiadamente con los artefactos necesarios.
2. Cada uno de los procesos mencionados los discute y aprueba el cliente y si existen cambios se comunica adecuadamente a través de los canales definidos en el plan de comunicaciones.
3. La planificación realizada es suficiente para cumplir con las expectativas del cliente.

6.5.5 Trabajo del proyecto

“El trabajo de proyecto está asociado con el establecimiento de los procesos y la realización del trabajo para permitir que el equipo de proyecto cumpla con los entregables y resultados esperados” (PMI, 2021, p. 69).

Procedimientos por realizar:

Una manera efectiva de medir el trabajo realizado por el equipo son las reuniones. Estas permiten que el equipo indique el estado del trabajo que está realizando y si existen bloqueos.

1. **Realice reuniones diarias con el equipo de trabajo:** En la metodología de trabajo Scrum se realiza una reunión diaria donde cada miembro del equipo responde las siguientes preguntas: ¿Qué se hizo ayer? ¿Qué se hará hoy? ¿Existe algún impedimento?
La respuesta a estas preguntas permite determinar qué es lo que ha realizado el equipo de trabajo, cómo se ha avanzado conforme a lo planeado y si hay algún miembro que requiera de asistencia.
2. **Realice reuniones retrospectivas:** Cada vez que se haga la entrega de una funcionalidad del sistema al cliente, haga las siguientes

preguntas a los miembros del equipo: ¿Qué funcionó o salió bien durante la iteración o periodo de trabajo? ¿Qué causó problemas o no salió bien? ¿Qué se puede mejorar? La información suministrada permite optimizar el proceso para la siguiente iteración o periodo de trabajo.

Artefactos/herramientas por utilizar:

- La pila del producto o el documento de requerimientos puede ser de utilidad para determinar cuánto se ha avanzado respecto al trabajo planeado.
- El cronograma también es útil para conocer cómo avanzan las actividades del proyecto respecto a lo establecido en el cronograma.

Verificación de resultados:

1. El equipo de trabajo produce los entregables acordados.
2. El equipo de trabajo informa constantemente si existen bloqueos que impidan realizar su trabajo.
3. Se tiene un claro conocimiento sobre el avance del proyecto y las tareas pendientes de realizar y se comparan contra lo planeado.

6.5.6 Entrega

“La entrega del proyecto se enfoca en cumplir con los requisitos, el alcance y las expectativas de calidad para producir los entregables esperados que impulsarán los resultados previstos” (PMI, 2021, p. 69).

Los entregables del proyecto están determinados por el alcance y los requisitos. En enfoques predictivos generalmente el producto funcionando se entrega hasta las últimas etapas del proyecto, mientras que en los enfoques adaptativos ágiles se realizan entregas de software funcional de manera

periódica. Para gestionar cada entregable del proyecto haga los siguientes procedimientos:

Procedimientos por realizar:

1. **Identifique las tareas que se deben realizar para poner en producción el entregable:** Por ejemplo, configurar los servidores de producción, migrar las bases de datos, establecer la configuración de las tuberías de integración y despliegue continuo (CI/CD).
2. **Indique a la persona que es responsable de realizar la tarea:** Identifique cuál miembro del equipo es el encargado de llevar a cabo esta tarea.
3. **Indique la fecha de inicio y de finalización de cada tarea.**
4. **Indique el estado de la tarea:** Indique el estado de cada tarea: Activo, en progreso y completado.

Herramientas/artefactos por utilizar:

1. **Plan de despliegue:** Utilice esta herramienta para listar todas las tareas necesarias para desplegar la aplicación para entregarse al cliente.

Plan de despliegue				
Tarea	Responsable	Fecha de inicio	Fecha de finalización	Estado

6.5.6.2 Calidad

La calidad es un proceso que debe tomarse en cuenta en todas las fases del proyecto. Sin embargo, en la fase de ejecución donde se desarrollan las funcionalidades del software, es recomendable que la empresa defina una estrategia para asegurar que tales requerimientos y funcionalidades posean la menor cantidad de errores posibles que puedan conducir a defectos que provoquen fallos.

Uno de los artefactos más importantes por producir es el **plan de pruebas** que se realiza al software. Este plan contiene una serie de casos de prueba para probar los requerimientos funcionales de este, además, contiene el tipo de pruebas que se realiza al sistema. Más adelante se explican ambos artefactos.

Procedimientos por realizar:

1. **Elabore el plan de pruebas que se realiza al sistema:** En el plan de pruebas coloque la siguiente información básica:

- **Recursos necesarios para ejecutar las pruebas:** Indique las herramientas de software que se requieren para ejecutar las pruebas.
- **Alcance:** Determine las funcionalidades que se probarán y las que no.
- **Tipos de pruebas:** Detalle los tipos de pruebas que se llevan a cabo (unitarias, integración, regresión, aceptación de usuario y otras).
- **Objetivos:** Para cada tipo de prueba se menciona cuál es el objetivo de realizarla.
- **Criterios de suspensión y reanudación:** Mencione los criterios de suspensión que indican en cuáles criterios el proceso de pruebas debe suspenderse. Por ejemplo, en caso de que el software posea una gran cantidad de defectos, el proceso de pruebas puede suspenderse hasta que se corrijan tales defectos. Por otra parte, los criterios de reanudación indican en cuáles criterios el proceso de pruebas debe reanudarse. Por ejemplo, si las pruebas permanecieron suspendidas debido a que el software tenía una gran cantidad de errores.

2. **Elabore los casos de prueba para el sistema:** Para cada caso de prueba identifique lo siguiente:

- **Pasos:** Se detallan los casos que la persona probadora debe seguir para realizar la prueba.

- **Resultado esperado:** El resultado que se espera obtener con la prueba. Generalmente, se relaciona con los criterios de aceptación de los requerimientos/historias de usuario.
- **Resultado obtenido:** El resultado que se obtuvo al ejecutar la prueba.
- **Estado:** Se indica si la prueba falló o no.

Artefactos/herramientas por utilizar:

1. **Plan de pruebas:** Utilice la siguiente plantilla como referencia.

Plan de pruebas				
Resumen: Resumen del contenido del plan de pruebas				
Código	Pasos	Resultado esperado	Resultado obtenido	Estado

2. **Plantilla de casos de prueba:** Los casos de prueba pueden ejecutarse de manera automática al utilizar una herramienta de automatización de pruebas. Sin embargo, se proporciona una plantilla que se puede utilizar en caso de que estos no estén automatizados.

Caso de prueba				
Recursos	Alcance	Tipos de prueba	Objetivos	Criterios de suspensión y reanudación

Verificación de resultados:

1. Antes de finalizar cada requerimiento o entregable se realiza una serie de pruebas que garanticen que lo que se entrega pasó por un proceso de calidad. Las pruebas se definieron en el plan de pruebas.
2. En el momento de otorgar el entregable se sigue el proceso de despliegue establecido.

6.5.7 Medición

“La medición involucra evaluar el desempeño del proyecto e implementar respuestas apropiadas para mantener un desempeño óptimo” (PMI, 2021, p. 93).

La medición del desempeño del proyecto es útil para determinar si las actividades realizadas cumplen el alcance, costo y tiempo establecidos. A estos últimos elementos se les llama líneas base del proyecto. La guía PMBOK menciona que la medición se realiza sobre:

- Las líneas base.
- Recursos.
- Valor de negocio.

Además, menciona que la información de las mediciones debe mostrarse de manera representativa para su comprensión por parte de los interesados, es decir, mediante tableros de control.

Procedimientos por realizar:

1. Monitoree el desempeño en cuanto a las líneas base: El desempeño en cuanto a las líneas base (costo, tiempo y alcance) puede monitorearse de la siguiente manera:

- **Fecha de inicio y finalización:** Visualice cuán alejada está la fecha de inicio y finalización del proyecto conforme a lo planeado con base en el trabajo que se ha realizado hasta el momento. Si el tiempo es un factor determinante puede realizar ajustes para lograr que el proyecto se entregue en la fecha propuesta. Una herramienta útil es observar las actividades de la ruta crítica en el diagrama de Gantt y su fecha de terminación esperada contra la que se presenta en el cronograma.
- **Tasa de finalización de características:** Para conocer si el trabajo avanza conforme a lo previsto tome el total de requerimientos del software por desarrollar. Si hay un número muy bajo de características completadas y el proyecto lleva tiempo en curso, probablemente exista un problema de desempeño en el equipo de trabajo. En este caso el equipo debe reunirse para identificar las causas del bajo rendimiento. Es importante mencionar que el trabajo realizado y pendiente de realizar puede observarse en la herramienta informática que se utiliza para gestionar el proyecto.
- **Costo real en comparación con el costo planificado para el proyecto:** Compare el costo gastado con el costo planificado. Si lo gastado hasta el momento sobrepasa o está cerca de llegar a lo planificado y el proyecto todavía está en una etapa temprana deben tomarse medidas.
- **Variación del costo de los entregables:** Compare cuánto costó desarrollar un entregable con el costo estimado para ese mismo entregable. Este punto es similar al anterior, solo que aquí se compara en el ámbito de entregable.

2. **Monitoree el desempeño en cuanto a los recursos:** El dominio de desempeño del trabajo del proyecto presentado anteriormente mencionó cómo monitorear al equipo de trabajo.
3. **Determine el valor de negocio:** Según PMI (2021): “Las mediciones del valor de negocio se utilizan para garantizar que el entregable del proyecto se mantenga alineado con el caso de negocio y los planes de realización de beneficios” (p. 102). Para determinar el valor de negocio existen las siguientes métricas:
 - **Relación costo-beneficio:** Esta métrica aplica tanto para la empresa contratante como para la contratista. Es importante que antes de iniciar un proyecto responda las siguientes preguntas:
¿Tenemos los suficientes recursos para ejecutar el proyecto?
¿Tenemos experiencia en las tecnologías que requiere el software que se desarrolla? ¿Los ingresos son los suficientes para invertir en gastos de capital futuros?
 - **Entrega de beneficios planificada en comparación con entrega real de beneficios:** Cada vez que haga presente un entregable debe considerar la retroalimentación proporcionada por el cliente. Si el cliente considera que el entregable no aporta el valor esperado debe tomar en cuenta esta retroalimentación y volver a presentarlo con las sugerencias que se implementan en el software.
 - **Retorno de la inversión (ROI):** Esta métrica la maneja la empresa contratante. La compañía determina si es viable continuar invirtiendo recursos en el proyecto desde un punto de vista financiero.
4. **Monitoree la satisfacción de los interesados:** En los proyectos de software la satisfacción de las necesidades de los interesados es primordial. Para realizar lo anterior utilice alguno de los siguientes artefactos sugeridos en el PMBOK: puntuación neta del promotor,

diagrama de estados de ánimo, o bien rellenen un formulario con escalamiento de Likert para determinar su nivel de satisfacción.

Artefactos/herramientas por utilizar

Las herramientas para medir el trabajo del proyecto son varias. Algunas no deben hacerse desde cero, sino que deben utilizarse para hacer las mediciones. Los artefactos/herramientas por utilizar para este dominio de desempeño son los siguientes:

- 1. Diagrama de Gantt.**
- 2. Cronograma de trabajo.**
- 3. Presupuesto del proyecto.**
- 4. Plan de adquisiciones.**
- 5. Puntuación neta del promotor:** Puede utilizar esta herramienta para medir el nivel de satisfacción del cliente en una escala del 1 al 10. De acuerdo con la puntuación que se otorga el cliente se categoriza en promotor, pasivo y detractor. El promotor asigna un puntaje de entre 9 y 10, el pasivo del 7 a 8 y el detractor de 0 a 6.
 - Los promotores son aquellos clientes que están satisfechos con el trabajo realizado y lo recomiendan a alguien más.
 - Los pasivos están moderadamente satisfechos y es posible que no recomienden la empresa a nadie más.
 - Los detractores son clientes insatisfechos que pueden dañar la reputación de la empresa.
- 6. Diagrama de estados de ánimo:** Los miembros del equipo también son interesados importantes del proyecto. Este diagrama se utiliza para conocer el estado de ánimo de los miembros del equipo en un periodo, de manera que se les brinde seguimiento para identificar áreas de mejora y posibles problemas. Para cada día de la semana los miembros del equipo indican su estado de ánimo con un emoticón, como se muestra en la siguiente imagen:

Figura 31

Diagrama de estados de ánimo

	Domingo	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado
Tom							
Lucy							

Nota. Tomado de *Guía de los Fundamentos para la Dirección de Proyectos y el Estándar para la Dirección de proyectos*, por Project Management Institute, 2021.

7. **Evalúe la moral del equipo de trabajo:** Otra alternativa al diagrama de estados de ánimo es solicitar a los miembros del equipo que evalúen en una escala del 1 al 5 (escalamiento de Likert) las siguientes afirmaciones establecidas por la guía PMBOK (2021):

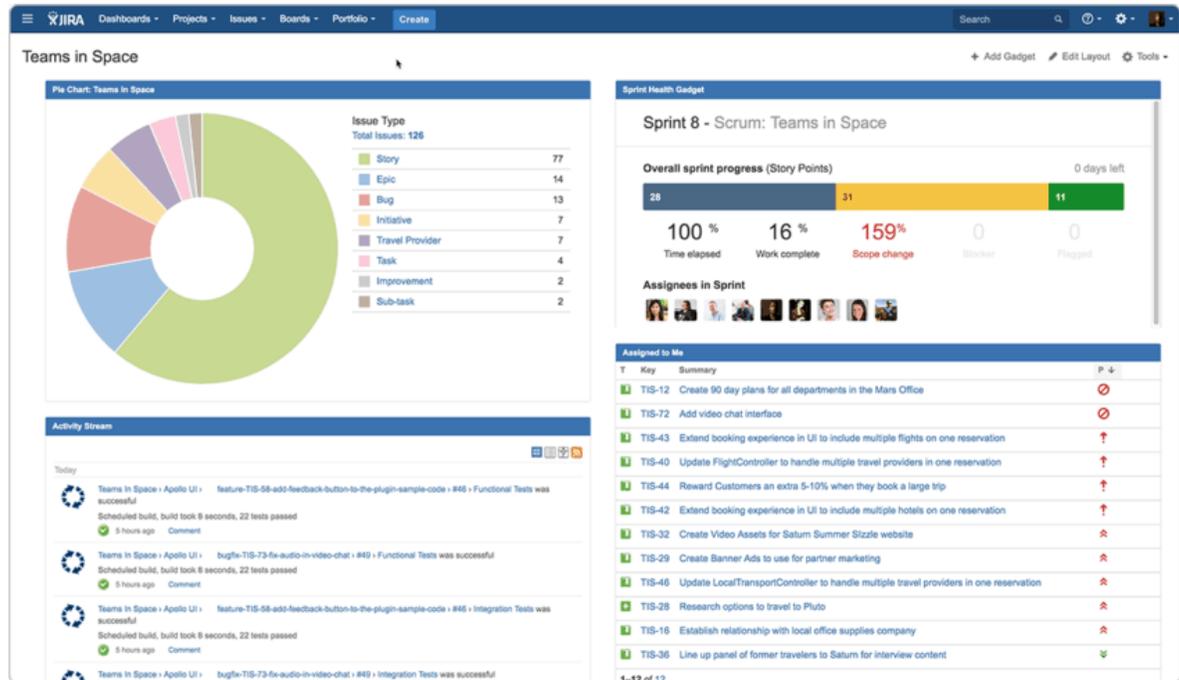
- Siento que mi trabajo contribuye con los resultados generales.
- Me siento apreciado.
- Estoy satisfecho con la forma en la que mi equipo de proyecto trabaja en conjunto.

6.5.7.1 Tableros de control

Para exponer las métricas del proyecto puede usar un software que muestre los datos en tiempo real y así evitar manejar la información de forma manual. Existe una gran cantidad de aplicativos de gestión de proyectos, sin embargo, el que más utilizan las empresas es Jira. Este sistema puede mostrar diferentes gráficos que muestren el avance del proyecto.

Figura 32

Página principal del sistema Jira



Nota. Tomado de Jira Dashboards Made Simple: The Why, How & Best Practices, por Idalko, 2021. <https://www.idalko.com/jira-dashboards/>

Procedimientos por realizar:

El único proceso por realizar es utilizar una herramienta de software que permita gestionar todo el proyecto de manera centralizada. Los diferentes artefactos como los cronogramas, presupuestos, requerimientos/historias de usuario y otros pueden gestionarse mediante estas herramientas. Algunas de ellas se mencionan a continuación:

Artefactos/herramientas por utilizar:

1. Software de gestión de proyectos como los siguientes:
 - Microsoft Project.
 - Jira.
 - Trello.
 - Asana.
 - Oddo.

Verificación de resultados:

- Las mediciones indican si el desempeño del proyecto en cuanto a las líneas base y al trabajo realizado es el que se espera o si existen variaciones.
- Las mediciones realizadas conducen a decisiones oportunas.
- Las mediciones realizadas muestran cuán alejadas estuvieron las estimaciones para futuros proyectos.

6.5.8 Incertidumbre

“Los proyectos existen en entornos con diferentes grados de incertidumbre. La incertidumbre presenta amenazas y oportunidades que los equipos de proyecto exploran, evalúan y deciden cómo manejar” (PMI, 2021, p.116).

Un proyecto es incierto porque no se puede predecir de antemano si es exitoso, por lo que tiene un alto nivel de incertidumbre. Durante el proyecto pueden surgir riesgos tanto positivos como negativos (oportunidades y amenazas) que lo afecten. Generalmente, los riesgos se identifican al inicio y se monitorean a lo largo de este. A continuación, se presenta una serie de procedimientos por realizar para gestionar los riesgos del proyecto según lo sugerido en el PMBOK.

Procedimientos por realizar:

1. **Identifique los posibles riesgos del proyecto:** Esta identificación puede hacerse una vez definido su alcance. Los riesgos pueden ser: operacional, técnico, financiero, estratégicos y otros.
2. **Identifique para cada riesgo el área que potencialmente se impactará:** Para cada riesgo identificado determine el área que se afectará en caso de que se materialice el riesgo. Es decir, se debe determinar si afectará el alcance, tiempo, costo.

3. **Identifique la probabilidad de ocurrencia:** Para determinar la probabilidad de que el riesgo se materialice utilice una matriz de riesgos. Esta matriz se proporciona más adelante.
4. **Identifique el costo y tiempo asociado con cada riesgo:** Para cada riesgo determine el costo y el tiempo asociado. Es decir, si existe un fallo en las computadoras de los miembros del equipo cuánto tiempo tardará en arreglarse y cuánto le costará al cliente el tiempo extra que no se trabajó.
5. **Determine la estrategia de mitigación para cada riesgo:** Para cada riesgo que pueda tener un impacto negativo o positivo indique la estrategia de mitigación. Es decir, la serie de pasos en caso de que se materialice.
6. **Al final del proyecto documente las lecciones aprendidas:** La documentación de las lecciones aprendidas puede ser útil para futuros proyectos. Puede seguir la siguiente estructura:
 - **Proyecto:** Proyecto donde se registra la lección aprendida.
 - **Descripción:** Descripción de la lección aprendida
 - **Impacto:** Se indica si tuvo un impacto positivo o negativo.
 - **Soluciones:** Solución brindada a la lección aprendida.

Artefactos/herramientas por utilizar:

1. **Matriz de riesgos:** Esta matriz es útil para colocar el porcentaje de probabilidad a un riesgo determinado. En el eje vertical se indica la frecuencia de que ocurra y en el eje horizontal el impacto que tiene el riesgo si se materializa.

Figura 33

Matriz de riesgos

Frecuente	100%	1	2	3	4	5
	90%	0,9	1,8	2,7	3,6	4,5
Probable	80%	0,8	1,6	2,4	3,2	4
	70%	0,7	1,4	2,1	2,8	3,5
Ocasional	60%	0,6	1,2	1,8	2,4	3
	50%	0,5	1	1,5	2	2,5
Posible	40%	0,4	0,8	1,2	1,6	2
	30%	0,3	0,6	0,9	1,2	1,5
Improbable	20%	0,2	0,4	0,6	0,8	1
	10%	0,1	0,2	0,3	0,4	0,5
		1	2	3	4	5
		Insignificante	Menor	Moderado	Mayor	Catastrófico

Nota. Tomado de Frecuencia e impacto en la matriz de riesgos, por Jiménez, 2021. <https://www.piranirisk.com/es/blog/matriz-de-riesgos-frecuencia-impacto/>

2. Plan para el manejo de riesgos: El plan para el manejo de riesgos puede crearse utilizando la siguiente plantilla:

RIESGOS IDENTIFICADOS							
CÓDIGO	DESCRIPCIÓN RIESGO	TIPO RIESGO	CATEGORÍA	OBJETIVO AFECTADO			
				ALCANCE	COSTO	TIEMPO	PROBABILIDAD
1	Fallo en los servidores que alojan el sistema, tanto en ambientes de desarrollo como de producción.	Amenaza	Operacional			X	Media
2	Fallo en las computadoras de los miembros del equipo, lo que causa retrasos en el cronograma y sobrecostos	Amenaza	Operacional		X	X	Baja
3	Retraso en la entrega de las	Amenaza	Financiero			X	Baja

	adquisiciones por parte del proveedor						
4	Cambios de requerimientos del cliente que modifiquen el alcance del proyecto	Amenaza	Operacional	X			Alta

RIESGOS IDENTIFICADOS					
CÓDIGO	DESCRIPCIÓN RIESGO	PROBABILIDAD	TIEMPO	COSTO	ESTRATEGIA DE MITIGACIÓN
1	Fallo en los servidores que alojan el sistema, tanto en ambientes de desarrollo como de producción.	0,10	5 horas	\$0	Utilizar servidores de respaldo e informar al cliente de la situación.
2	Fallo en las computadoras de los miembros del equipo, lo que causa retrasos en el cronograma y sobrecostos	0,05	2 días	\$1,200	Informar al cliente de la situación y proceder a enviar la computadora a reparación.
3	Retraso en la entrega de las adquisiciones por parte del proveedor	0,03	8 días	\$200	N/A
4	Cambios de requerimientos del cliente que modifiquen el alcance del proyecto	0,5	N/A	\$150	

3. Lecciones aprendidas:

Lecciones aprendidas			
Proyecto	Descripción	Impacto	Soluciones

Verificación de resultados:

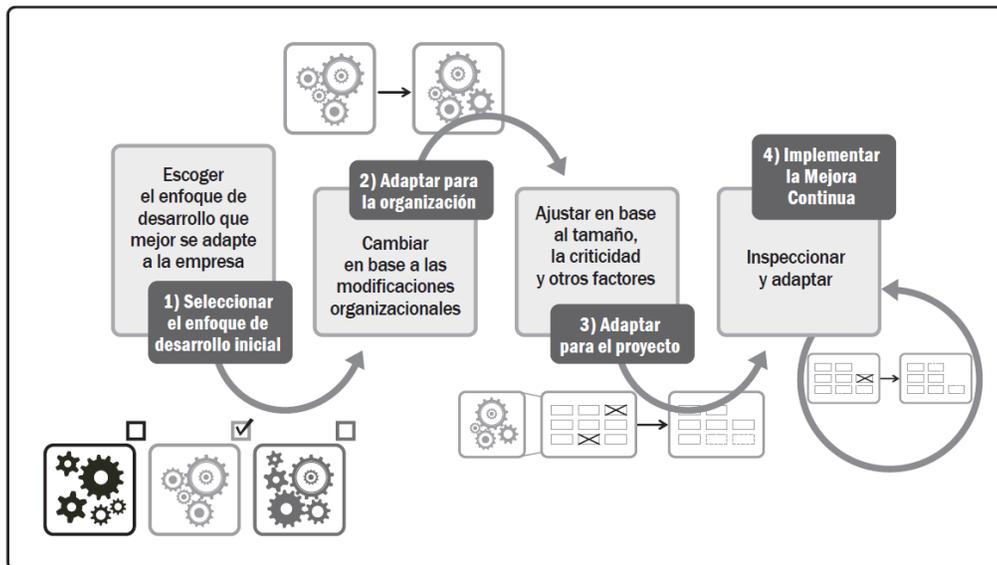
1. Cuando se materializa un riesgo el equipo sabe cómo responder de acuerdo con las estrategias de mitigación establecidas. Si el riesgo no está identificado se consulta al cliente quién es el dueño del proyecto.
2. Los riesgos se monitorean constantemente durante una frecuencia de tiempo.

6.6 Adaptación

Todos los proyectos son diferentes. Cada proyecto incluye particularidades que lo hacen diferente de otros, por lo que en el momento en el que la empresa asuma un nuevo proyecto debe contar con los mecanismos necesarios para adaptar los métodos y herramientas de la organización. La guía PMBOK presenta una serie de pasos en el momento de adaptar un proyecto al contexto.

Figura 34

Proceso de adaptación según la guía PMBOK



Nota. Tomado de *Guía de los Fundamentos para la Dirección de Proyectos y el Estándar para la Dirección de proyectos*, por Project Management.

- **Seleccione el enfoque de desarrollo inicial**

En el momento de iniciar un proyecto seleccione su enfoque de desarrollo. Generalmente, las empresas que gestionan proyectos de software poseen un enfoque de desarrollo que se replica en todos los proyectos. Debido a la naturaleza de los proyectos de software es común que el enfoque de desarrollo sea adaptativo.

Cada empresa determina cuál es la tropicalización que le desea brindar a la metodología que se utiliza para el proyecto en particular. Por ejemplo, un proyecto puede tener una duración de las iteraciones de dos semanas, mientras que otro con diferentes características puede tener una duración de cuatro semanas. Es importante que la empresa determine si es necesario que el enfoque de desarrollo deba adaptarse para algún proyecto en específico.

- **Adaptar para la organización**

Como se mencionó, las empresas generalmente poseen un proceso definido para administrar sus proyectos, aunque no utilicen alguna metodología en específico. En ocasiones, las compañías que desarrollan software a la medida para varios clientes deben adaptarse a las políticas y procedimientos de la empresa del cliente que impliquen cambiar el proceso de administración de proyectos existente en la entidad.

- **Adaptar para el proyecto**

La guía PMBOK (2021) menciona que existen atributos que deben adaptarse para cada proyecto como el **producto y los entregables, el equipo de proyecto y la cultura organizacional**. Para cada atributo se puede realizar una serie de preguntas al cliente y al equipo de proyecto que guiarán el proceso de adaptación de las metodologías que se utilizan en el proyecto en particular.

Tabla 18

Proceso de adaptación según el PMBOK

Adaptación	
Atributo	Pregunta
Producto/Entregable	<p>¿Es adecuado el nivel de rigor del proceso y garantía de calidad?</p> <p>¿El producto es familiar y tangible, como algo fácil de identificar y describir, como un edificio? ¿O algo intangible, como un <i>software</i>?</p> <p>¿Es la tecnología estable y madura o está evolucionando rápidamente y en riesgo de obsolescencia? ¿La duración del proyecto es corta, como semanas o meses, o larga, como años?</p>

	<p>¿Cuán probable es que cambien los requisitos básicos? ¿El elemento comercial del producto es confidencial o privado?</p> <p>¿Puede el equipo del proyecto desarrollarlo gradualmente y obtener retroalimentación de las partes interesadas o es difícil de evaluar hasta que esté casi completo?</p>
Equipo de proyecto	<p>¿Cuántos empleados de tiempo completo y de medio tiempo están en el proyecto?</p> <p>¿Dónde se encuentran principalmente los miembros del equipo? ¿Algunos o todos los miembros del equipo son remotos?</p> <p>¿Dónde está el grupo de apoyo para el equipo y otras partes interesadas? ¿Los miembros del equipo del proyecto tienen experiencia en la industria, organización o asociación?</p> <p>¿Tienen los conocimientos, herramientas y técnicas necesarias para el proyecto? ¿Es posible obtener información frecuente y oportuna de los clientes o representantes de los clientes?</p>
Cultura	<p>¿Hay aceptación, apoyo y entusiasmo por el método de entrega propuesto?</p> <p>¿Existe un alto nivel de confianza en la capacidad y el compromiso del equipo del proyecto para entregar los resultados</p>

	<p>del proyecto? ¿Se confía, se apoya y se alienta a los equipos de proyecto a adoptar y desarrollar su entorno de trabajo, acuerdos y decisiones?</p> <p>¿Los valores y la cultura de la organización son consistentes con el enfoque del proyecto?</p>
--	--

- **Implementar la mejora continua**

Los procesos de adaptación anteriores no deben hacerse únicamente al inicio de cada proyecto, sino que deben refinarse conforme avanza su ejecución y se obtiene retroalimentación apropiada que permita adaptar el proceso de desarrollo. Es importante que durante todo el proceso de administración de proyectos el equipo se involucre para mejorar el proceso, lo anterior permite que el equipo fomente sus habilidades blandas y técnicas. Esta labor no debe ser única de una sola persona, debido a que cada miembro del equipo posee una cosmovisión diferente de las situaciones gracias a la experiencia adquirida durante el tiempo.

7 REFERENCIAS BIBLIOGRÁFICAS

- Arias Chaves, M. (2005). La ingeniería de requerimientos y su importancia en el desarrollo de proyectos de software. InterSedes: Revista de las Sedes Regionales. Universidad de Costa Rica.
- Barrantes, R. (1999). Investigación: Un Camino al Conocimiento. Un Enfoque Cualitativo y Cuantitativo. EUNED.
- Bruegge, B., Dutoit, A.H. (2002). Ingeniería de Software Orientado a Objetos (1ª ed.). Pearson Education.
- Cámara de Comercio de Costa Rica. (S.F). Estadísticas Económicas [Archivo PDF]. http://camara-comercio.com/camara2/wp-content/uploads/2015/11/17_docestadisticasempresas.pdf
- Cerrada. (2006). Introducción a la Ingeniería del Software (3ª Reimpresión.). UNED.
- Cervante Ojeda, J., Gómez Fuentes, M.C. (2012). Taxonomía de los modelos y metodologías del desarrollo de software más utilizados. Universidades. Unión de Universidades de América Latina y el Caribe.
- Chrissis, M.B., Konrad, M., Shrum, S. (2011). CMMI for Development. Guidelines for Process Integration and Product Improvement (3ª ed.). Addison-Wesley.
- Durango, A. (2014). Diseño de Software (2ª ed.). CreateSpace Independent Publishing Platform.
- Espinoza Palma, H (2017) Propuesta de metodología para la gestión de proyectos de software en la empresa Go-Labs [Tesis de Maestría, Instituto Tecnológico de Costa Rica]. <https://hdl.handle.net/2238/7081/>
- Estayno, M., Dapozo, G., Greiner, C., Cuenta Pletsch, L., Pelozo, S. (2009). Caracterización de las PYMES de software de la región NEA orientada hacia un marco de mejora de la calidad. Core. <https://core.ac.uk/download/pdf/15777868.pdf/>
- Fernández Iparraguirre, J.L. (2014). El modelo de Costes en Empresas Comerciales y de Servicios (1ª ed.). ESIC Editorial.

- García, Y.M., Muñoz, M., Mejía, H., Martínez, J.M., Gasca, G.P., Hincapié, J. A (2017) Desarrollo de Herramientas Enfocadas en Ayudar a las PYMES de Desarrollo de Software en la Implementación de Buenas Prácticas de Gestión de Proyectos. Revista Recibe. <https://doi.org/10.32870/recibe.v6i1.63/>
- Gómez Palomo, S.R., Moradela Gil, E. (2020). Aproximación a la Ingeniería del Software (2ª ed.). UNED.
- González Moreno, M., Gómez, C.R., Domínguez Martínez, J.M. (1989). Los Servicios: concepto, clasificación y problemas de medición. Vasca de Economía.
- González Muñoz, M.A (2020) Propuesta de una metodología de trabajo basada en un marco de trabajo ágil para el uso y la aplicación de base datos multiplataforma aplicando los principios y buenas prácticas de la Administración de Proyectos del Project Management Institute, para el centro de desarrollo de Software de la Universidad Latina de Costa Rica, en el año 2020 [Tesis de Maestría, Universidad Latina de Costa Rica]. <https://hdl.handle.net/20.500.12411/337/>
- González, M. (1989). Los servicios: concepto, clasificación y problemas de medición. Dialnet. <https://dialnet.unirioja.es/servlet/articulo?codigo=1317365/>
- Gray, & Larson. (2009). Administración de proyectos. McGraw-Hill Education.
- Idalko (2021). *Jira Dashboards Made Simple: The Why, How and Best Practices*. iDalko. <https://www.idalko.com/jira-dashboards/>
- IEEE. (1998). IEEE Standard for Software Project Management Plans. Software Engineering Standards Committee of the IEEE Computer Society. https://cours.etsmtl.ca/log792/private/restreint/IEEE_1058_Project_Management_Plan.pdf
- ISO (2012). Guidance on project management. International Organization for Standardization (ISO). http://www.isopm.ru/download/iso_21500.pdf
- Jiménez, M. M. (s. f.). *Frecuencia e impacto en la matriz de riesgos*. <https://www.piranirisk.com/es/blog/matriz-de-riesgos-frecuencia-impacto/>
- Jucá Junior, A. D. S. Conforto, E. C., & Amaral, D. C. (2010). Maturidade em gestão de projetos em pequenas empresas desenvolvedoras de software

- do Polo de Alta Tecnologia de São Carlos. *Gestão & Produção*, 17(1), 181–194. <https://doi.org/10.1590/s0104-530x2010000100014/>
- Laporte, C. Y., Renault, A., Alexandre, S., & Crowder, K. V. (2008). 5.3.2 The Development of International Standards for Very Small Enterprises. *INCOSE International Symposium*, 18(1), 620–631. <https://doi.org/10.1002/j.2334-5837.2008.tb00831.x/>
- Loo, W.K., Leau, Y.B., Tham, W.Y., Tan, S. (2012). Software Development Life Cycle AGILE vs Traditional Approaches. *International Conference on Information and Network Technology*. https://www.researchgate.net/publication/268334807_Software_Development_Life_Cycle_AGILE_vs_Traditional_Approaches
- Miranda, L. A. (2017). *Administración de Proyectos* (1.a ed.). Pearson Education.
- Ministerio de Economía, Industria y Comercio. (2021, noviembre). Estado de Situación PYMES en Costa Rica 2021 [Archivo PDF]. <http://reventazon.meic.go.cr/informacion/estudios/2021/PYMES/DIGEPYMES-INF-038-2021.pdf/>
- Ocala (2012). *Metodología de Desarrollo y Mantenimiento de Software para una Fábrica de Software* [Tesis de Maestría: Universidad Nacional de Ingeniería, Lima Perú]. <https://1library.co/document/z31631ey-metodologia-desarrollo-mantenimiento-software-fabrica-software.html>
- Olarte Gervacio, L. (23 de abril de 2018). *Lenguaje de Programación*. <https://conogasi.org/articulos/lenguaje-de-programacion/>
- Pardo, C., García, F., Pino, F., Piattini, M., Universidad Autónoma de Occidente Colombia. (2013). *Producto y proceso: una relación compleja en la ingeniería del software*. *El Hombre y la Máquina*.
- Portny, J.L., Portny S.E. (2022). *Project Management for Dummies* (2ª ed.). John Wiley & Sons.
- Pressman. (2011). *Ingeniería del Software: Un enfoque práctico* (7ª ed.). McGraw Hill.
- PMI (2017). *Guía de los Fundamentos para la Dirección de Proyectos* (6ª ed.). Project Management Institute.

- PMI (2021). Guía de los Fundamentos para la Dirección de Proyectos (7ª ed.). Project Management Institute.
- Rastogi, V. (2015). Software Development Life Cycle Models Comparison, Consequences. International Journal of Computer Science and Information Technologies.
- Rivera Céspedes, O (2017) Propuesta de una metodología de gestión de la calidad que apoye la administración de proyectos de software en la empresa Avantica Technologies [Tesis de Maestría, Instituto Tecnológico de Costa Rica]. <https://hdl.handle.net/2238/9131/>
- Sampieri, R. H., Collado, C. F., Lucio, P. B., Valencia, S. M., & Torres, C. P. M. (2014). Metodología de la investigación. McGraw-Hill Education.
- Schulmeyer, G. G. (2008). Handbook of Software Quality Assurance. Penguin Random House.
- Somerville I. (2011). Ingeniería de Software (9.a ed.). Pearson Education.
- Torres, C. A. B., Bernal, C. A., Buevas, L. E. C., & Palma, O. F. (2010). Metodología de la investigación. Pearson Educación.
- Wright, J. (2020). Scrum: The Complete Guide to the Agile Project Management Framework that Helps the Software Development Lean Team to Efficiently Structure and Simplify the Work & Solve Problems in Half the Time. Independently Published. Efficiently Structure and Simplify the Work & Solve Problems in Half the Time. Independently Published.

ANEXOS Y APÉNDICES

8 Anexos y apéndices

Apéndice A: Cuestionario enviado a las empresas

25/10/22, 11:41

Cuestionario sobre la implementación de estándares de administración de proyectos en las PYMES desarrolladoras de software en...

Cuestionario sobre la implementación de estándares de administración de proyectos en las PYMES desarrolladoras de software en la GAM.

Estimado empleador:

El presente cuestionario tiene como objetivo conocer sobre el proceso de administración de proyectos de software que se lleva a cabo en su empresa.

Los datos que se recolectan en este formulario no tienen el carácter de datos sensibles ni serán publicados a terceros, manteniendo en todo momento el anonimato y la custodia de los mismos. En el proyecto de investigación solo serán presentadas estadísticas y resúmenes de todas las entidades participantes.

Agradezco de antemano la información brindada.

***Obligatorio**

1. ¿Cuál es la cantidad de colaboradores aproximado de su empresa? *

Marca solo un óvalo.

- De 1 a 5
- De 5 a 15
- Más de 15

2. ¿Cuáles metodología de desarrollo de software se utiliza dentro de su empresa? *
Puede seleccionar varias opciones.

Selecciona todas las opciones que correspondan.

- Scrum
 Kanban
 Cascada tradicional
 XP
 Otros: _____

3. ¿Cuáles de las siguientes metodologías de administración de proyectos se utiliza *
en su empresa? Puede seleccionar varias opciones.

Selecciona todas las opciones que correspondan.

- PMI
 CMMI-DEV
 IEEE
 ISO
 Ninguna
 Otros: _____

4. ¿Su relación con la empresa se da a través de la siguiente área? *

Marca solo un óvalo.

- Gerencia
 Área administrativa
 Técnica
 Proyectos de software

5. ¿Ha recibido alguna capacitación sobre administración de proyectos? *

Marca solo un óvalo.

- Sí
- No
- Otros: _____

6. ¿Cuál es su nivel de conocimiento sobre administración de proyectos? *

Marca solo un óvalo.

- Bajo
- Medio
- Alto

7. ¿Qué tipo de software se desarrolla en su empresa? Puede seleccionar varias opciones. *

Selecciona todas las opciones que correspondan.

- De Sistemas (compiladores, editores y herramientas para administrar archivos)
- Aplicación (aplicaciones procesan datos comerciales o técnicos para facilitar las operaciones del negocio o la toma de decisiones administrativas y técnicas)
- Incrustado (Es el tipo de software que reside dentro de un producto o sistema para controlar las características para el usuario final)
- Inteligencia Artificial.
- Otros: _____

8. ¿Cuál de los siguientes aplicativos utiliza para gestionar sus proyectos y darles trazabilidad? *

Marca solo un óvalo.

- JIRA
- Microsoft Project
- Mantis
- Trello
- No utilizamos algún software para gestionar los proyectos
- Otros: _____

Administración del
alcance de los
proyectos de software

Esta sección contiene preguntas sobre como se
administra el alcance de los proyectos de su empresa.

9. ¿Los interesados del proyecto están identificados? *

Marca solo un óvalo.

- Sí
- No
- No sé

10. ¿Existe algún documento que gestione el alcance del proyecto? *

Marca solo un óvalo.

- Sí
- No
- No sé
- Otros: _____

11. A la hora de definir el alcance de sus proyectos ¿Cuáles de los siguientes criterios son tomados en cuenta? Puede seleccionar varias opciones. *

Selecciona todas las opciones que correspondan.

- Requerimientos
- Fechas de Entrega
- Costos
- Entregables
- Exclusiones
- Restriciones
- Ninguna
- Otros: _____

Administración del
Tiempo de los proyectos
de software

Esta sección contiene preguntas sobre como se
administra el tiempo de los proyectos en su empresa.

12. ¿Las actividades del proyecto tienen un tiempo de duración establecido? *

Marca solo un óvalo.

- Sí
- No
- No sé

13. ¿Se desarrolla un cronograma de actividades? *

Marca solo un óvalo.

- Sí
- No
- No sé

14. ¿Se elabora algún documento donde se establezcan las responsabilidades de cada miembro del equipo de acuerdo con las tareas asignadas? *

Marca solo un óvalo.

- Sí
 No
 No sé

15. ¿Se mide constantemente el rendimiento del equipo de trabajo? *

Marca solo un óvalo.

- Sí
 No
 No sé

16. ¿Cómo se estima el tiempo de las diferentes actividades del proyecto? Puede seleccionar varias opciones. *

Selecciona todas las opciones que correspondan.

- Puntos de Historia
 Diagrama de PERT
 COCOMO II
 Ninguna
 Otros: _____

Administración del Costo y Recursos de los proyectos de software

Esta sección contiene preguntas sobre como se administran los costos de los proyectos en su empresa.

17. ¿Los clientes se mantienen informados sobre los avances del proyecto? *

Marca solo un óvalo.

- Sí
 No
 No sé

18. ¿Existe un plan de adquisiciones dentro de los proyectos? *

Marca solo un óvalo.

- Sí
 No
 No sé

19. ¿Se cuenta con un documento donde se detallan los costos del proyecto? *

Marca solo un óvalo.

- Sí
 No sé
 Otros: _____

20. ¿Se desarrolla un presupuesto para el proyecto de manera detallada? *

Marca solo un óvalo.

- Si
 No
 No sé

21. ¿Los costos del proyecto son monitoreados constantemente? *

Marca solo un óvalo.

- Sí
- No
- No sé

Administración
del Riesgo

Esta sección contiene preguntas sobre como se administran los riesgos de los proyectos en su empresa.

22. ¿Se documentan los riesgos de los proyectos? *

Marca solo un óvalo.

- Si
- No
- No sé

23. ¿Se le da monitoreo a los riesgos de los proyectos? *

Marca solo un óvalo.

- Sí
- No
- No sé

24. ¿Cuando un riesgo se materializa se cuenta con las mecanismos necesarias para mitigar el riesgo? *

Marca solo un óvalo.

- Sí
 No
 No sé

Cierre de los proyectos de software

Esta sección contiene preguntas sobre el proceso de cierre de los proyectos en su empresa.

25. ¿Existe alguna documentación para transferir el producto, servicio o resultado final? *

Marca solo un óvalo.

- Sí
 No
 No sé

26. ¿Se documentan las lecciones aprendidas del proyecto? *

Marca solo un óvalo.

- Sí
 No
 No sé

Calidad de los proyectos de software

Esta sección contiene preguntas sobre el proceso de control de calidad de los proyectos en su empresa.

27. ¿Se cuenta con un procedimiento de calidad para los productos? *

Marca solo un óvalo.

- Sí
 No
 No sé

28. ¿Se mantiene un registro de las habilidades técnicas de cada uno de los miembros del equipo? *

Marca solo un óvalo.

- Sí
 No
 No sé

29. ¿La calidad de los productos es probada constantemente? *

Marca solo un óvalo.

- Sí
 No
 No sé

30. ¿Se realiza un informe de calidad del proyecto? *

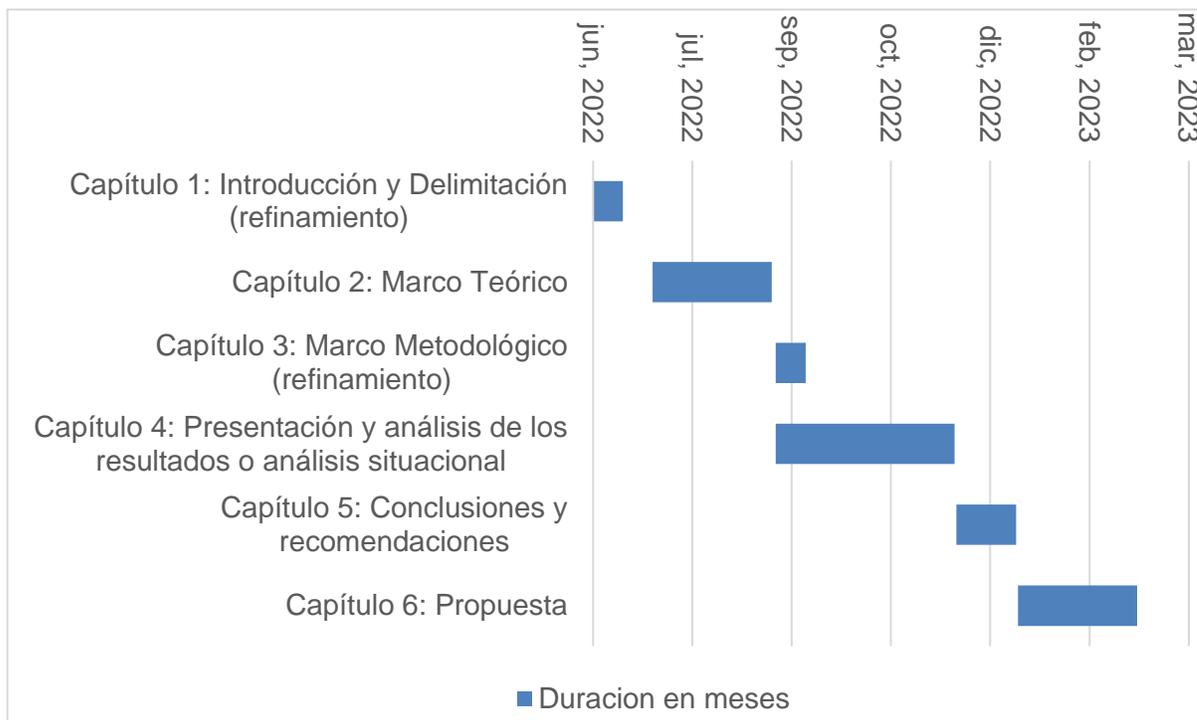
Marca solo un óvalo.

- Sí
 No
 No sé

Anexo A: Cronograma de trabajo

Actividad	Inicio aproximado	Fin aproximado
Capítulo 1: Introducción y delimitación (refinamiento)	1/6/2022	15/6/2022
Capítulo 2: Marco teórico	1/7/2022	30/8/2022
Capítulo 3: Marco metodológico (refinamiento)	1/9/2022	15/9/2022
Capítulo 4: Presentación y análisis de los resultados o análisis situacional	1/9/2022	31/11/2022
Capítulo 5: Conclusiones y recomendaciones	1/12/2022	31/12/2022
Capítulo 6: Propuesta	1/1/2023	29/02/2023

Anexo B: Diagrama de Gantt



CARTA DE AUTORIZACIÓN DE USO



Universidad Técnica Nacional

Anexo III

CARTA DE AUTORIZACIÓN PARA USO Y MANEJO DE LOS TRABAJOS FINALES

DE GRADUACIÓN UNIVERSIDAD TÉCNICA NACIONAL

(Trabajo Individual)

Ciudad,

Fecha.

Señores/as

Vicerrectoría de Investigación. Sistema Integrado de Bibliotecas y Recursos Digitales

Estimados señores/as:

Yo Daniel Bello Ramírez portador (a) de la cédula de identidad número 117170075. En mi calidad de autor (a) del trabajo de graduación titulada:

PROPUESTA DE UNA GUÍA DE ADMINISTRACIÓN DE PROYECTOS BASADA EN ESTÁNDARES INTERNACIONALES PARA MICRO Y PEQUEÑAS EMPRESAS DESARROLLADORAS DE SOFTWARE DE LOS TRES CANTONES CON MAYOR CANTIDAD DE EMPRESAS EN LA GRAN ÁREA METROPOLITANA DE COSTA RICA

El cual se presenta bajo la modalidad de, marque una opción:

Proyecto de Graduación

Tesis de Graduación

Presentado en la fecha 04/05/2023, autorizo a la Universidad Técnica Nacional, sede central, para que mi trabajo pueda ser manejado de la siguiente manera:

Autorizo
Ver capítulo V, disposiciones finales, artículo 41 (O aquel que refiera a derechos patrimoniales)

Marque con una X o un ✓	
Conservación de ejemplares para préstamo y consulta física en biblioteca.	X
Inclusión en el catálogo digital del SIBIREDI (Cita catalográfica)	X
Comunicación y divulgación a través del Repositorio Institucional	X
Resumen (Describe en forma breve el contenido del documento)	X
Consulta electrónica con texto protegido	X
Descarga electrónica del documento en texto completo protegido	X
Inclusión en bases de datos y sitios web que se encuentren en convenio con la Universidad Técnica Nacional contando con las mismas condiciones y limitaciones aquí establecidas.	X
Divulgación del resumen en el Repositorio UTN, con una cantidad de 200 a 500 palabras	X

Por otra parte, declaro que el trabajo que aquí presento es de plena autoría, es un esfuerzo realizado de forma personal, académica e intelectual con plenos elementos de originalidad y creatividad. Garantizo que no contiene citas, ni transcripciones de forma indebida que puedan devenir en plagio, pues se ha utilizado la normativa vigente de la American Psychological Association (APA). Las citas y transcripciones utilizadas se realizan en el marco de respeto a las obras de terceros. La responsabilidad directa en el diseño y presentación son de competencia exclusiva, por tanto, eximo de toda responsabilidad a la Universidad Técnica Nacional.

Consciente de que las autorizaciones no reprimen mis derechos patrimoniales como autor del trabajo. Confío en la que Universidad Técnica Nacional respete y haga respetar mis derechos de propiedad intelectual.

Firma del estudiante:

DANIEL
BELLO
RAMIREZ
(FIRMA)

Firmado digitalmente por
DANIEL BELLO
RAMIREZ (FIRMA)
Fecha: 2023.05.04
20:56:16 -06'00'

Cédula: 117170075

Día: 4/5/2023