



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE CIENCIAS

**IDENTIFICACIÓN Y MODELADO DE ALPHAS
BASADAS EN EL PMBOK PARA EL SEGUIMIENTO
Y CONTROL DE PROYECTOS DE SOFTWARE**

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

**LICENCIADA EN CIENCIAS DE LA
COMPUTACIÓN**

P R E S E N T A:

BRENDA SANTIAGO MARTÍNEZ

DIRECTOR DE TESIS:

DR. MIGUEL EHÉCATL MORALES TRUJILLO

2016

Ciudad Universitaria, D. F.





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Hoja de Datos del Jurado

1. Datos del alumno
Santiago
Martínez
Brenda
57 57 02 81
Universidad Nacional Autónoma de México
Facultad de Ciencias
301090140
2. Datos del tutor
Dr
Miguel Ehécatl
Morales
Trujillo
3. Datos del sinodal 1
Dra
Hanna Jadwiga
Oktaba
4. Datos del sinodal 2
Dra
Amparo
López
Gaona
5. Datos del sinodal 3
M. en C.
Eréndira Miriam
Jiménez
Hernández
6. Datos del sinodal 4
M. en I.
Gerardo
Avilés
Rosas
7. Datos del trabajo escrito
Identificación y modelado de ALPHAs basadas en el PMBOK para el seguimiento y control de proyectos de software
78 p
2016

*Dedicado a mis padres, Vicente Santiago y Ma. Guadalupe Martínez, a mis
hermanos, Vicente y Oscar, y a Amalia.*

AGRADECIMIENTOS

En momentos como éste, haces un recuento de cada instante transcurrido para llegar al día de hoy, en donde finaliza una larga etapa llena de variadas emociones y no queda más que agradecer por cada momento y a cada persona presente en todo éste camino.

Culminar un trabajo de la magnitud que significa una tesis, es un arduo proceso que involucra un sinnúmero de personas y que sin ellas este logro no sería posible. Iniciaré agradeciendo a la Universidad Nacional Autónoma de México, que ha sido forjadora de conocimiento invaluable para mi desarrollo profesional, especialmente a la Facultad de Ciencias, institución que me aportó los fundamentos necesarios para prepararme como profesionista en ésta carrera.

En mi transcurso por la facultad, conocí a diversos amigos y compañeros que hicieron que esa etapa estuviera llena de momentos grandiosos. Gracias Lili, Pame, Adriana, Sandra, Daniel, Dai, Miguel, Iván, Samuel, Noé.

Quiero agradecer de manera especial a Liliana Cruz, mi compañera y amiga de toda la carrera, juntas pasamos bastantes horas de estudio, clases, prácticas y de una gran cantidad de aventuras que hicieron maravillosa mi estancia en la facultad.

Gracias Pame Ávila, por ser mi apoyo durante esos años y por todos los consejos brindados, los cuales casi nunca seguía jeje, pero son recuerdos muy gratos.

Gracias Carlos Cavazos, por todos los momentos y las enseñanzas que me brindaste.

Llevar a cabo, el presente trabajo, fue sin lugar a duda por las enseñanzas, consejos y guía del Dr. Miguel Ehécatl Morales Trujillo, a quien tuve la fortuna de conocer en el momento preciso de mi vida para lograr finalmente este reto denominado “Titulación”. Mi sincero agradecimiento por el apoyo otorgado.

Agradezco también a mis sinodales que aportaron su experiencia para enriquecer éste trabajo: Dra. Hanna Oktaba, Dra. Amparo López Gaona, M. en I. Gerardo Avilés Rosas y M. en C. Eréndira Miriam Jiménez Hernández.

Expreso mi agradecimiento nuevamente a la Dra. Amparo, por el apoyo y consejos durante todos estos años de recorrer este camino lleno de tropiezos pero también de alegrías, y que por fin puedo decir “lo logré”.

También mi más sincero agradecimiento a la Lic. Leticia Venegas Cruz, por su aportación en la mejora de este trabajo, y las enseñanzas sobre la administración de proyectos.

Gracias, L.C.C. Rubén Sáenz González por permitirme pertenecer al gran equipo de trabajo de la CCUD, que me ha dejado grandes experiencias. Así como por la gran oportunidad brindada, que marcó el inicio de éste trabajo, y que me permitió conocer las bases de la administración de proyectos, que son en gran medida el sustento de ésta tesis.

A todo el equipo de la Oficina de Desarrollo Tecnológico, de quienes he aprendido algo diferente, gracias.

Gracias, Daniel Juárez, por esa manera tan tuya de ser, los consejos, las risas y por ser amigos y compañeros de trabajo desde hace ya bastante tiempo.

Gracias, Daniel Acevedo, por tu amistad y el apoyo que me brindaste.

Gracias, David Zaragoza, por tu amistad, las risas y las pláticas.

Gracias, Karen Zavala, por ser una gran compañera de trabajo, de quien he aprendido mucho durante este tiempo. Recorrimos juntas parte de éste proceso, y te agradezco el apoyo en esta etapa y por supuesto por tu amistad.

Durante mi etapa posterior al finalizar la carrera, conocí y trabajé con varias personas que me apoyaron y creyeron en mí, y que además de ser pieza importante en mí desarrollo profesional me brindaron también su amistad. Gracias, DGTIC- Acervos Digitales y todo el equipo que lo conforma, especialmente expreso mi sincero agradecimiento a Guillermo Chávez por la oportunidad de pertenecer a ese gran equipo.

En mi vida, he estado rodeada de personas que han sido un soporte para lograr lo que me he propuesto hasta el día de hoy. Me han llenado de alegría, y amor durante muchos momentos y gracias a ellos he podido culminar ésta etapa.

Gracias, Oscar Hernández, has sido uno de los grandes apoyos en varios momentos. Este proceso no fue fácil pero con tu cariño y comprensión permitieron menguar muchos de los obstáculos presentados y agradezco enormemente tu paciencia principalmente en los momentos de estrés. Por acompañarme en esta etapa llena de un sinfín de emociones, eres una gran persona, y esta meta es también gracias a ti.

Marielia López, mi gran amiga, en los casi 15 años que tenemos de conocernos siempre me has demostrado tu cariño, tu apoyo, comprensión. Recuerdo con cariño todas las ocasiones que te platicaba de cada intento por lograr éste objetivo, y siempre estuviste ahí conmigo, escuchándome, apoyándome, aconsejándome. Gracias amiga, por permanecer a mi lado durante todo este tiempo, y gracias también a tu hermosa familia.

Gracias, Lili Sánchez, es un gusto tenerte en mi vida como mi amiga, por apoyarme en esta etapa, y darme tus consejos, escucharme y reír juntas. Eres una de las personas que siempre tiene las palabras ideales para cualquier momento, nuevamente muchas gracias.

Finalmente, agradezco profundamente a mi familia, gracias a mis papás por su amor incondicional. Siempre han sido los mejores guías en esta vida, y que han hecho que hoy pueda lograr ésta meta. A mis queridos hermanos, Vicente, mi hermano mayor, que ha estado conmigo siempre, que repetidas ocasiones me hacía enojar pero también reír con todas sus ocurrencias, por las veces que ayudaste a traducir los textos en inglés que no entendía, gracias!. Oscar, estuviste conmigo varias madrugadas, ayudándome, explicándome qué era una *base de datos*, un *varchar*, un *tipo de dato*, una *clase*, etc., en todas esas tareas y proyectos finales que no entendía, gracias por apoyarme. También aprovecho para agradecer a mis cuñadas Lety Juárez y Judith Nieto, ambas son geniales. A mi nana, Amalia, que durante toda mi vida ha estado a mi lado apoyándome y cuidándome, y por supuesto a mi querido Coqui, mi gordito bello.

¡Infinitas gracias por todo lo que me han dado!

Índice general

Índice de figuras	iii
Índice de tablas	iv
Introducción	1
Contexto	1
Problema.....	3
Objetivo.....	3
Solución Propuesta	3
Estructura de la tesis.....	4
Antecedentes	5
1.1. Ingeniería de Software	5
1.2. ESSENCE – Kernel and Language for Software Engineering Methods	7
1.2.1. Arquitectura.....	8
1.2.2. Kernel.....	8
1.2.3. Lenguaje.....	15
1.3. PMBOK.....	16
1.4. Otros estándares relacionados	18
Metodología	20
2.1. Guía genérica para construir un ALPHA.....	20
2.2. Selección de fuentes de información y cuerpos de conocimiento	21
2.3. Estructura básica de las ALPHAs	27
2.4. Verificación y validación	28
Construcción de ALPHAs	30
3.1. ALPHA Integración	33
3.2. ALPHA Alcance	35
3.3. ALPHA Tiempo	37
3.4. ALPHA Costos	39
3.5. ALPHA Calidad	41
3.6. ALPHA Recursos Humanos	43
3.7. ALPHA Comunicaciones	45
3.8. ALPHA Riesgos	47
3.9. ALPHA Adquisiciones.....	49
3.10. ALPHA Stakeholders	51
Resultados	54

4.1. Verificación y validación	54
4.2. ALPHAs	58
4.3. Guía de uso de ALPHAs	60
4.3.1. Estructura del ALPHA	60
4.3.2. Recomendaciones	61
4.4. Observaciones finales	62
Conclusiones	63
Anexos	65
Glosario	67
Bibliografía	68

Índice de figuras

- Figura 1.1. Capas de la ingeniería de software.
- Figura 1.2. Arquitectura del método.
- Figura 1.3. The kernel Alphas [imagen] (2014)
- Figura 1.4. The kernel Activity Spaces [imagen] (2014)
- Figura 1.5. The kernel Competencias [imagen] (2014)
- Figura 2.1. Descripción General de la Gestión del Alcance del Proyecto [imagen] (2013)
- Figura 2.2. Grupos de procesos del ciclo de vida [imagen] (2008)
- Figura 2.3. Estructura de un ALPHA
- Figura 2.4. Formato para el registro de defectos.
- Figura 3.1. ALPHA Integración
- Figura 3.2. ALPHA Alcance
- Figura 3.3. ALPHA Tiempo
- Figura 3.4. ALPHA Costos
- Figura 3.5. ALPHA Calidad
- Figura 3.6. ALPHA Recursos Humanos
- Figura 3.7. ALPHA Comunicaciones
- Figura 3.8. ALPHA de Riesgos
- Figura 3.9. ALPHA Adquisiciones
- Figura 3.10. ALPHA Stakeholders
- Figura 4.1. Encuestas
- Figura 4.2. Defectos registrados
- Figura 4.3. ALPHAs
- Figura 4.4. Estructura ALPHA Tiempo
- Figura 4.5. Listas de verificación

Índice de tablas

Tabla 1.1.	Factores de éxito en un proyecto
Tabla 3.1.	Grupos de procesos y áreas de conocimiento.
Tabla 3.2.	Lista de verificación para el ALPHA Integración
Tabla 3.3.	Lista de verificación del ALPHA Alcance
Tabla 3.4.	Lista de verificación del ALPHA Tiempo
Tabla 3.5.	Lista de verificación del ALPHA Costos
Tabla 3.6.	Lista de verificación del ALPHA Calidad
Tabla 3.7.	Lista de verificación del ALPHA Recursos Humanos
Tabla 3.8.	Lista de verificación del ALPHA Comunicaciones
Tabla 3.9.	Lista de verificación del ALPHA Riesgos
Tabla 3.10.	Lista de verificación del ALPHA Adquisiciones
Tabla 3.11.	Lista de verificación del ALPHA Stakeholders

Introducción

Contexto

Desde la antigüedad, el ser humano ha creado grandes civilizaciones, aldeas, templos, medios de transporte, navegación, etc., que de acuerdo a la época perteneciente, sus objetivos, técnicas y necesidades han ido evolucionando.

Desde aquellos tiempos se han llevado a cabo grandes logros, mediante los cuales se ha dado solución a problemas acontecidos en cada época, en la mayoría de los casos con éxito pero otros, por variadas razones, han sido fallidos. Sin embargo, a pesar de las limitantes presentes en cada época y al entorno específico de la realización de cada proyecto, la planificación de cada uno ha tenido como eje primordial el tiempo y el esfuerzo requeridos para su implementación. Estos conceptos con el paso del tiempo han ido evolucionando y ampliando su conjunto con el propósito de mejorar la manera de emprender un proyecto.

Hoy en día, gracias a esa experiencia transmitida a través de los años se puede deducir que las áreas principales en el desarrollo de cualquier proyecto son el tiempo, alcance y recursos designados para llevarlo a cabo.

El Project Management Institute¹ provee la siguiente definición de *proyecto*:

¹ PMI. Project Management Institute, organización internacional que asocia a profesionales relacionados con la Administración de Proyectos.

“Esfuerzo temporal que se lleva a cabo para crear un producto, servicio y resultado único”

En la actualidad, en nuestra vida diaria nos hemos visto involucrados en proyectos de diversas índoles, ya sea en el ámbito personal o profesional, los cuales para lograr las metas los dividimos en diversas fases y, en muchos de los casos, se ejecutan con herramientas y técnicas basadas en el conocimiento empírico.

En proyectos específicos, por razones de complejidad, tamaño, objetivos y características propias del producto que se requiere obtener, además de los elementos que son equiparables a cualquier proyecto, existen ciertos aspectos adicionales que hacen que sea único y es en éste caso en el que se encuentran los proyectos relacionados con las tecnologías de la información (TI).

Las organizaciones hoy en día, sin importar su giro o tamaño, han ido incrementando el uso de las TI, y específicamente de software que permita mejorar sus procesos en los distintos niveles de la organización.

El software a partir de los últimos años ha tomado una importancia considerable, en cualquier ámbito de nuestras vidas hemos estado en contacto, directa o indirectamente en este amplio mundo. Ya sea en el ámbito cultural, comercial y en la mayoría de nuestras actividades cotidianas el uso de un herramienta que permita realizarlas con mayor rapidez y eficiencia genera que las personas basen un alto porcentaje de sus tareas cotidianas en dichos sistemas.

El desarrollo de un proyecto de software, dadas las características particulares que posee y la importancia de lograr el éxito, se puede basar en distintas metodologías que sirven de guía para su implementación. El papel que toma contar con una metodología es esencial en un proyecto, ya que es la metodología la que dicta las actividades a realizar.

Las metodologías se pueden dividir en dos grandes enfoques, en un principio el desarrollo de software se basaba en las llamadas metodologías tradicionales, definidas por la generación exhaustiva de documentación, conformadas por grandes etapas y que hacen un mayor énfasis en la planificación y control del proyecto, pero en una baja adaptabilidad a los cambios. Este tipo de metodologías tienen como distintivo que son fuertemente basadas en procesos.

Por otra parte, en el 2001 nace el término ágil, este “nuevo” tipo de metodologías surgen con el fin de ofrecer una alternativa a lo tradicional, centrándose en el factor humano, la colaboración en equipo y la flexibilidad ante el cambio.

Un proyecto, cualquiera que sea su naturaleza, debe abordarse con los fundamentos necesarios que permitan acercarse en mayor medida al éxito, con las herramientas y técnicas adecuadas que proporcionen un adecuado control y monitoreo del mismo. Razón por la cual resulta fundamental el elegir una metodología adecuada.

Problema

Actualmente estamos inmersos en un mundo, en donde la creciente demanda de software que se ha desarrollado en los últimos años requiere cada vez de sistemas más eficientes, construidos en el menor tiempo posible, con altos estándares de calidad y que se mantengan alineados con la misión, visión y estrategias de negocio de la organización.

Aunado a lo anterior, factores como los cambios constantes en las condiciones económicas, en el mercado tecnológico, la globalización y la intensa competencia, propicia que muchos de los proyectos de software no logren completarse exitosamente por diversas razones. Entre estas razones, destaca el poco o nulo control que se tiene sobre los aspectos esenciales que deben monitorearse para evaluar el estado del proyecto.

Por otra parte, los cambios constantes en un proyecto de software generan que la vigilancia que se debería tener en los aspectos esenciales resulte en un proceso pesado y en ocasiones ineficiente, derivando en un aumento en la probabilidad de no cumplir con las expectativas planeadas.

Objetivo

Definido el contexto anterior, esta tesis tiene como objetivo identificar y describir el conjunto de elementos esenciales (ALPHAs²) en el desarrollo de un proyecto de software con el fin de que los involucrados puedan conocer y evaluar el progreso del mismo, sustentando su análisis en los principios del PMBOK³.

Solución Propuesta

Con el propósito de ofrecer una solución ligera, se modelarán los elementos fundamentales, ALPHAs, que deben monitorearse en un proyecto de software para tener un control más preciso del estado del mismo. De acuerdo a estos elementos, se proveerá de un conjunto de puntos para verificar su cumplimiento.

Esto es, cada ALPHA estará compuesta por un conjunto de estados relacionados entre sí, mismos que permitirán establecer, en primera instancia, el progreso de un elemento particular, y en consecuencia, al analizar el estado de cada ALPHA involucrada en el proyecto, será posible establecer el estado global del proyecto.

² Acrónimo de Abstract-Level Progress Health Attribute. Elemento esencial del esfuerzo de ingeniería de software para la valoración de su progreso y salud.

³ Acrónimo de Project Management Body of Knowledge. Guía para la administración de proyectos que describe normas, procesos y prácticas.

Estructura de la tesis

El trabajo está organizado de la siguiente manera: el Capítulo 1 presenta los antecedentes en que se basa el trabajo, se describe inicialmente al estándar ESSENCE⁴ y al PMBOK, de igual manera se abordan algunos estándares relacionados con el desarrollo de software. En el capítulo 2 se describe la metodología planteada para la construcción y modelado de un ALPHA, así como su estructura básica. También se presentan las fuentes de información y cuerpos de conocimiento utilizados, así como los métodos de verificación y validación para cada ALPHA. El capítulo 3 presenta el conjunto de ALPHAs presentes en un proyecto basadas en las áreas de conocimiento del PMBOK. Es en esta sección donde se modela cada una de las ALPHAs propuestas, así como su estructura y lista de verificación respectiva. El capítulo 4 presenta los resultados obtenidos. Finalmente, se presentan las conclusiones que se desprenden del análisis del trabajo desarrollado.

⁴ Estándar del Object Management Group, ESSENCE -Kernel and Language for Software Engineering Methods 1.0, 2014. Es una base teórica que proporciona definiciones y descripciones, y un lenguaje de los métodos de ingeniería de software.

Capítulo 1

Antecedentes

En el área de las TI existen diversas metodologías y estándares que guían a los profesionales y les proveen de buenas prácticas para su adopción dentro de las organizaciones ofreciendo una amplia variedad de opciones de acuerdo al tipo de proyecto. Específicamente en el área de desarrollo de software se han realizado esfuerzos para generar métodos y prácticas que apoyen la labor de los desarrolladores, analistas, testers, administradores de proyectos y profesionales afines al área con el objetivo de generar un producto de calidad.

Bajo este contexto, es importante definir en primera instancia a la disciplina que es base fundamental en el desarrollo de cualquier proyecto de software, la Ingeniería de Software, descrito en el presente capítulo, posteriormente se muestra ESSENCE, estándar que define los conceptos clave de la Ingeniería de Software y que da la pauta para abordar en un futuro las ideas principales de este trabajo. También se presentarán los conceptos y áreas que conforman a la Guía de los Fundamentos de la Administración de Proyectos, mejor conocido como PMBOK y otros estándares relacionados al desarrollo de software.

1.1. Ingeniería de Software

Siendo el software un producto complejo, intangible y con características propias, hacen que se único e irreplicable, por esta razón los profesionales en el área buscan continuamente nuevas ideas, herramientas y técnicas que ayuden a mejorar sus prácticas con la finalidad de construir un mejor producto que satisfaga en su totalidad los requerimientos del usuario.

Desarrollar software es un proceso altamente complejo por tal motivo es fundamental tener como base una serie de factores, actividades, métodos y prácticas que provean de un soporte

para la adecuada gestión y control del proceso, siendo la Ingeniería de Software el pilar esencial para su construcción y mantenimiento.

El IEEE⁵ define a la Ingeniería de Software como:

“La aplicación de un enfoque sistemático, disciplinado y cuantificable para el desarrollo, operación y mantenimiento del software”

La Ingeniería de Software se compone por un conjunto de capas base [1] (Figura 1.1):

- **Calidad:** Fundamento en que se apoya para desarrollar enfoques más eficaces, siendo el compromiso con la calidad base esencial para la ingeniería de software.
- **Proceso:** Define una estructura para la base de administración de proyectos de software, estableciendo un contexto y referencias para la aplicación de métodos.
- **Métodos:** Proporcionan la manera de construir técnicamente el software. Se basan en un conjunto de principios fundamentales en cada área tecnológica.
- **Herramientas:** Proporcionan un apoyo automatizado para el proceso y los métodos, denominadas CASE⁶.



Figura 1.1 Capas de la ingeniería de software.

Fuente: Adaptado de Pressman, Roger S., Ingeniería del software. Un enfoque práctico

Dado lo anterior, la ingeniería de software incluye procesos, métodos y herramientas para generar software sustentados en bases de calidad presente en todos los niveles del desarrollo y así como del producto generado.

Actualmente los profesionistas en el área han desarrollado estándares como lo son UML (Unified Modeling Language) [3], CMMI (Capability Maturity Model Integration) [4], Scrum [5], XP [6], Kanban [7], etc., que a su vez se han adaptado a sus necesidades para crear un entorno único de trabajo útil para el equipo y el logro de sus objetivos.

Sin embargo, el inconveniente surge al crear entornos únicos y específicos de difícil comparación con otros métodos utilizados en las organizaciones, lo que complica la

⁵ Institute of Electrical and Electronics Engineers

⁶ Computer Aided Software Engineering

preservación de tal entorno para un uso posterior. Debido a esto, es relevante dar mayor importancia a los elementos esenciales presentes en la ingeniería de software.

1.2. ESSENCE – Kernel and Language for Software Engineering Methods

En 2009, Ivar Jacobson, Bertrand Meyer y Richard Soley fundan SEMAT⁷, quienes en su afán de mejorar, proponen tomar acción para la creación de una teoría general para la ingeniería de software. SEMAT identificó los obstáculos que aquejan hoy en día derivado de prácticas inmaduras en el área como son, [8]:

- El uso de metodologías por seguir una moda y no por ser una disciplina.
- La ausencia de una base teórica.
- Gran cantidad de métodos y sus variantes que son poco entendibles.
- La ausencia de evaluaciones y validaciones experimentales creíbles.
- La división entre prácticas de la industria y la investigación académica.

Esta iniciativa propone la creación en un conjunto de elementos ampliamente aceptados pero lo suficientemente extensibles para su uso en cualquier contexto, naciendo así ESSENCE [2].

ESSENCE es una especificación que define un kernel y un lenguaje para la creación, uso y mejora de los métodos de ingeniería de software. Las características que tiene son:

- Ofrece una base común para los esfuerzos de ingeniería de software de cualquier tamaño con la facilidad para extender el kernel.
- Apoyo para los practicantes en el desarrollo de su trabajo mediante una guía basada en prácticas.
- Se enfoca principalmente en el uso de los métodos y no en descripciones exhaustivas. En este caso, el uso de las ALPHAs permite medir la salud y progreso del proyecto.
- Permite la construcción de métodos mediante la composición de prácticas que pueden ser fácilmente adaptados a las necesidades del equipo y proyecto.
- Soporta metodologías ágiles.
- Aplica el principio de separar los aspectos preocupantes (SoC⁸) y se enfoca en los que son más importantes.

⁷ Software Engineering Method and Theory, <http://www.semat.org>

⁸ Separation of Concerns

1.2.1. Arquitectura

Los elementos clave de ESSENCE radican en cuatro niveles, ver Figura 1.2:

- Métodos. Un método es una composición de prácticas coherentes, consistentes y completas con un propósito específico y que los practicantes deben cumplir bajo ciertas restricciones.
- Prácticas. Guía de trabajo que se repite para obtener un resultado originado de una entrada⁹ y lo transforman en una salida. El conjunto de prácticas debe contribuir al objetivo del método.
- El kernel. Integrado por los elementos esenciales de la ingeniería de software, y comunes en cualquier método.
- El lenguaje. Es el lenguaje de dominio específico para definir los métodos, prácticas y kernels.

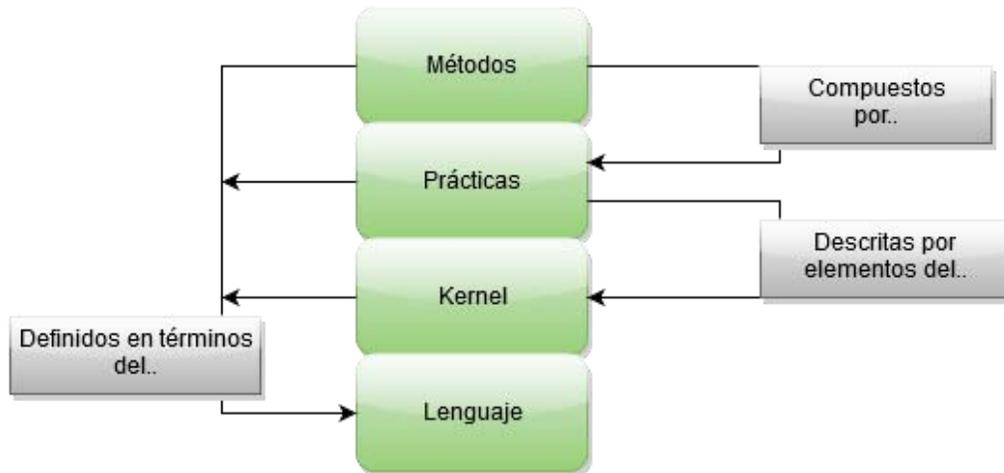


Figura 1.2. Arquitectura del método [imagen] (2014)

Fuente: Adaptado de Kernel and Language for Software Engineering Methods, Essence v1.0

1.2.2. Kernel

El kernel ó núcleo visto desde cualquier enfoque constituye parte fundamental de algo, en este caso es la esencia de la ingeniería de software. Es un conjunto de definiciones simples y ligeras que capturan la parte central de la ingeniería de software, conformando así una base común para la definición de prácticas del desarrollo de software, se caracteriza por ser escalable y extensible.

Está compuesto por tres elementos fundamentales:

⁹ Producto de trabajo, tales como documentos, diagramas y condiciones.

- ALPHAs. Son una descripción de los elementos esenciales con los que se trabaja así como su relevancia para la evaluación del progreso y salud del desarrollo de software.
- Espacios de actividades. Representación de las cosas que se hacen durante el desarrollo, mantenimiento y soporte de sistemas de software.
- Competencias. Habilidades que se requieren para llevar a cabo las tareas de ingeniería de software.

Las áreas de interés del kernel son: cliente, solución y esfuerzo.

En lo referente a *cliente* se incluye lo relacionado al uso actual y explotación del sistema de software que se va a desarrollar.

En la *solución* se incluye lo relacionado a las especificaciones y el desarrollo del sistema de software.

En el *esfuerzo* se incluye lo relacionado al equipo y forma de trabajo.

i. ALPHAs

Son los elementos con los que se trabaja, captura los conceptos clave en la ingeniería de software para la evaluación del progreso y salud de los esfuerzos de software, integrando así la base para definir los métodos y prácticas.

Un ALPHA contiene un conjunto de estados predefinidos para evaluar el progreso y cada estado tiene una lista de verificación, esta lista contiene los elementos que deben cumplirse para completar un estado.

Dado que ESSENCE se basa en el principio de ser flexible y extensible, los estados de un ALPHA pueden variar de acuerdo a las prácticas definidas, es decir, que puede iterarse sobre un estado o múltiples estados, que su progreso no necesariamente es lineal.

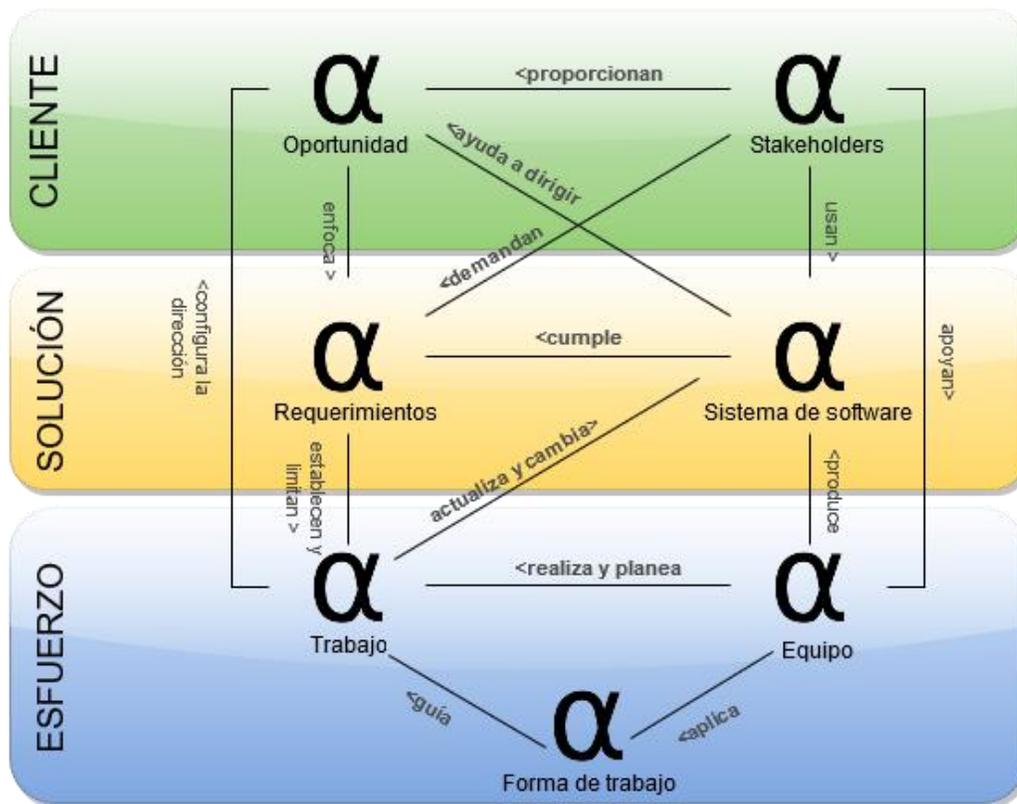


Figura 1.3. The kernel Alphas [imagen] (2014)

Fuente: Adaptado de Kernel and Language for Software Engineering Methods, Essence v1.0

El kernel está compuesto por siete ALPHAS (Ver Figura 1.3), los cuales están concentrados en las áreas específicas de acuerdo a sus características. En *cliente* se encuentran Oportunidad y Stakeholders; en *solución* los Requerimientos y el Sistema de software; en *esfuerzo* el Trabajo, Equipo y Forma de trabajo:

- Oportunidad. El motivo por el cual surge la necesidad de la creación o mejora a un sistema de software. Los estados que lo componen son:
 - Identificado.
 - Solución confirmada
 - Valor establecido
 - Viable
 - Dirigida
 - Beneficios acumulados

Está relacionado con Requerimientos, ya que provee el enfoque central para el desarrollo de los requerimientos del sistema de software.

- Stakeholders. Las personas involucradas que afectan o son afectadas por el desarrollo del sistema de software. Sus estados son:

- Reconocidos
- Representados
- Integrados
- De acuerdo
- Satisfechos por el desarrollo
- Satisfechos en el uso

Los stakeholders plantean la oportunidad para la creación o mejora de un sistema de software involucrándose en el desarrollo desde varias perspectivas, ya sea siendo parte del equipo o estableciendo los requisitos que el software debe cumplir, así como en el uso del mismo.

- Requerimientos. Parte fundamental para iniciar con el sistema de software, son las necesidades que surgen de los stakeholders y que deben verse reflejadas en el sistema. Los estados son:
 - Concebido
 - Delimitado
 - Coherente
 - Aceptable
 - Dirigido
 - Cumplido

De acuerdo a los requerimientos se establece el alcance y restricciones del sistema.

- Sistema de software. Sistema compuesto por software, hardware y datos. Sus estados son:
 - Arquitectura seleccionada.
 - Demostrable.
 - Usable.
 - Listo
 - En operación
 - Retirado

El sistema de software es el medio por el cual se logra cumplir con la oportunidad que origina su inicio cubriendo todos los requerimientos solicitados.

- Trabajo. Es el esfuerzo físico o mental para lograr un resultado, en este caso, son las actividades que el equipo realiza para obtener el sistema de software. Los estados son:

- Admitido
- Preparado
- Iniciado
- Controlado
- Concluido
- Cerrado

El trabajo deriva en actualizaciones y cambios en el sistema de software y en la manera de dirigir la oportunidad.

- Equipo. El conjunto de personas encargadas de realizar las actividades necesarias para cumplir con los objetivos del sistema de software. Los estados son:

- Motivados
- Formado
- Colaborando
- Ejecutando
- En espera

El equipo es el encargado de generar el sistema de software, planeando y realizando el trabajo requerido mediante una específica forma de trabajo.

- Forma de trabajo. Conjunto de técnicas y herramientas utilizados para guiar a un equipo con la finalidad de cumplir un objetivo. Los estados son:

- Principios establecidos
- Fundamento establecido
- Usándolo
- Establecido
- Trabajando bien
- Retirado

La forma de trabajo guía el trabajo del equipo.

ii. Espacios de actividad

El kernel también provee un conjunto de elementos que conforman un espacio de acuerdo a las áreas: cliente, solución y esfuerzo; compuesto por las actividades basadas en la visión de la ingeniería de software y complementado de esta forma los ALPHAs (Ver Figura 1.4).

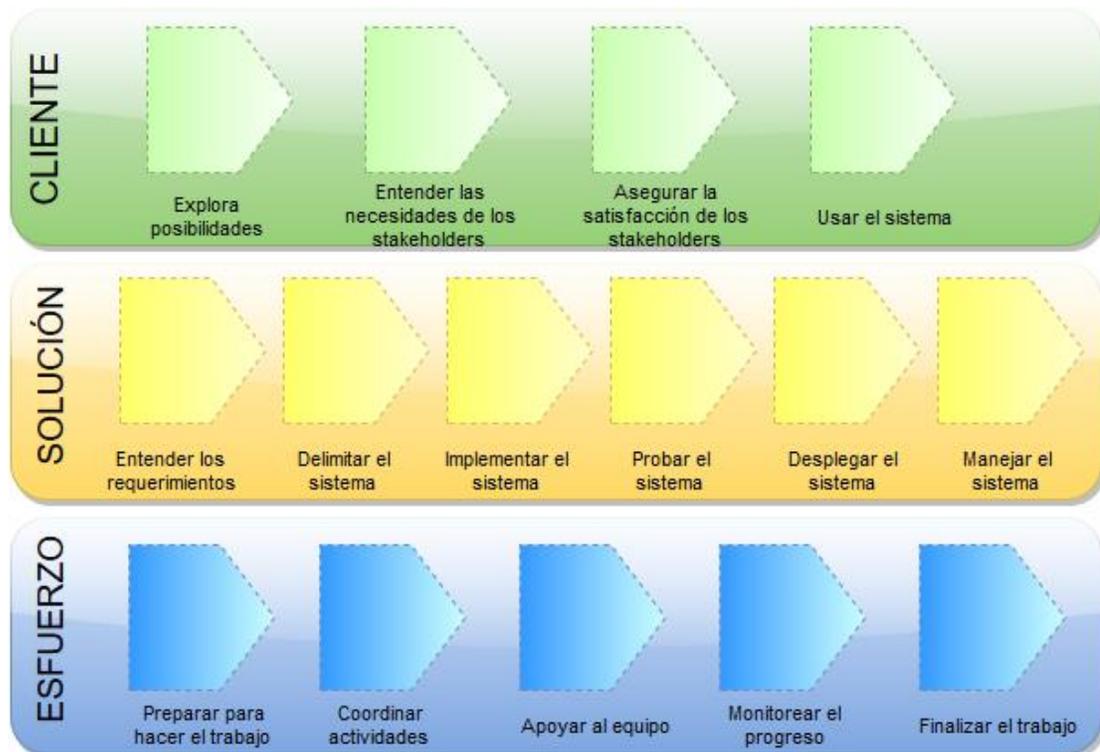


Figura 1.4. The kernel Activity Spaces [imagen] (2014)

Fuente: Adaptado de Kernel and Language for Software Engineering Methods, Essence v1.0

En el área de *cliente* se encuentran:

- Explorar posibilidades
- Comprender las necesidades de los stakeholders
- Asegurar la satisfacción de los stakeholders
- Usar el sistema.

En *solución* se encuentran:

- Entender los requerimientos
- Formar el sistema

- Implementar el sistema
- Probar el sistema
- Integrar el sistema
- Hacer funcionar el sistema

En *esfuerzo* se encuentran:

- Preparar el trabajo que se debe hacer
- Coordinar actividades
- Apoyar al equipo
- Monitorear el progreso
- Finalizar el trabajo

iii. Competencias

Para complementar el espacio de actividades y ALPHAs, el kernel también provee un conjunto de habilidades que son necesarias para realizar las actividades de ingeniería de software (Figura 1.5).



Figura 1.5. The kernel Competencies [imagen] (2014)

Fuente: Adaptado de Kernel and Language for Software Engineering Methods, Essence v1.0

En el área de *cliente* se encuentra:

- Representación de los stakeholders. Este término se refiere a la habilidad de comunicar, obtener y gestionar las necesidades de los involucrados así como representar su punto de vista.

En la *solución* se encuentran:

- Análisis. Integra la habilidad para comprender la oportunidad y las necesidades de los involucrados para generar requerimientos consistentes y entendibles.
- Desarrollo. Integra la habilidad para diseñar y desarrollar sistemas de software de acuerdo a estándares y normas definidas por el equipo.
- Pruebas. Integra la habilidad para hacer pruebas al sistema, verificar si es usable y si cumple o no con los requerimientos.

En el *esfuerzo* se encuentran:

- Liderazgo. Habilidad que comprende entre otras cosas la capacidad para guiar y motivar al equipo para cumplir exitosamente con los objetivos de trabajo.
- Gestión. Habilidad para coordinar, planear y monitorear el trabajo que realiza el equipo.

1.2.3. Lenguaje

Mediante un conjunto de reglas y una sintaxis establecida se definen las prácticas, métodos y la esencia del kernel, además de contener características que aportan un uso dinámico de estos.

La especificación de este lenguaje está compuesto por tres elementos:

1. Metamodelo. Expresa la sintaxis abstracta y restricciones en las relaciones entre los elementos.
2. Lenguaje formal. Permite expresar instancias o elementos del kernel basados en el metamodelo.
3. Lenguaje natural.

1.3. PMBOK

En la actualidad el término “proyecto” es muy utilizado en la vida diaria en cualquier área, ya sea desde un enfoque profesional o personal y con una complejidad muy variada. Esto da como resultado que a través del tiempo se requiera el uso de herramientas y técnicas especializadas para administrar los elementos, factores y recursos que componen a un proyecto para lograr exitosamente los objetivos establecidos.

Durante el paso de los años la administración de proyectos comenzó a tener un gran auge, es entonces que en 1969 surge el Project Management Institute (PMI), una asociación sin fines de lucro que tiene como misión reunir a profesionales interesados en el área.

Un proyecto es la agrupación de actividades temporales con la finalidad de generar un producto o resultado único, mientras que la administración de un proyecto es la aplicación de herramientas y técnicas en actividades de planificación, monitoreo y control de recursos y tiempo.

El PMI publica en el año de 1996 la primera edición del estándar que engloba un conjunto de buenas prácticas y base de conocimiento para la gestión de un proyecto, y que es denominado PMBOK, acrónimo de “A Guide to the Project Management Body of Knowledge”, subsecuentemente se publica en el 2000 la 2da edición, en el 2004 la 3era edición, en el 2009 la 4ta y actualmente se encuentra en vigencia la 5ta edición publicada en 2013.

El PMBOK puede ser utilizado en una amplia variedad de proyectos, sin importar su tamaño y complejidad. La última edición del PMBOK tiene entre sus aspectos más sobresalientes la creación de la nueva área de conocimiento sobre la gestión de los stakeholders. El objetivo de esta guía es ofrecer a los administradores de proyecto fundamentos conocidos como “buenas prácticas”, un vocabulario y conceptos comunes al tema para su desarrollo y aplicación, aumentando de esta manera la probabilidad de éxito del proyecto.

De acuerdo al Standish Group International¹⁰, se observa que los principales factores de éxito en un proyecto, es el involucramiento de los usuarios, apoyo en la administración ejecutiva y requerimientos claros (Tabla 1.1).

¹⁰ Standish Group International es una firma de asesoría independiente de investigación en TI.
<http://www.standishgroup.com/>

Tabla 1.1. Factores de éxito en un proyecto

Factores de éxito de un proyecto	%
1. El usuario está involucrado.	15.9
2. Apoyo en la administración ejecutiva	13.9
3. Requerimientos claros	13.0
4. Adecuada planeación	9.6
5. Expectativas realistas	8.2
6. Hitos más pequeños	7.7
7. Equipo competente	7.2
8. Patrocinio	5.3
9. Visión y objetivos claros	2.9
10. Arduo trabajo, equipo enfocado	2.4
11. Otros	13.4

El PMBOK 5ta edición está conformado por 47 procesos, integrados en 10 áreas de conocimiento y 5 grupos de procesos.

Un proceso es un conjunto de actividades relacionadas y sucesivas que a través de elementos de entrada generan un resultado.

Las categorías de los grupos de procesos de la administración de proyectos son:

- Inicio. Conjunto de procesos utilizados para definir y establecer los acuerdos iniciales de un proyecto.
- Planificación. Conjunto de procesos utilizados para establecer el alcance, los objetivos y definir la manera en que se llevarán a cabo las actividades que necesarias para lograr los objetivos del proyecto.
- Ejecución. Conjunto de procesos que se realizan para cumplir con los requerimientos del proyecto.
- Monitoreo y control. Conjunto de proceso que se realizan para revisar el progreso y desempeño del proyecto, y en dado caso identificar las áreas del plan que requieren cambios e iniciar con los mismos.

- Cierre. Conjunto de procesos que se realizan para dar por finalizadas las actividades y cerrar formalmente el proyecto.

Las áreas de conocimiento son:

- Gestión de la integración del proyecto. Incluye los procesos y actividades para unificar, consolidar y comunicar las diversas acciones para que el proyecto se complete adecuadamente y se maneje de manera controlada.
- Gestión del alcance del proyecto. Incluye los procesos que definen y controlan lo que debe incluir además de limitar el trabajo que se requiere para completar el proyecto.
- Gestión del tiempo del proyecto. Incluye los procesos relacionados con planificar, estimar y controlar el plazo que se requiere para concluir el proyecto.
- Gestión de los costos del proyecto. Incluye los procesos requeridos para planificar, estimar y controlar los costos para concluir el proyecto dentro del presupuesto definido.
- Gestión de la calidad del proyecto. Incluye los procesos requeridos para asegurar cabalmente el cumplimiento de los requisitos.
- Gestión de los recursos humanos del proyecto. Incluye los procesos que organizan y conducen a las personas que tienen asignados roles y responsabilidades en el proyecto.
- Gestión de las comunicaciones del proyecto. Incluye los procesos que permiten asegurar que la información que surge durante el proyecto sea comunicada efectivamente.
- Gestión de los riesgos del proyecto. Incluye los procesos para requeridos para identificar, analizar y controlar los riesgos existentes en el proyecto.
- Gestión de las adquisiciones del proyecto. Incluye los procesos para adquirir o comprar productos o servicios que se requiere obtener fuera del equipo.
- Gestión de los interesados del proyecto. Incluye los procesos para identificar y a las personas que afectan o son afectadas por el proyecto, así como conocer sus expectativas e impacto.

1.4. Otros estándares relacionados

El desarrollo de software comprende una amplia cantidad de conceptos y enfoques que sustentan este proceso, el cual depende de la complejidad y los objetivos específicos que se requieren cumplir. Para esto han surgido diversos estándares, modelos y metodologías que guían el proceso de desarrollo en sus diferentes enfoques, entre los que se encuentran:

- ISO/IEC 12207 Systems Software Engineering – Software Life Cycle Processes [9]. Es un estándar que contiene un conjunto de procesos del ciclo de vida, de las actividades y tareas para los sistemas de software.

Los procesos del ciclo de vida del sistema son:

- Procesos que involucran acuerdos
- Procesos referentes a la organización
- Procesos del proyecto
- Procesos técnicos

Los procesos específicos del software son:

- Procesos de implementación de software
- Procesos de soporte de software
- Procesos de reutilización de software

- ISO/IEC 15504 Software Process Improvement Capability Determination (SPICE). Modelo para la evaluación y mejora de la capacidad y madurez de los procesos de desarrollo de software.

Los niveles de madurez:

- 0: Inmadura. No existe una implementación efectiva de procesos.
- 1: Básica. Se realizan y logran los objetivos de los procesos.
- 2: Gestionada. Se gestionan los procesos y los productos resultantes.
- 3: Establecida. Se utilizan procesos basados en estándares.
- 4: Predecible. Se gestionan cuantitativamente los procesos.
- 5: Optimizada. Mejora continua de los procesos.

- Metodologías ágiles: Scrum, XP, Crystal Clear, etc
- SWEBOK. Guide to the Software Engineering Body of Knowledge.

Capítulo 2

Metodología

En todo proyecto, cualquiera que sea su naturaleza, se requiere el uso de metodologías para ejecutar su conjunto de procesos, métodos y técnicas que permitirán obtener un producto.

El presente capítulo muestra los factores tomados en cuenta para el desarrollo del presente trabajo y los pasos seguidos para construir las ALPHAs propuestas.

2.1. Guía genérica para construir un ALPHA

Un ALPHA¹¹, descrito en el capítulo anterior, es un elemento esencial que debe vigilarse para controlar el progreso de un proyecto.

Dado que estos elementos son fundamentales y proveen una visión general del proyecto, es importante su adecuada identificación y posterior construcción, para así integrar un conjunto completo que proporcione a los profesionales los pilares para llevar una adecuada gestión y control de sus proyectos.

Un proyecto, de acuerdo al área a la que pertenezca, está integrado por diversos elementos, fases, procesos y atributos específicos, debido a esto, la identificación de ALPHAs conlleva un proceso principalmente de análisis y comprensión de todo este conjunto, entre los que resaltan:

- Análisis de las fases que componen el proceso de desarrollo.
- Análisis de las áreas que integran las componentes de un proyecto.

¹¹ Abstract-Level Progress and Health Attribute

- Observación de la ejecución de proyectos del área interesada y análisis del entorno en ambientes diversos.

Una vez identificados estos elementos se recomienda un análisis exhaustivo de cada ALPHA, las cuales a su vez comprenden un conjunto de estados que establecen condiciones que deben cumplirse para controlar el avance del ALPHA, por lo cual deben ser definidos claramente. Para lograr lo anterior es importante lo siguiente:

- Identificado un ALPHA, se debe analizar el entorno de este elemento de acuerdo al contexto en que se desarrolla.
- Derivado del análisis anterior, se establecen los estados que conformarán a un ALPHA. Estos estados forman un flujo consecutivo dependiente de su estado previo.
- Cada estado de un ALPHA debe tener un conjunto de condiciones que determinan el cumplimiento del estado.

2.2. Selección de fuentes de información y cuerpos de conocimiento

Parte fundamental en el desarrollo de este trabajo es la identificación de las fuentes de información y cuerpos de conocimiento que constituirán la base para definir y construir cada ALPHA.

Una fuente de información proporciona datos y conocimiento fidedigno acerca de temas particulares, entre otras fuentes, se encuentran [13]:

- Enciclopedias
- Biografías
- Libros
- Revistas académicas
- Periódicos

Existen principalmente dos tipos de fuentes principales [14]:

- **Fuentes primarias:** Contienen información original y reciente, sin reinterpretación por parte de alguien más. En este grupo, se incluyen: diarios, entrevistas, fotografías, cartas.
- **Fuentes secundarias:** Contienen información extraída de fuentes primarias que ha sido sintetizada y reorganizada. Se incluyen: diccionarios, comentarios, libros de texto.

Un cuerpo de conocimiento, en inglés BoK (Body of Knowledge), es un tipo de fuente de información que integra un conjunto completo de elementos que proveen conceptos específicos en áreas de conocimiento particulares. Estos son realizados por profesionales que aportan su conocimiento y experiencia en la construcción de dichas fuentes.

Este tipo de fuentes de información, al ser conocimiento recopilado en el transcurso de los años y analizados posteriormente para conformar los cimientos de un cuerpo de conocimiento, se consideran una fuente secundaria.

Los cuerpos de conocimiento, específicamente en el ámbito de las TI forman una sólida base de conceptos, herramientas, competencias y buenas prácticas que guían la realización de diversos esfuerzos en el amplio mundo tecnológico, algunos ejemplos son:

- BABOK – Business Analysis Body of Knowledge¹².
- SWEBOK – Software Engineering Body of Knowledge [12].
- SEBOK – Systems Engineering Body of Knowledge¹³.
- EITBOK – Enterprise Information Technology Body of Knowledge¹⁴.

Seleccionar fuentes de información involucra la búsqueda e identificación de recursos que provean el conocimiento necesario para encaminar el desarrollo del presente trabajo. Por lo cual, los criterios utilizados en este proceso son: la relevancia que representan en el tema, el renombre que tienen en la comunidad perteneciente y su nivel de especialización. Las fuentes elegidas son:

- **ESSENCE.** Kernel and Language for Software Engineering Methods.

Este documento creado por el OMG¹⁵, publicado en Noviembre de 2014 su versión 1.0, provee de una base común de conceptos y sus descripciones que dan la pauta para la definición de prácticas de ingeniería de software mediante elementos esenciales prevalecientes en cualquier desarrollo de software.

ESSENCE tiene tres aristas primordiales en las que se centra:

- El kernel
- El lenguaje
- Prácticas

Este trabajo se basa primordialmente en el **kernel**, el cual incluye los elementos esenciales denominados ALPHAs que a través de sus estados y listas de verificación ofrecen a los practicantes un medio para controlar el progreso y salud de un desarrollo de software. Los ALPHAs definidos en ESSENCE son:

- Oportunidad
- Stakeholders
- Requerimientos
- Sistema de software
- Trabajo
- Equipo
- Forma de trabajo

¹² Creado por el International Institute of Business Analysis (IIBA) <https://www.iiba.org/babok-guide.aspx>

¹³ Creado por el Body of Knowledge and Curriculum to Advance Systems Engineering (BKCASE), <http://www.bkcase.org/sebok/>

¹⁴ Creado por la IEEE Computer Society, <http://www.computer.org/web/education/eitbok>

¹⁵ Object Management Group, <http://www.omg.org/>

De ESSENCE se aprovechará el enfoque basado en ALPHAs y su estructura para proponer un conjunto de ALPHAs genéricas que permitan la administración de proyectos de software.

- **PMBok.** Project Management Body of Knowledge [11].

Describe los conceptos de los procesos y las mejores prácticas de la administración de proyectos de cualquier índole. Es desarrollado por profesionales afines al tema y actualmente la última edición (5ta) proporciona el marco teórico de los elementos y sus relaciones en un proyecto divididos en 10 áreas de conocimiento.

Está conformado principalmente por dos secciones iniciales que describen conceptos comunes de un proyecto y de las organizaciones. En una tercera sección se abordan los grupos de procesos que integran un proyecto. A partir de la cuarta sección se describe cada área de conocimiento y los procesos que lo conforman.

Las áreas de conocimiento son:

- Gestión de la integración
- Gestión del alcance
- Gestión del tiempo
- Gestión de los costos
- Gestión de la calidad
- Gestión de los recursos humanos
- Gestión de las comunicaciones
- Gestión de los riesgos
- Gestión de las adquisiciones
- Gestión de los stakeholders

De este conjunto, los elementos que conformarán las fuentes de información para el presente trabajo son los procesos de las 10 áreas de conocimiento que integran al PMBoK. Especialmente la *Descripción General* de cada área, que enlista entradas, salidas, herramientas y técnicas de cada proceso ofreciendo así una visión general de dichos procesos, por ejemplo, la Figura 2.1 del área ***Alcance del Proyecto***.

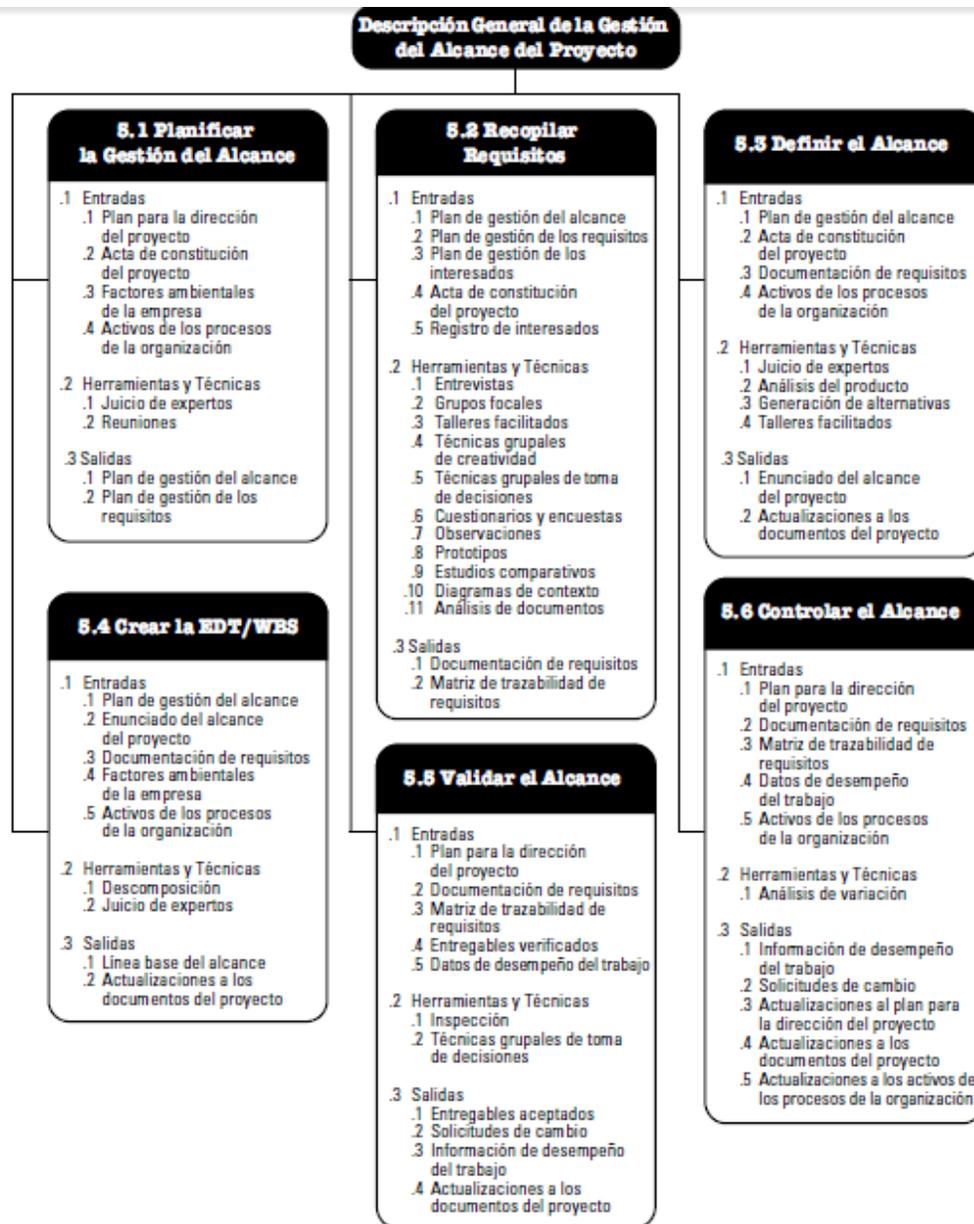


Figura 2.1. Descripción General de la Gestión del Alcance del Proyecto [imagen] (2013)

Fuente: PMBOK Guide, Fifth Edition

- **SWEBoK.** Software Engineering Body of Knowledge [12].

Creado por la IEEE¹⁶ Computer Society, congrega el conocimiento de la ingeniería de software describiendo los términos comúnmente utilizados en el área, así como prácticas y teorías que constituyen una referencia en la práctica de la profesión.

¹⁶ Institute of Electrical and Electronics Engineers, <https://www.ieee.org>

SWEBoK V3.0 publicado en 2014, comprende 15 áreas de conocimiento que abarcan los temas esenciales de la ingeniería de software además de fundamentos de computación, matemáticas e ingeniería:

- Requerimientos de software
- Diseño de software
- Construcción de software
- Pruebas de software
- Mantenimiento de software
- Gestión de la configuración de software
- Gestión de la ingeniería de software
- Proceso de ingeniería de software
- Métodos y modelos de la ingeniería de software
- Calidad del software
- Práctica profesional de la ingeniería de software
- Economía de la ingeniería de software
- Fundamentos de computación
- Fundamentos matemáticos
- Fundamentos de ingeniería

De los elementos previamente mencionados, se utilizarán los siguientes:

- **Requerimientos de software.** Proporciona conceptos correspondientes a la obtención, clasificación y análisis de los requisitos indispensables para el desarrollo del software.
 - **Diseño de software.** Proporciona conceptos correspondientes a la abstracción, estructuración y modelado de los componentes del software.
 - **Construcción de software.** Proporciona conceptos correspondientes a los modelos del ciclo de vida del software, planificación, diseño, así como la importancia del análisis para la elección de los lenguajes de programación utilizados para la construcción del software.
 - **Pruebas de software.** Proporciona conceptos acerca del proceso y técnicas de las pruebas de software.
 - **Gestión de la configuración del software.** En ésta área específicamente se utiliza el desglose de los temas que proporciona una visión global acerca de esta área de conocimiento.
 - **Calidad del software.** Proporciona conceptos acerca de los procesos, herramientas y técnicas de gestión de la calidad.
- **ISO/IEC 12207.** Systems and software engineering – Software life cycle processes.

Estándar creado por la IEEE¹⁷, que establece una terminología y conceptos de los procesos relacionados al ciclo de vida del software. Proveen así un contexto de los productos y servicios de las distintas categorías:

- Procesos que involucran acuerdos.
- Procesos referentes a la organización.

¹⁷ Institute of Electrical and Electronics Engineers, <https://www.ieee.org/>

- Procesos del proyecto.
- Procesos técnicos.
- Procesos de implementación del software.
- Procesos para el soporte del software.
- Procesos para la reutilización del software.

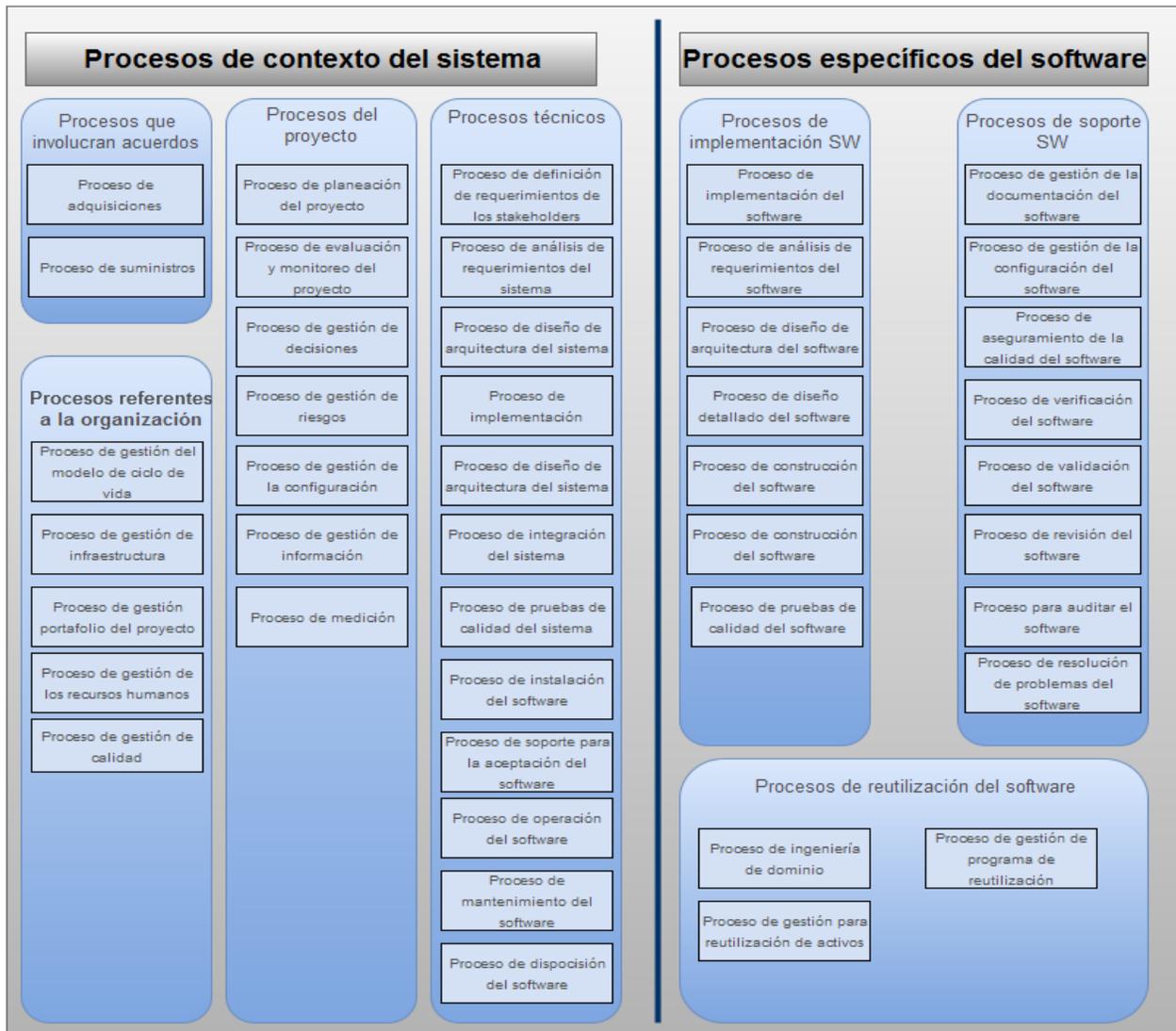


Figura 2.2. Grupos de procesos del ciclo de vida [imagen] (2008)

Fuente: Adaptado de ISO/IEC 12207

De acuerdo a la Figura 2.2 que muestra los grupos de procesos y categorías del ciclo de vida, los utilizados para la realización de este trabajo serán:

- Proceso de adquisiciones.
- Proceso de gestión de recursos humanos.
- Proceso de gestión de calidad.
- Proceso de gestión de riesgos.

- Proceso de análisis de requerimientos del sistema.
- Proceso de definición de requerimientos de los stakeholders.
- Proceso de integración del sistema.
- Proceso de integración del software.
- Proceso de verificación del software.
- Proceso de validación del software.
- Proceso de aseguramiento de la calidad del software.

2.3. Estructura básica de las ALPHAs

Un ALPHA, visto como un elemento esencial para controlar el seguimiento y progreso, debe abstraer completamente el entorno que lo conforma e integrarse por los aspectos básicos siguientes (Ver Figura 2.3):

- **Nombre:** Es importante que cada ALPHA tenga un nombre representativo.
- **Estados:** Representan las transiciones de un ALPHA durante el proceso de desarrollo, estos permiten el control del progreso.
- **Lista de verificación:** Conjunto de condiciones que deben ser cumplidas para completar el estado correspondiente.

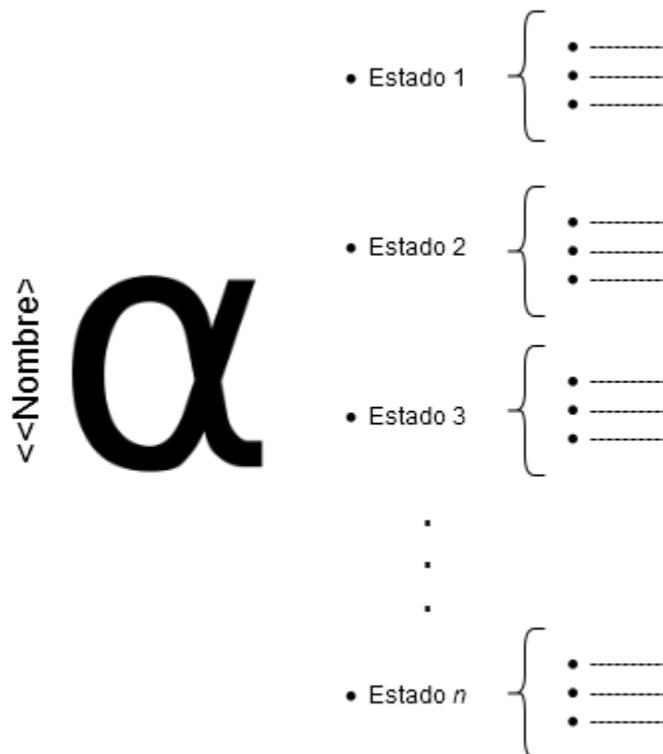


Figura 2.3. Estructura de un ALPHA

2.4. Verificación y validación

Cada ALPHA debe ser sometida al proceso de verificación y validación para comprobar que están acorde a lo que se está representando y, en caso de identificación de defectos, aplicar las correcciones necesarias con el fin de obtener un producto de calidad. Los procesos de verificación y validación abarcan una serie de actividades que deben ser llevadas a cabo para evaluar el cumplimiento de las especificaciones.

La verificación de un ALPHA se refiere al cumplimiento de los aspectos que integran el proceso de construcción así como de su estructura. Para esto será necesario realizar revisiones para monitorear tales elementos, descritos previamente en la sección 2.1 y 2.3.

En la validación de un ALPHA se comprobará que ésta cumpla con el objetivo primordial, que es, proporcionar los elementos esenciales que provean a los profesionales de TI las bases para monitorear el progreso y salud de los proyectos. Para realizar la validación se requiere la intervención de personal perteneciente al ámbito tecnológico para aportar su juicio experto acerca de las ALPHAs, y así determinar si cumplen sus expectativas en el uso de dichos elementos para el control de un proyecto.

La verificación y validación de ALPHAs consistirá en diversas actividades agrupadas en dos rubros:

- Estructural. Tiene como objetivo comprobar que se ha generado un ALPHA con la estructura definida en la sección 2.3., es decir, compuesto por un nombre, un conjunto de estados y sus respectivas listas de verificación. Las actividades relacionadas para corroborar dicha estructura son:
 - Inspección formal por parte de un revisor que identifique errores o inconsistencias en cada ALPHA.
 - Reporte y registro de la revisión del ALPHA con la descripción de los defectos encontrados y observaciones complementarias. El formato para el registro se observa en la Figura 2.4 (ver Anexo 1).

Nombre	Nombre del revisor	Fecha	dd/mm/aaaa
ALPHA	Nombre del ALPHA		
Defectos			
1.			
2.			
3.			
Observaciones			

Figura 2.4. Formato para el registro de defectos.

- Operacional. En este grupo de actividades el objetivo se centrará en la validación de cada ALPHA, primordialmente en su semántica, secuencia de sus estados y suficiencia de las listas de verificación. Las actividades que lo conforman son:
 - Presentación y uso de las ALPHAs a profesionales del área tecnológica que de acuerdo a su *expertise* emitan un juicio que permita comprobar si los ALPHAs poseen la capacidad suficiente para monitorear el progreso y salud de un proyecto de software. La información se recolectará a través de encuestas (ver Anexo 2), entre las preguntas que se realizarán, se encuentran:
 - ¿El nombre del ALPHA es representativo?
 - ¿Los estados están ordenados correctamente?
 - ¿Los nombres de los estados son representativos?
 - ¿Agregarías/quitarías algún estado?
 - ¿Las listas de verificación son suficientes?
 - ¿Agregarías/quitarías algún elemento de las listas de verificación?

Los resultados de la verificación y validación de las ALPHAs servirán para generar una versión revisada de éstas, mismas que conformarán el resultado principal de este trabajo de tesis. Los detalles de la verificación y validación se presentan en el capítulo 4.

Capítulo 3

Construcción de ALPHAs

Este capítulo mostrará las ALPHAs generadas del análisis y clasificación de los procesos del PMBoK, siendo este el marco de referencia más utilizado para la administración de proyectos. Aunque el trabajo se basa principalmente en este cuerpo de conocimiento se fundamenta también en otros estándares relacionados a proyectos de software.

Las ALPHAs se cimientan en los 47 procesos que conforman al PMBoK en su 5ta edición, agrupados en 5 categorías y 10 áreas de conocimiento (Tabla 3.1), respecto a estas se construye cada ALPHA:

- ALPHA Integración
- ALPHA Alcance
- ALPHA Tiempo
- ALPHA Costos
- ALPHA Calidad
- ALPHA Recursos Humanos
- ALPHA Comunicaciones
- ALPHA Riesgos
- ALPHA Adquisiciones
- ALPHA Stakeholders

Tabla 3.1. Grupos de procesos y áreas de conocimiento.

Fuente: PMBoK Quinta edición, 2013

Áreas de conocimiento	Grupo de procesos de inicio	Grupo de proceso de planificación	Grupos de proceso de ejecución	Grupos de proceso monitoreo y control	Grupos de proceso de cierre
4.Gestión de integración del proyecto	4.1.Desarrollar el acta de constitución del proyecto	4.2.Desarrollar el plan para la dirección del proyecto	4.3.Dirigir y gestionar el trabajo	4.4.Monitorear y controlar el trabajo del proyecto. 4.5.Realizar el control integral de cambios.	4.5.Cierre del proyecto.
5.Gestión del alcance del proyecto		5.1.Planificar la gestión del alcance. 5.2.Recopilar requisitos. 5.3. Definir el alcance 5.4. Crear WBS		5.5.Validar el alcance. 5.6. Controlar el alcance.	
6.Gestión del tiempo del proyecto		6.1.Planificar la gestión del cronograma. 6.2.Definir las actividades. 6.3.Secuenciar las actividades. 6.4.Estimar los recursos de las actividades. 6.5.Estimar la duración de las actividades. 6.6.Desarrollar el cronograma.		6.7.Controlar el cronograma.	
7.Gestión de los costos del proyecto		7.1.Planificar la gestión de los costos. 7.2.Estimar los costos. 7.3.Determinar el presupuesto.		7.4.Controlar los costos.	
8.Gestión de la calidad del proyecto		8.1.Planificar la gestión de la calidad.	8.2.Realizar el aseguramiento de la calidad.	8.3.Controlar la calidad.	

Tabla 3.1. Grupos de procesos y áreas de conocimiento.

Fuente: PMBoK Quinta edición, 2013

Áreas de conocimiento	Grupo de procesos de inicio	Grupo de proceso de planificación	Grupos de proceso de ejecución	Grupos de proceso de monitoreo y control	Grupos de proceso de cierre
9.Gestión de los recursos humanos del proyecto		9.1.Planificar la gestión de los recursos humanos.	9.2.Adquirir el equipo del proyecto. 9.3.Desarrollar el equipo del proyecto. 9.4.Dirigir el equipo del proyecto.		
10.Gestión de las comunicaciones del proyecto		10.1.Planificar la gestión de las comunicaciones.	10.2.Gestionar las comunicaciones.	10.3.Controlar las comunicaciones.	
11.Gestión de los riesgos del proyecto		11.1.Planificar la gestión de los riesgos. 11.2.Gestionar los riesgos. 11.3.Realizar el análisis cualitativo de los riesgos. 11.4.Realizar el análisis cuantitativo de los riesgos. 11.5.Planificar la respuesta a los riesgos.		11.6.Controlar los riesgos.	
12.Gestión de las adquisiciones del proyecto		12.1.Planificar la gestión de las adquisiciones.	12.2.Efectuar las adquisiciones.	12.3.Controlar las adquisiciones.	12.4.Cerrar las adquisiciones.
13.Gestión de los stakeholders del proyecto	13.1.Identificar a los stakeholders.	13.2.Planificar la gestión de los stakeholders.	13.3.Gestionar la participación de los stakeholders.	13.4. Controlar la participación de los stakeholders.	

En las subsecuentes secciones se presentará la parte central de este trabajo que mostrará las diez ALPHAS y su importancia en el desarrollo de proyectos. De acuerdo al capítulo previo, cada ALPHA tendrá en primera instancia la representación gráfica de sus estados y su transición, posteriormente la lista de verificación de cada estado.

3.1. ALPHA Integración

En ésta área de conocimiento se encuentran los procesos que coadyuvan al logro de los objetivos de un proyecto mediante la consolidación y unificación de todas las actividades dentro de todos los grupos de procesos.

Gestionar la integración de un proyecto tiene por objetivo completar los elementos que lo conforman desde un alto nivel de trabajo, entre otros procesos, se encuentran el de establecer formalmente el inicio, planificar, monitorear, controlar y cerrar las actividades del proyecto. Siendo estos, la base para encaminar de manera efectiva el proyecto, el producto y el proceso. La Figura 3.1 muestra los estados que componen el ALPHA Integración.

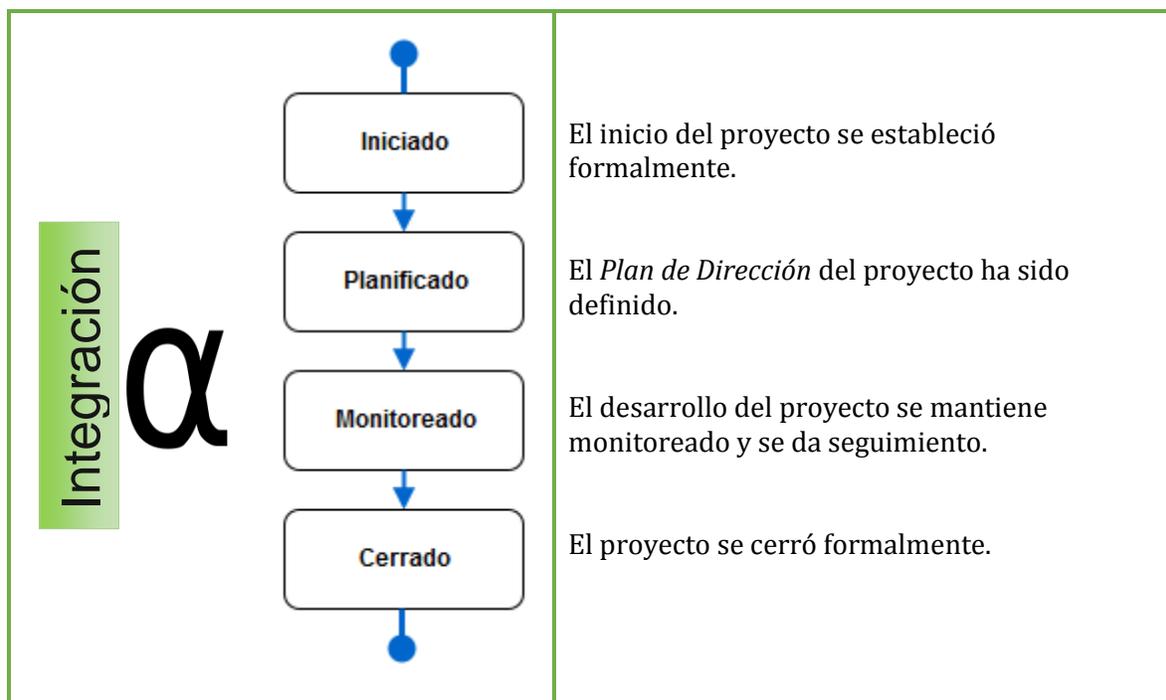


Figura 3.1. ALPHA Integración

La Tabla 3.2 muestra la lista de verificación para el ALPHA Integración.

Tabla 3.2. Lista de verificación para el ALPHA Integración

Estado	Lista de verificación
Iniciado	<ul style="list-style-type: none"> <input type="checkbox"/> Se han identificado que los recursos para la administración y ejecución del proyecto están disponibles y son viables. <input type="checkbox"/> El <i>Acta Constitutiva</i> del proyecto ha sido definida. <input type="checkbox"/> Los stakeholders han sido identificados. <input type="checkbox"/> Las partes involucradas han aprobado el <i>Acta Constitutiva</i>. <input type="checkbox"/> Los stakeholders conocen el alcance y objetivos generales del proyecto.
Planeado	<ul style="list-style-type: none"> <input type="checkbox"/> El <i>Plan de Dirección</i> del proyecto ha sido establecido. <input type="checkbox"/> Los stakeholders conocen y están de acuerdo con lo contenido en el <i>Plan del Dirección</i>. <input type="checkbox"/> Los planes subsidiarios han sido desarrollados. <input type="checkbox"/> El <i>Plan de Dirección</i> del proyecto ha sido aprobado.
Monitoreado	<ul style="list-style-type: none"> <input type="checkbox"/> El trabajo se ha gestionado de acuerdo al <i>Plan de Dirección</i>. <input type="checkbox"/> La ejecución del proyecto ha sido monitoreada. <input type="checkbox"/> Los cambios realizados al plan han sido aprobados y validados. <input type="checkbox"/> Los procesos que integran el proyecto han sido aplicados.
Cerrado	<ul style="list-style-type: none"> <input type="checkbox"/> El trabajo ha sido completado. <input type="checkbox"/> Los planes subsidiarios han sido finalizados. <input type="checkbox"/> Los contratos han sido cerrados. <input type="checkbox"/> Se hizo transferencia del producto final en caso de aplicar. <input type="checkbox"/> Se ha aprobado el producto final. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

Elementos tomados de otras fuentes de conocimiento:

ISO/IEC 12207

- De la sección **Procesos del proyecto** se tomaron como referencia los procesos de configuración, planeación, monitoreo y control, integración del sistema, los cuales proporcionan el objetivo, las salidas, actividades y tareas relacionadas a los correspondientes procesos.

SWEBoK

- Se tomaron como referencia los elementos de **Gestión de la configuración del software** y **Construcción del software** que aportan conceptos respecto a modelos del ciclo de vida del software y la gestión de la configuración.

3.2. ALPHA Alcance

La gestión del alcance de un proyecto abarca los procesos que contribuyen a definir y planificar las actividades requeridas para cumplir con los objetivos del proyecto. Cabe resaltar que para encaminar el alcance del proyecto es importante recopilar, clasificar, monitorear los requisitos tanto del software como del proyecto para delimitar el trabajo requerido.

Lograr los objetivos del proyecto implica entre otras cosas, cumplir con las características del software a generar y sus restricciones, por lo cual es esencial llevar a cabo las actividades necesarias para cubrir tales requerimientos. Los procesos que cubre la Gestión del Alcance son de gran relevancia para transformar los requerimientos de los stakeholders en actividades que faciliten y guíen el desarrollo, además de constituir también la base para planificar los costos, el tiempo y la calidad del proyecto. La Figura 3.2 muestra el ALPHA Alcance.

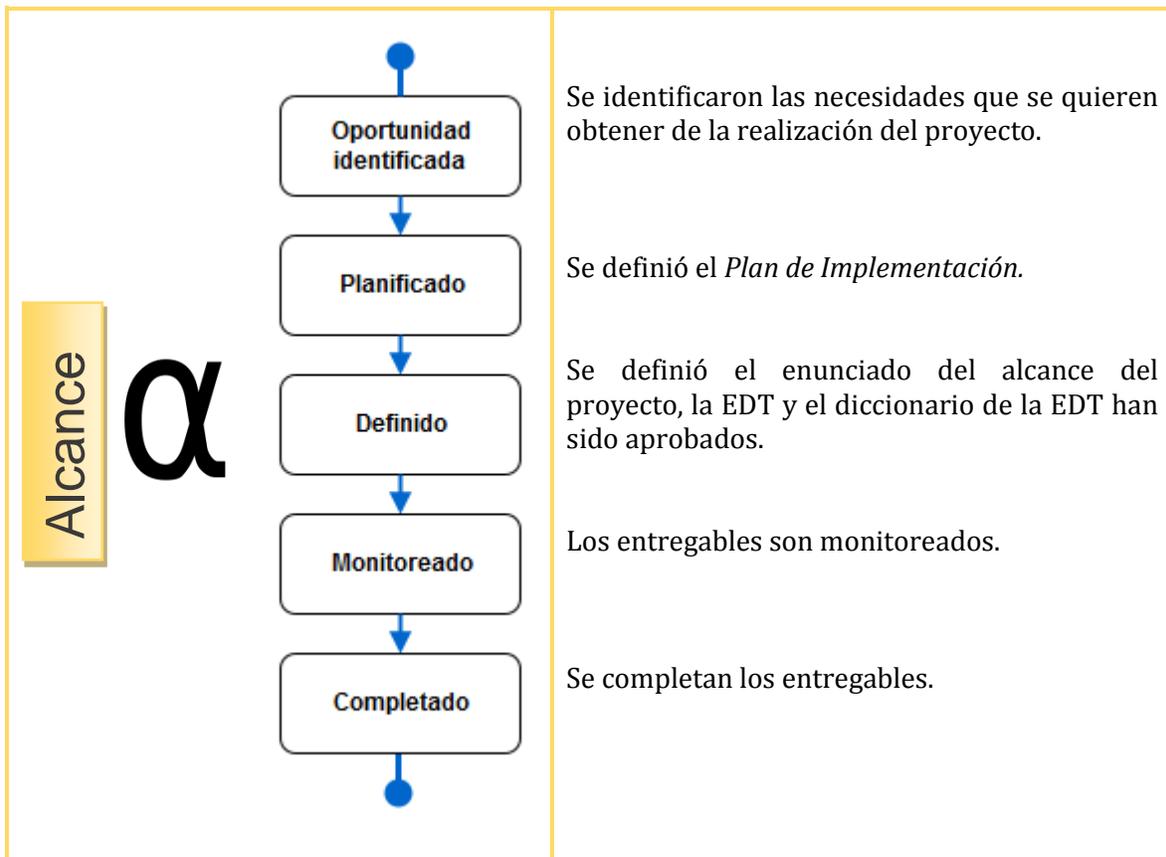


Figura 3.2. ALPHA Alcance

La Tabla 3.3 muestra la lista de verificación para el ALPHA Alcance.

Tabla 3.3. Lista de verificación del ALPHA Alcance

Estado	Lista de verificación
Oportunidad identificada	<ul style="list-style-type: none"> <input type="checkbox"/> La oportunidad derivada a partir de un problema es identificada. <input type="checkbox"/> La oportunidad para generar una solución al cliente ha sido identificada. <input type="checkbox"/> Los stakeholders tienen claro el problema y la oportunidad generada.
Planificado	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha establecido una estrategia para definir el alcance del proyecto. <input type="checkbox"/> Se ha analizado herramientas y técnicas para definir el alcance. <input type="checkbox"/> El <i>Plan de Implementación</i> es definido. <input type="checkbox"/> Los stakeholders están en común acuerdo de lo establecido en el <i>Plan de Implementación</i>.
Definido	<ul style="list-style-type: none"> <input type="checkbox"/> El cliente está en comunicación con el equipo para proveer de información que ayude a definir los requerimientos del proyecto. <input type="checkbox"/> El equipo de trabajo ha utilizado técnicas y herramientas para obtener los requerimientos del cliente. <input type="checkbox"/> Los requerimientos han sido recopilados y aceptados por los stakeholders y cliente. <input type="checkbox"/> El alcance del proyecto está acotado, claro y aceptado por los stakeholders. <input type="checkbox"/> Los stakeholders conocen detalladamente el proyecto y el producto a generar. <input type="checkbox"/> La <i>estructura de desglose de trabajo</i> (EDT) ha sido realizada y aprobada. <input type="checkbox"/> El <i>diccionario de EDT</i> se ha definido y aprobado.
Monitoreado	<ul style="list-style-type: none"> <input type="checkbox"/> Los componentes de la <i>EDT</i> son completados. <input type="checkbox"/> Los entregables son verificados. <input type="checkbox"/> Las actividades para completar el alcance del proyecto son monitoreadas. <input type="checkbox"/> El <i>Plan de Implementación</i> ha sido actualizado <input type="checkbox"/> Los <i>requerimientos funcionales y no funcionales</i> se han desarrollado. <input type="checkbox"/> Los documentos generados del proyecto se han actualizado.
Completado	<ul style="list-style-type: none"> <input type="checkbox"/> Los entregables se han validado. <input type="checkbox"/> Todos los entregables han sido aceptados y firmados por el cliente. <input type="checkbox"/> Todos los requerimientos han sido cubiertos por los entregables. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

Elementos tomados de otras fuentes de conocimiento:

ISO /IEC 12207

- De la sección **Proceso de análisis de requerimientos del sistema**, se toma como referencia sus actividades y tareas en los sistemas de software.

SWEBoK

- Se tomaron como referencia los elementos de **Requerimientos de software** y **Diseño de software**.

3.3. ALPHA Tiempo

La Gestión del Tiempo de un proyecto incluye los procesos para planificar, establecer y controlar el cronograma que permitan la finalización en el tiempo definido. Entre los procesos principales se encuentran estimar los recursos a cada actividad y desarrollar el cronograma.

Mediante esta ALPHA se pretende monitorear el progreso de los procesos relacionados al tiempo del proyecto. Esta área es pieza angular de la denominada triple restricción (alcance, tiempo, costo) que son parámetros que determinan en gran medida si un proyecto tiene éxito o no. La Figura 3.3 muestra el ALPHA de gestión del tiempo de un proyecto.

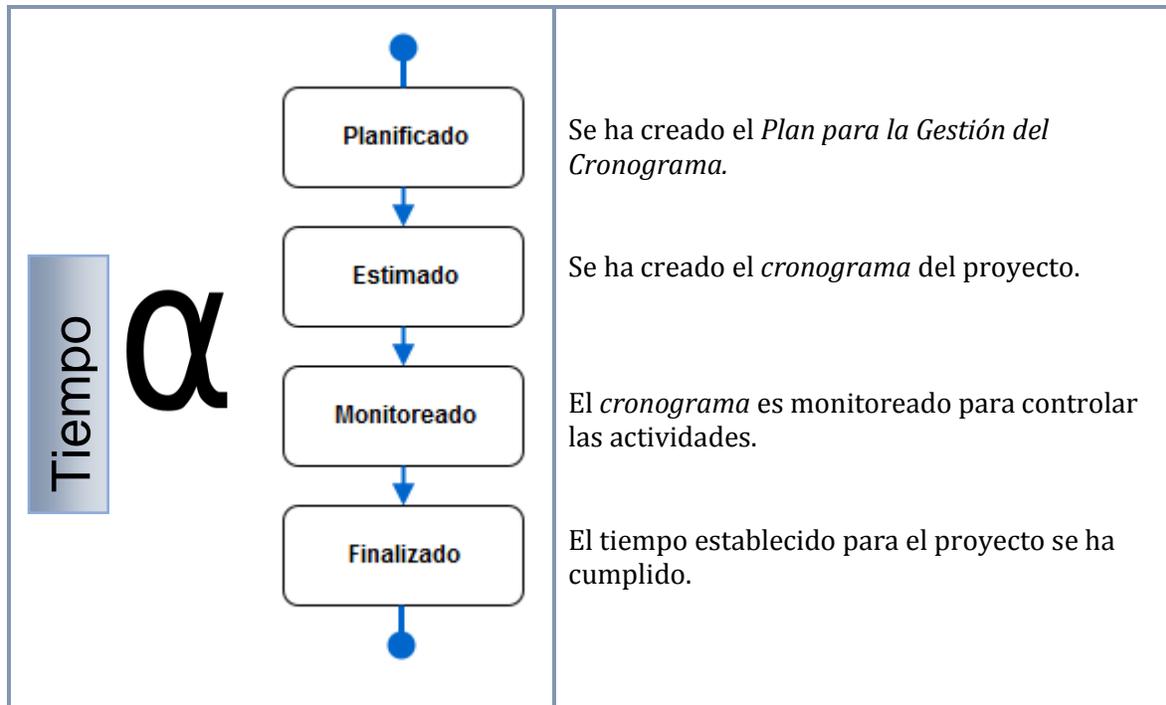


Figura 3.3. ALPHA Tiempo

La Tabla 3.4 muestra la lista de verificación para el ALPHA Tiempo.

Tabla 3.4. Lista de verificación del ALPHA Tiempo

Estado	Lista de verificación
Planificado	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha establecido una estrategia para gestionar el tiempo del proyecto. <input type="checkbox"/> El <i>Plan para la Gestión del Cronograma</i> es definido. <input type="checkbox"/> Los stakeholders están en común acuerdo con las fechas establecidas en el plan.
Estimado	<ul style="list-style-type: none"> <input type="checkbox"/> Las actividades requeridas para completar el proyecto han sido definidas. <input type="checkbox"/> Las actividades han sido secuenciadas. <input type="checkbox"/> Los recursos han sido asignados a cada actividad. <input type="checkbox"/> La duración ha sido estimada para cada actividad. <input type="checkbox"/> El <i>cronograma</i> ha sido desarrollado. <input type="checkbox"/> Los stakeholders conocen y han aceptado el cronograma. <input type="checkbox"/> El cronograma ha sido aprobado.
Monitoreado	<ul style="list-style-type: none"> <input type="checkbox"/> El <i>cronograma</i> es monitoreado y se da seguimiento <input type="checkbox"/> El equipo ha realizado sus actividades acorde al cronograma. <input type="checkbox"/> Se ha ajustado el <i>cronograma</i> por cambios en el proyecto. <input type="checkbox"/> Los cambios hechos al <i>cronograma</i> son validados. <input type="checkbox"/> Los cambios son documentados y comunicados.
Finalizado	<ul style="list-style-type: none"> <input type="checkbox"/> Las actividades correspondientes para completar el proyecto han finalizado. <input type="checkbox"/> El tiempo establecido en el cronograma ha finalizado. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

En ésta ALPHA sólo se consideró al PMBoK como fuente de información.

3.4. ALPHA Costos

La gestión de los costos en un proyecto incluye los procesos para planificar, estimar y controlar el presupuesto establecido.

El ALPHA de costos contribuye principalmente al monitoreo y control del presupuesto con el fin de completar las actividades dentro de lo establecido. Los procesos que contempla el área son importantes porque forman parte de la triple restricción de un proyecto, siendo fundamental en el logro de los objetivos y su éxito. En la Figura 3.4 se muestra el ALPHA Costos.

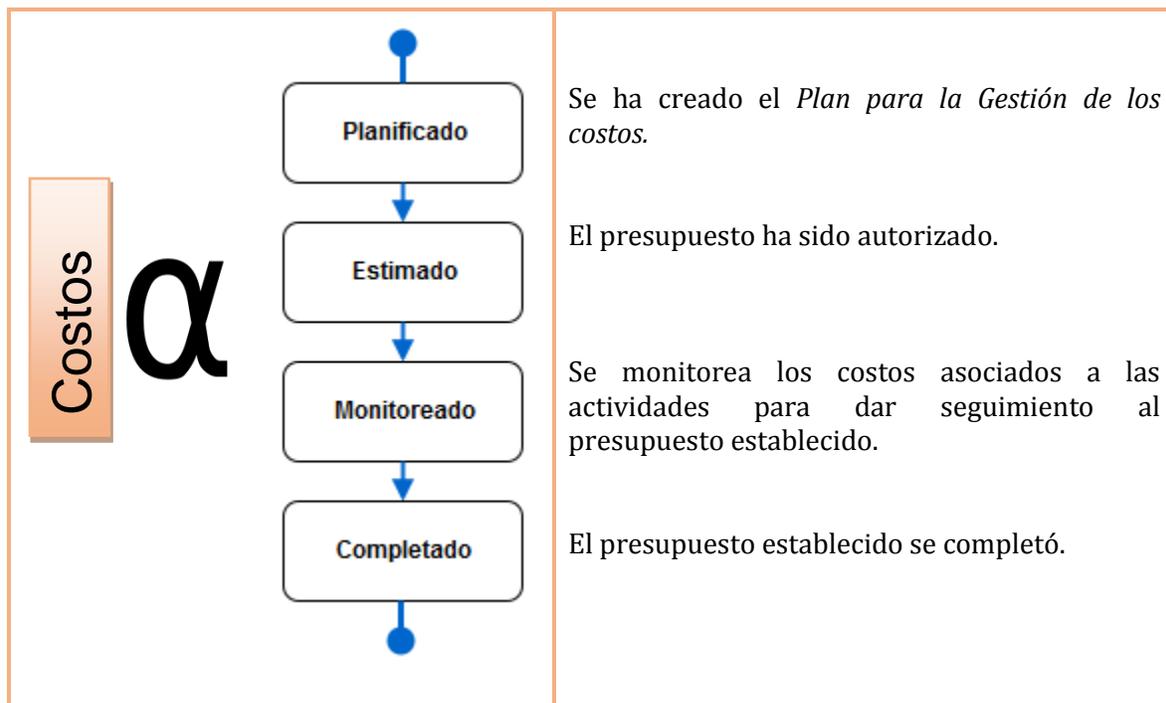


Figura 3.4. ALPHA Costos

La Tabla 3.5 muestra la lista de verificación para el ALPHA Costos.

Tabla 3.5. Lista de verificación del ALPHA Costos

Estado	Lista de verificación
Planificado	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha establecido una estrategia para definir los costos del proyecto. <input type="checkbox"/> El <i>Plan para la Gestión de los Costos</i> es definido. <input type="checkbox"/> Los stakeholders están en común acuerdo de lo establecido en el plan.
Estimado	<ul style="list-style-type: none"> <input type="checkbox"/> Los costos que se requieren para completar el proyecto han sido estimados. <input type="checkbox"/> Los costos han sido aprobados. <input type="checkbox"/> El presupuesto ha sido autorizado
Monitoreado	<ul style="list-style-type: none"> <input type="checkbox"/> Las actividades correspondientes al desarrollo del proyecto son controladas de acuerdo a los costos establecidos. <input type="checkbox"/> El presupuesto ejercido se registra. <input type="checkbox"/> Se registran cambios de costos al plan para actualizarlo. <input type="checkbox"/> Se controla el impacto de los cambios realizados.
Completado	<ul style="list-style-type: none"> <input type="checkbox"/> El equipo ha completado todas las actividades que integran el proyecto. <input type="checkbox"/> El costo del proyecto se utilizó conforme al presupuesto establecido. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

En ésta ALPHA sólo se consideró al PMBoK como fuente de información.

3.5. ALPHA Calidad

La calidad se refiere a *las funciones y características de un producto o servicio que cumplen y satisfacen las necesidades o requerimientos implícitos o explícitos del mismo*¹⁸.

La gestión de la calidad en un proyecto cubre los procesos necesarios para satisfacer los requerimientos del cliente, para validar y alcanzar los requisitos del proyecto y del producto, asegurando también el logro de los objetivos de calidad de la organización.

Gestionar los procesos relacionados a la calidad del proyecto es vital para entregar un producto que cumpla con las expectativas del cliente, y asegurar en todo momento que se cumplen con las normas y estándares establecidos. En la Figura 3.5 se muestra el ALPHA Calidad.

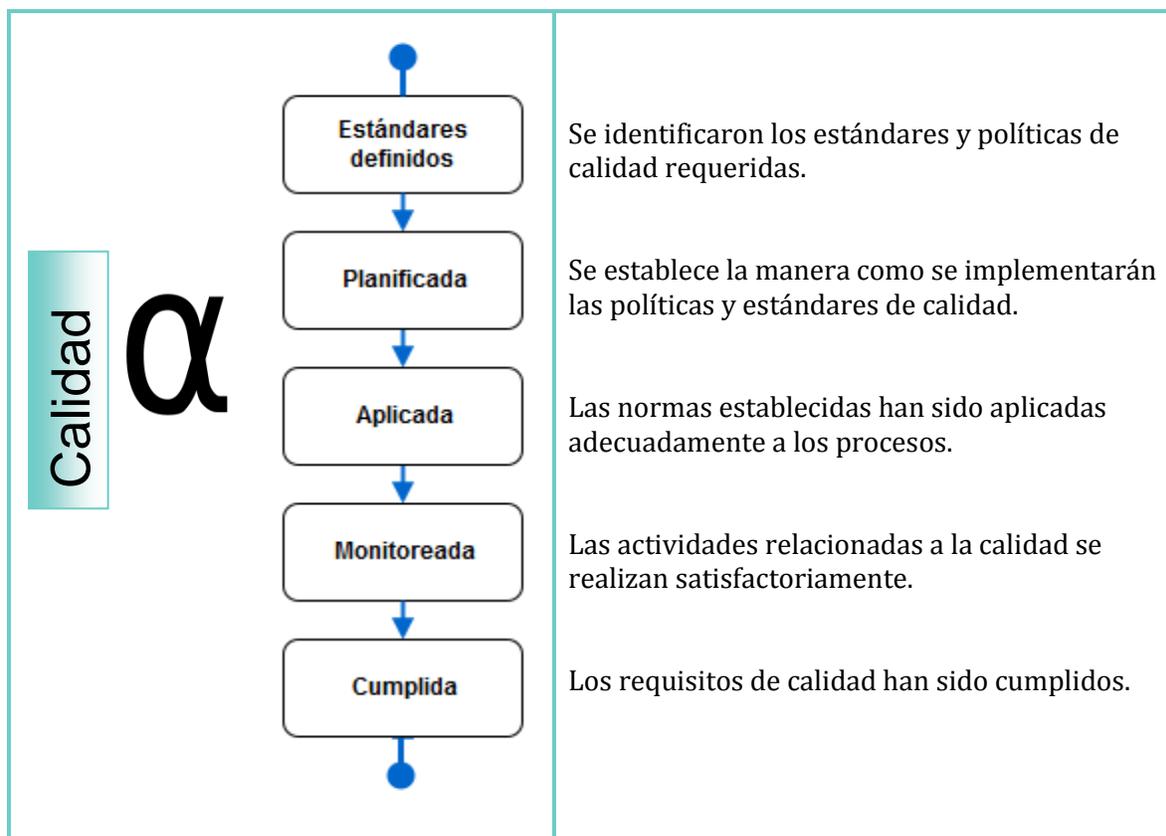


Figura 3.5. ALPHA Calidad

¹⁸ ISO 9001:2008. Norma de sistemas de gestión de la calidad.

La Tabla 3.6 muestra la lista de verificación para el ALPHA Calidad.

Tabla 3.6. Lista de verificación del ALPHA Calidad

Estado	Lista de verificación
Estándares definidos	<ul style="list-style-type: none"> <input type="checkbox"/> Se han determinado las políticas y estándares que definen los procesos de calidad. <input type="checkbox"/> Los stakeholders conocen y aceptan los estándares definidos. <input type="checkbox"/> Se han establecido metas y objetivos de calidad de la organización.
Planificada	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha establecido una estrategia para definir los procesos de calidad del proyecto. <input type="checkbox"/> Se han analizado las herramientas y técnicas para definir las actividades relacionadas al control y aseguramiento de la calidad. <input type="checkbox"/> El <i>Plan para la Gestión de la Calidad</i> es definido. <input type="checkbox"/> Los stakeholders están en común acuerdo de lo establecido en el plan.
Aplicada	<ul style="list-style-type: none"> <input type="checkbox"/> Se han definido responsables para la implementación de calidad. <input type="checkbox"/> Se aplican los estándares y normas definidos en el plan.
Monitoreada	<ul style="list-style-type: none"> <input type="checkbox"/> Se han revisado los requerimientos de calidad para vigilar su cumplimiento. <input type="checkbox"/> Se monitorean los procesos de calidad establecidos. <input type="checkbox"/> Se han controlado los cambios realizados para ajustar el plan. <input type="checkbox"/> Se contiene el impacto de los cambios realizados.
Cumplida	<ul style="list-style-type: none"> <input type="checkbox"/> Los requisitos de calidad se han cumplido según el plan. <input type="checkbox"/> El cliente ha aprobado los entregables. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

Elementos tomados de otras fuentes de conocimiento:

ISO /IEC 12207

- De la sección **Proceso de gestión de calidad** se toma como referencia sus actividades y tareas.

SWEBoK

- Se tomaron como referencia los elementos de **Calidad del software** y **Pruebas de software**.

3.6. ALPHA Recursos Humanos

La gestión de los recursos humanos tiene como objetivo conducir al equipo del proyecto para desarrollar el producto, con las habilidades y experiencia requerida para tal propósito.

En cualquier proyecto, los recursos humanos son un factor esencial para llevar a cabo las actividades y tareas correspondientes, y considerado como el mejor activo de una empresa. La selección del equipo adecuado con las habilidades y capacidades requeridas, aunado a una correcta gestión y control, amplía en gran medida el logro de los objetivos establecidos. La Figura 3.6 muestra el ALPHA Recursos Humanos.

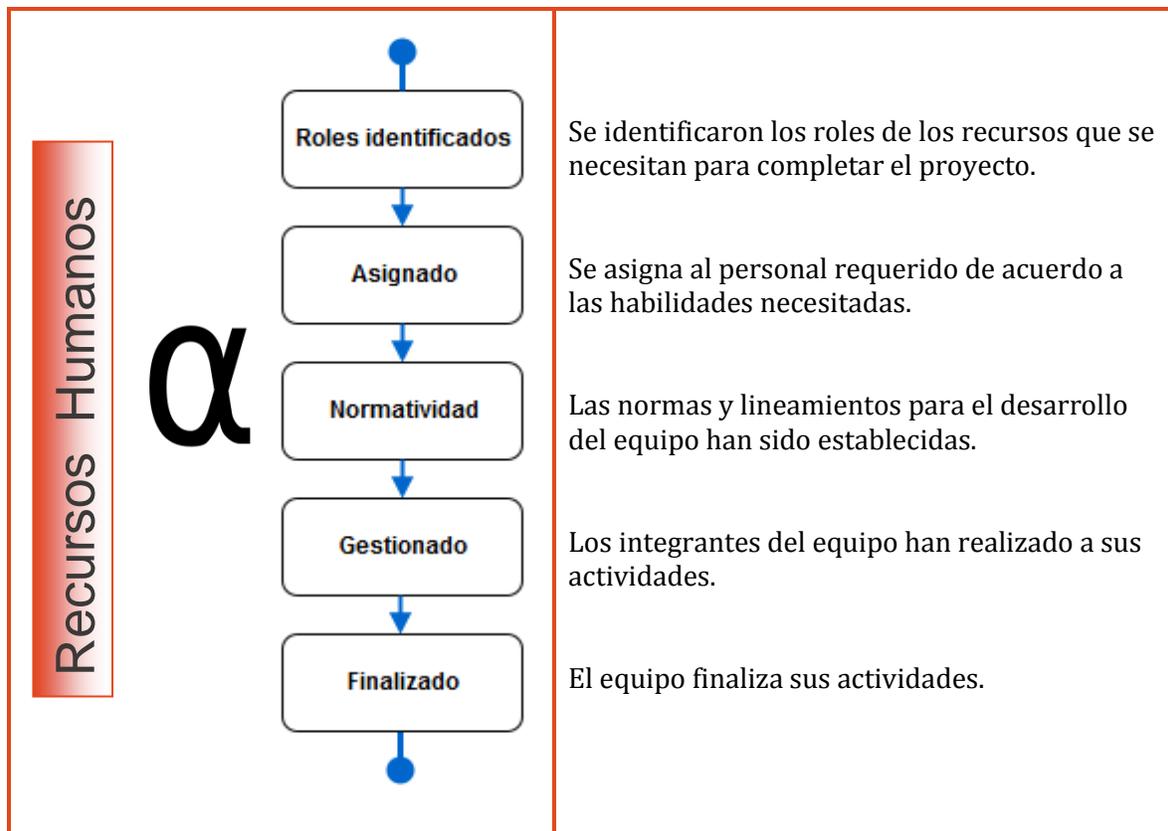


Figura 3.6. ALPHA Recursos Humanos

La Tabla 3.7 muestra la lista de verificación para el ALPHA Recursos Humanos.

Tabla 3.7. Lista de verificación del ALPHA Recursos Humanos

Estado	Lista de verificación
Roles identificados	<ul style="list-style-type: none"> <input type="checkbox"/> Se han identificado los roles necesarios para realizar el proyecto. <input type="checkbox"/> Se han identificado las habilidades y conocimientos necesarios para cada perfil de puesto. <input type="checkbox"/> Se ha determinado una estrategia para asignar a los recursos humanos.
Asignado	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha establecido el <i>Plan para la Gestión de los Recursos Humanos</i>. <input type="checkbox"/> Se han aplicado los procesos para la asignación de recursos humanos. <input type="checkbox"/> Se han evaluado a los candidatos para la asignación. <input type="checkbox"/> Los recursos humanos han sido asignados.
Normatividad	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha definido la descripción de los roles y responsabilidades de cada recurso. <input type="checkbox"/> Las reglas de trabajo se han acordado. <input type="checkbox"/> Los recursos humanos conocen y aceptan las normas establecidas.
Gestionado	<ul style="list-style-type: none"> <input type="checkbox"/> El recurso humano tiene asignado sus correspondientes actividades. <input type="checkbox"/> El recurso humano se ha acoplado a la forma de trabajo. <input type="checkbox"/> El recurso humano realiza sus actividades de manera eficiente.
Finalizado	<ul style="list-style-type: none"> <input type="checkbox"/> Se han finalizado todas las actividades correspondientes al proyecto. <input type="checkbox"/> El recurso humano está libre para ser asignado en otro proyecto. <input type="checkbox"/> El recurso humano ha entregado documentos y código fuente generado. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

Elementos tomados de otras fuentes de conocimiento:

ISO /IEC 12207

- De la sección **Proceso de gestión de recursos humanos** se toma como referencia sus actividades y tareas.

3.7. ALPHA Comunicaciones

La gestión de las comunicaciones abarca los procesos para establecer, planificar y controlar los medios para conformar una comunicación efectiva entre los interesados del proyecto, que permita transmisión de información entre ellos.

La comunicación, es la acción de transmitir información de un emisor a un receptor y en nuestro día a día es trascendental comunicarnos con el mundo. En un proyecto, forma parte del núcleo para poder intercambiar ideas, decisiones, pensamientos e integrar un puente de comunicación entre los diversos interesados para llevar a cabo las acciones pertinentes para completar satisfactoriamente el proyecto. La Figura 3.7 muestra el ALPHA Comunicaciones.

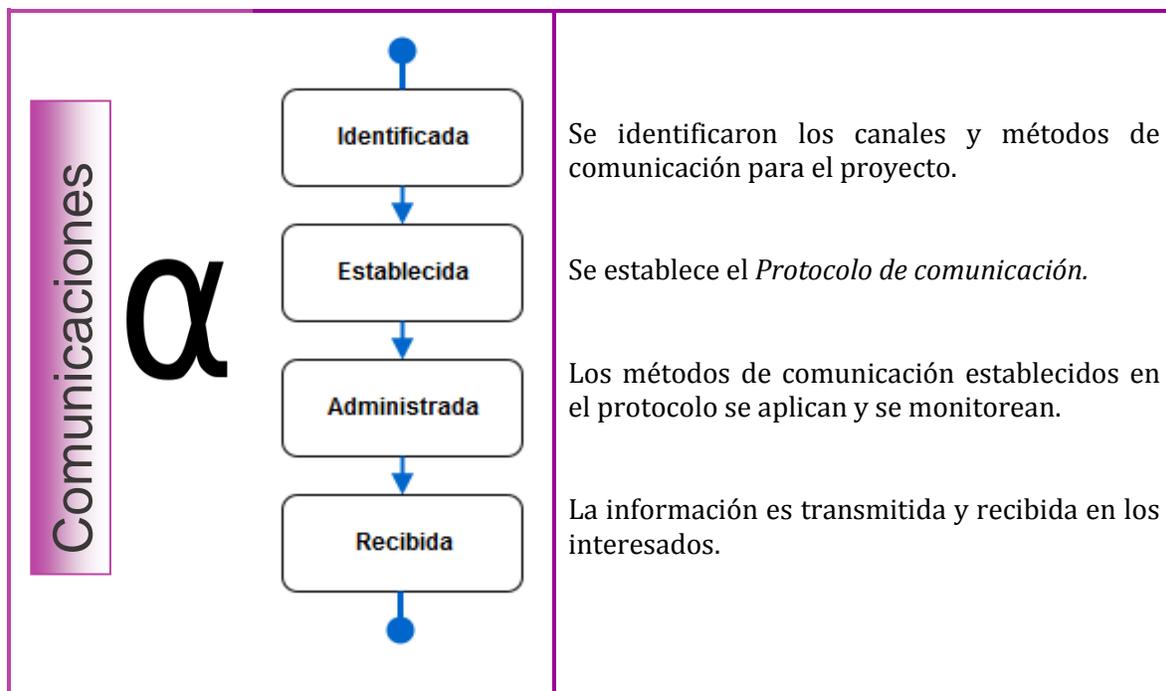


Figura 3.7. ALPHA Comunicaciones

La Tabla 3.8 muestra la lista de verificación para el ALPHA Comunicaciones.

Tabla 3.8. Lista de verificación del ALPHA Comunicaciones

Estado	Lista de verificación
Identificada	<ul style="list-style-type: none"> <input type="checkbox"/> Se identifican los canales de comunicación. <input type="checkbox"/> Se analizan los métodos y canales de comunicación.
Establecida	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha establecido una estrategia para definir la gestión de las comunicaciones. <input type="checkbox"/> Se ha analizado herramientas y técnicas para gestionar las comunicaciones. <input type="checkbox"/> El <i>Protocolo de comunicación</i> es definido. <input type="checkbox"/> Los stakeholders involucrados están en común acuerdo de lo establecido en el plan.
Administrada	<ul style="list-style-type: none"> <input type="checkbox"/> Los stakeholders conocen el funcionamiento de los canales de comunicación definidos para el proyecto. <input type="checkbox"/> Los stakeholders utilizan los canales de comunicación establecidos. <input type="checkbox"/> Las comunicaciones entre los interesados es funcional. <input type="checkbox"/> Los cambios en el <i>Protocolo de comunicación</i> son monitoreados.
Recibida	<ul style="list-style-type: none"> <input type="checkbox"/> Los interesados han enviado y recibido correctamente información mediante los canales de comunicación establecidos. <input type="checkbox"/> Las tareas y actividades de comunicación durante el proyecto han sido ejecutadas. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

En ésta ALPHA sólo se consideró al PMBoK como fuente de información.

3.8. ALPHA Riesgos

Un riesgo es un *evento o condición incierta que, de producirse, tiene un efecto positivo o negativo en los objetivos del proyecto* [11], gestionar los riesgos abarca, entre otras cosas, el identificarlos, analizarlos y tomar medidas de control o mitigación para lograr los objetivos establecidos.

Tener una adecuada gestión de riesgos durante el ciclo de vida del proyecto permitirá disminuir el impacto de los eventos negativos y aumentar los positivos. En la Figura 3.8 se muestra el ALPHA Riesgos.

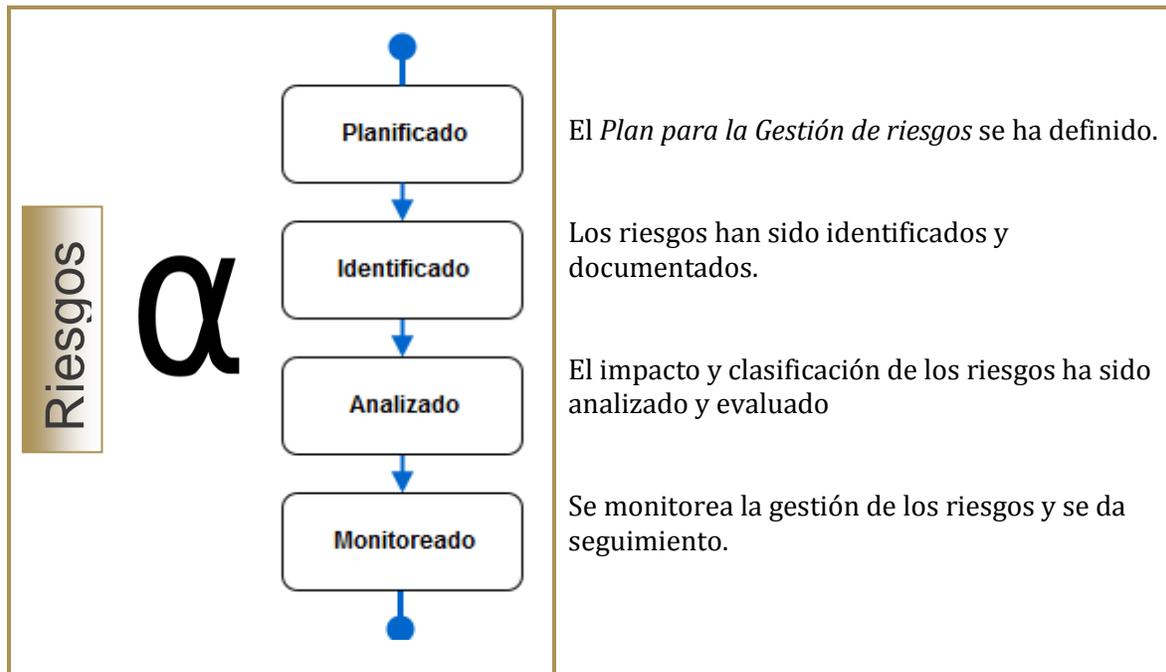


Figura 3.8. ALPHA de Riesgos

La Tabla 3.9 muestra la lista de verificación para el ALPHA Riesgos.

Tabla 3.9. Lista de verificación del ALPHA Riesgos

Estado	Lista de verificación
Planificado	<ul style="list-style-type: none"> <input type="checkbox"/> Se han descrito las directrices que ayudarán a gestionar los riesgos. <input type="checkbox"/> El <i>Plan para la Gestión de los riesgos</i> es definido. <input type="checkbox"/> Los stakeholders están en común acuerdo de lo establecido en el plan. <input type="checkbox"/> Los stakeholders conocen las categorías de riesgos.
Identificado	<ul style="list-style-type: none"> <input type="checkbox"/> Los riesgos se han identificado. <input type="checkbox"/> Se han clasificado los riesgos en las categorías establecidas. <input type="checkbox"/> Los stakeholders tienen conocimiento de los riesgos identificados.
Analizado	<ul style="list-style-type: none"> <input type="checkbox"/> Se han analizado cualitativamente los riesgos. <input type="checkbox"/> Se han analizado cuantitativamente los riesgos.
Monitoreado	<ul style="list-style-type: none"> <input type="checkbox"/> Los riesgos han sido tratados de acuerdo al plan. <input type="checkbox"/> Los riesgos identificados son continuamente monitoreados y se da seguimiento. <input type="checkbox"/> Se ha dado respuesta a los riesgos. <input type="checkbox"/> Los riesgos seleccionados de acuerdo al análisis han sido atendidos. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

Elementos tomados de otras fuentes de conocimiento:

ISO /IEC 12207

- De la sección **Proceso de gestión de riesgos** se toma como referencia sus actividades y tareas.

3.9. ALPHA Adquisiciones

En un proyecto, la adquisición de servicios o productos fuera del alcance del equipo es necesaria cuando por razones diversas no pueden ser obtenidas dentro. Para esto, la gestión de las adquisiciones incluye el control, y administración de proveedores y contratos emitidos.

Estos procesos son necesarios debido a la inminente adquisición de algún producto o servicio para el cumplimiento de las expectativas del cliente y objetivos del proyecto. La Figura 3.9 muestra el ALPHA Adquisiciones.

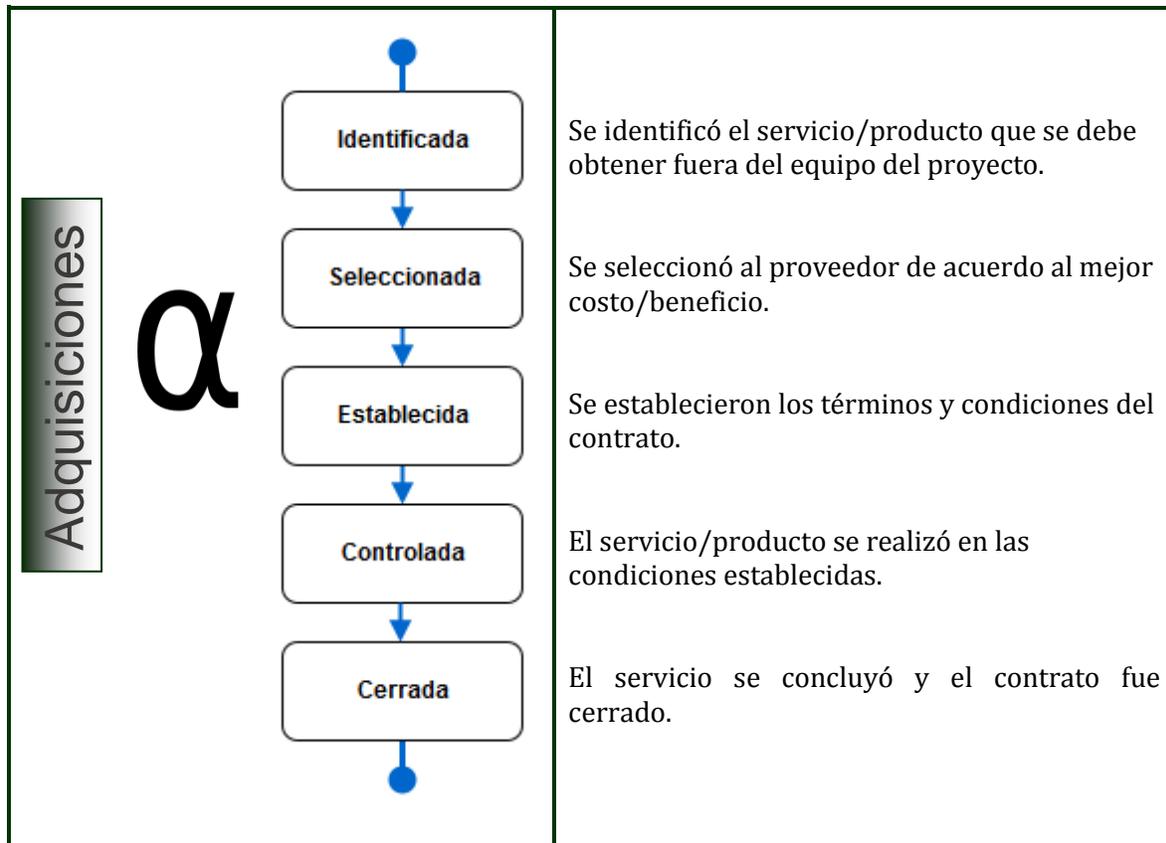


Figura 3.9. ALPHA Adquisiciones

La Tabla 3.10 muestra la lista de verificación para el ALPHA Adquisiciones.

Tabla 3.10. Lista de verificación del ALPHA Adquisiciones

Estado	Lista de verificación
Identificada	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha realizado un análisis de los requerimientos de negocio, organizaciones y técnicos para establecer cuales deben desarrollarse externamente. <input type="checkbox"/> Se han establecido y son claras las necesidades que no pueden ser completadas por el equipo. <input type="checkbox"/> El análisis de costos, riesgos y beneficios de adquirir un producto o servicio externamente se ha realizado.
Seleccionada	<ul style="list-style-type: none"> <input type="checkbox"/> Los stakeholders han determinado una estrategia para seleccionar proveedor. <input type="checkbox"/> Los criterios de evaluación para la selección de propuestas de los proveedores son establecidos por los stakeholders correspondientes. <input type="checkbox"/> El proveedor es seleccionado de acuerdo a los criterios establecidos.
Establecida	<ul style="list-style-type: none"> <input type="checkbox"/> El tipo de contrato es establecido. <input type="checkbox"/> Los requerimientos y responsabilidades son definidas y comunicadas al proveedor. <input type="checkbox"/> El costo, fechas, garantías, licencias han sido definidos en el contrato.
Monitoreada	<ul style="list-style-type: none"> <input type="checkbox"/> Las actividades del proveedor son monitoreadas para verificar el cumplimiento del contrato. <input type="checkbox"/> Los stakeholders cooperan con el proveedor para que éste realice las actividades respectivas. <input type="checkbox"/> El producto o servicio se desarrolla de acuerdo al contrato. <input type="checkbox"/> Se realizan pruebas a los entregables para verificar que cumple con lo establecido.
Cerrada	<ul style="list-style-type: none"> <input type="checkbox"/> El proveedor hace entrega del producto o servicio. <input type="checkbox"/> El producto final cumple con los requerimientos. <input type="checkbox"/> Las condiciones del contrato son cumplidas. <input type="checkbox"/> El contrato es cerrado.

Elementos tomados de otras fuentes de conocimiento:

ISO /IEC 12207

- De la sección **Proceso de adquisiciones** se toma como referencia sus actividades y tareas.

3.10. ALPHA Stakeholders

La gestión de los stakeholders identifica, controla y analiza el impacto de las personas que pueden ser afectadas por el proyecto. Mediante procesos específicos se administra la participación de dichos involucrados para completar satisfactoriamente el proyecto.

Dado que existen stakeholders que pueden tener un alto impacto en el proyecto, es importante desarrollar estrategias para gestionar sus expectativas, siendo este un objetivo del proyecto. La Figura 3.10 muestra el ALPHA Stakeholders.

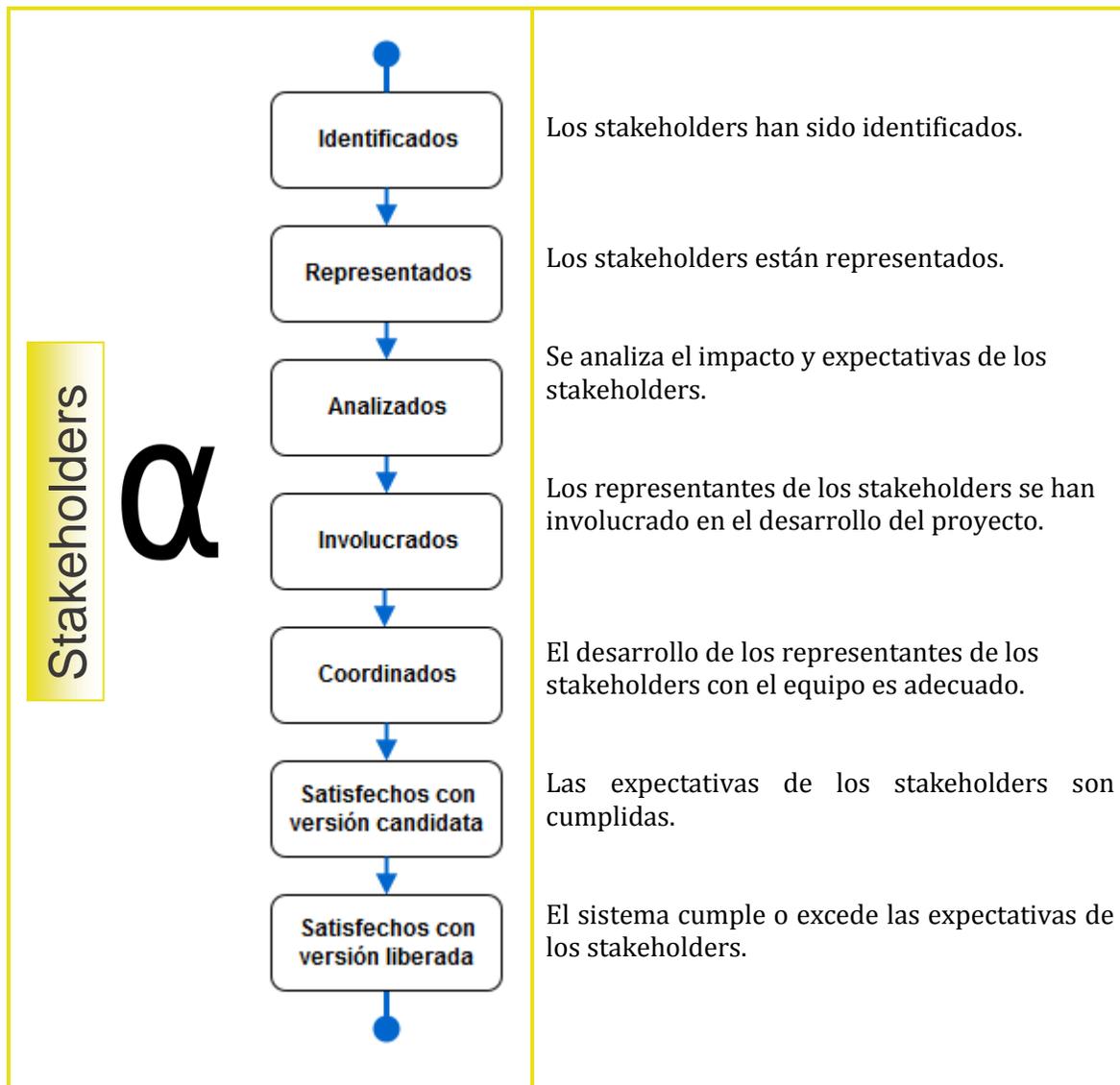


Figura 3.10. ALPHA Stakeholders

La Tabla 3.11 muestra la lista de verificación para el ALPHA de Stakeholders.

Tabla 3.11. Lista de verificación del ALPHA Stakeholders

Estado	Lista de verificación
Identificados	<ul style="list-style-type: none"> <input type="checkbox"/> Se han identificado los tipos de stakeholders pertenecientes al proyecto. <input type="checkbox"/> Los stakeholders tienen definidas sus expectativas y responsabilidad hacia el proyecto.
Representados	<ul style="list-style-type: none"> <input type="checkbox"/> Los stakeholders han identificado a sus representantes en el proyecto. <input type="checkbox"/> Los representantes de los stakeholders están de acuerdo en sus responsabilidades asignadas.
Analizados	<ul style="list-style-type: none"> <input type="checkbox"/> Se ha analizado el impacto de los stakeholders. <input type="checkbox"/> Los stakeholders han sido clasificados de acuerdo a sus expectativas, requisitos e impacto en el proyecto. <input type="checkbox"/> Se define la estrategia para su atención y cumplimiento de expectativas.
Involucrados	<ul style="list-style-type: none"> <input type="checkbox"/> Los intereses y requisitos de los stakeholders han sido controlados y monitoreados. <input type="checkbox"/> Los incidentes entre los representantes de los stakeholders han sido registrados y controlados. <input type="checkbox"/> Los representantes de los stakeholders han cumplido sus responsabilidades. <input type="checkbox"/> Los representantes de los stakeholders están en comunicación con el equipo.
Coordinados	<ul style="list-style-type: none"> <input type="checkbox"/> Los representantes de los stakeholders están conformes con el trabajo realizado. <input type="checkbox"/> Los representantes de los stakeholders están conformes con el trato recibido por parte del equipo. <input type="checkbox"/> El equipo está conforme con el trato recibido por parte de los representantes de los stakeholders.
Satisfechos con versión candidata	<ul style="list-style-type: none"> <input type="checkbox"/> Se han cumplido las expectativas y requisitos de los stakeholders en la versión del sistema de software que se planea lanzar. <input type="checkbox"/> Los representantes de los stakeholders dan su retroalimentación a la versión candidata del sistema de software.
Satisfechos con versión liberada	<ul style="list-style-type: none"> <input type="checkbox"/> Los stakeholders están satisfechos con el sistema de software que se encuentra liberado. <input type="checkbox"/> Los stakeholders confirman que el sistema de software cumple con sus expectativas. <input type="checkbox"/> Los stakeholders firman el acta de entrega del software.

Elementos tomados de otras fuentes de conocimiento:

ESSENCE

- ALPHA Stakeholders.

Capítulo 4

Resultados

El capítulo actual pretende dar a conocer los resultados obtenidos durante el desarrollo del presente trabajo, además del proceso de validación y verificación del producto generado.

El enfoque central del trabajo fue la identificación de los elementos esenciales en el desarrollo de un proyecto de software basado en el PMBOK, razón por la cual el estudio y conocimiento de dicho compendio y de otros estándares enfocados en el tema requirió de un proceso altamente analítico. Además de una evaluación cualitativa que avaló, entre otros aspectos, que cada ALPHA fuera completo en sus estados y listas de verificación.

4.1. Verificación y validación

De acuerdo al capítulo 2 se llevaron a cabo las actividades relacionadas para corroborar la estructura y semántica de las ALPHAs. Para tal efecto se realizaron revisiones continuas, entrevistas y encuestas a personas involucradas en el desarrollo de software.

El objetivo de la verificación y validación fue:

- Comprobar la adecuada estructura de cada ALPHA.
- Verificar la consistencia de los estados de las ALPHAs y sus respectivas listas de verificación.
- Confirmar que cada ALPHA contenga los elementos suficientes para monitorear el progreso de un proyecto de software.

Con el propósito de llevar a cabo la evaluación del trabajo, se seleccionó en primera instancia a un grupo de expertos que aportaran su conocimiento y experiencia en el área con la finalidad de dar validez a las ALPHAs.

El método de selección de los expertos comenzó con la identificación de un grupo de individuos que cumplieran con características especiales y que tuvieran disponibilidad para llevar a cabo el proceso de validación del trabajo.

Para tal efecto, fue necesario elegir a personas de acuerdo a los siguientes lineamientos:

- Tres expertos con experiencia en el área de TI.
- Experiencia en proyectos de desarrollo de software.
- Experiencia como líderes de proyectos de desarrollo de software.

El modo de validación se realizó mediante encuestas, en donde cada experto respondió el cuestionario mostrado en el Capítulo 2 para cada una de las ALPHAs.

Una vez establecidas las características para realizar el proceso de verificación y validación, se procedió a elegir a los expertos, haciendo de su conocimiento previamente la descripción y objetivo del presente trabajo. A continuación se presenta el perfil de cada uno de los expertos elegidos:

Experto 1

- **Carrera:** Ciencias de la Computación.
- **Conocimientos principales:** dirección de proyectos, arquitecturas empresariales, diseño e implementación de soluciones tecnológicas que involucran entre otros aspectos sistemas de información geográfica, manejo de grandes volúmenes de datos e interfaces de usuario.
- **Experiencia:** 7 años como líder de proyecto, tales como el desarrollo de la Plataforma Aurora y el Portal de Datos Abiertos UNAM, coordinando actividades administrativas y técnicas en los equipos de trabajo. Vasta experiencia como desarrollador de software en diversos lenguajes de programación: Java, C++, C, PHP.
- **Roles desempeñados:** Líder de proyecto, desarrollador y analista de software.
- **Sector:** Experiencia primordialmente en proyectos de instituciones con enfoque educativo pertenecientes a la Universidad Nacional Autónoma de México.

Experto 2

- **Carrera:** Informática Administrativa.
- **Conocimientos principales:** posee la certificación Project Management

Professional (PMP) que otorga el PMI¹⁹, que la avala como profesional en la administración de proyectos.

- **Experiencia:** Experiencia de 11 años como líder de proyectos de TI en empresas privadas, así como de coordinación e impartición de diplomados en Administración de Proyectos.
- **Roles desempeñados:** Líder de Proyecto, desarrollador de software.

Sector: Experiencia el sector privado, en empresas como MetLife, Bancomer, GlobalCard.

Experto 3

- **Carrera:** Actuaría
- **Conocimientos principales:** posee la certificación Project Management Professional (PMP) que otorga el PMI, que la avala como profesional en la administración de proyectos.
- **Experiencia:** 11 años como líder de proyectos de TI, así como de coordinación e impartición de diplomados en Administración de Proyectos.
- **Roles desempeñados:** Líder de proyecto.

Sector: Experiencia en proyectos con fines culturales, de educación y gubernamental.

Para tener un resultado más certero y que el juicio emitido por los expertos no tuviera sesgos, se decidió optar por una evaluación individual, es decir, se obtuvo información de cada experto sin que estos tuvieran contacto entre sí.

Realizar la validación, conllevó entrevistar a cada experto, tomando en consideración factores como el lugar y hora de fácil acceso para ellos, además de proporcionar un adecuado ambiente para realizar la entrevista.

Finalizadas las entrevistas y respondidas las encuestas (Figura 4.1), se generó un compendio de las observaciones emitidas por los expertos, que posteriormente fueron evaluadas para determinar las modificaciones que se realizarían a las ALPHAs.

¹⁹ Project Management Institute, organización que asocia a profesionales relacionados con la administración de proyectos.

Figura 4.1. Encuestas

Cabe resaltar que la construcción de las ALPHAs fue un proceso iterativo que requirió de constantes ajustes en su estructura y semántica para lograr una versión candidata de cada ALPHA, las cuales fueron sometidas al juicio de expertos previamente descrito. Los defectos encontrados se registraron en el formato establecido para ello (ver Figura 4.2).

Nombre	Experto1	Fecha	31/10/2015
ALPHA	Todas		
Defectos			
1. Los nombres de los estados no están unificados.			
2.			
3.			
Observaciones			
Los nombres de los estados deberán ser todos en singular o todos en plural. Se deberá asignar un color a cada ALPHA para su identificación.			

Nombre	Experto 2	Fecha	06/11/2015
ALPHA	Tiempo		
Defectos			
1. La descripción del estado "Finalizado" no expresa correctamente lo que se necesita.			
2.			
3.			
Observaciones			

Nombre	Experto 3	Fecha	25/10/2015
ALPHA	Comunicaciones		
Defectos			
1. En la lista de verificación hace falta agregar lo referente a "Canales de comunicación identificados".			
2.			
3.			
Observaciones			

Figura 4.2. Defectos registrados

Del proceso de juicio de expertos se desprendieron diversas observaciones referentes, principalmente, a la semántica de las ALPHAs. Cada experto de acuerdo a su área de experiencia aportó sus conocimientos en la mejora de estos elementos, proponiendo lo que consideraban de mayor relevancia en la práctica, de manera general se listan las siguientes observaciones:

- Delimitar y definir específicamente elementos de las listas de verificación.
- Añadir elementos a las listas de verificación y así completarlas.
- Modificar nombres de estado que no eran representativos.
- Modificar términos de acuerdo al área en cuestión por otros que definan de mejor manera lo que se desea expresar.
- Unificar los nombres de los estados de las ALPHAs en singular.

Del conjunto previo de observaciones, resultaron las siguientes modificaciones:

- Cambió el nombre del estado anteriormente denominado “Controlado” a “Monitoreado”, de las ALPHAs: “Comunicaciones”, “Costos”, “Integración”, ya que plasma con mayor exactitud la transición del ALPHA.
- Se agregó el elemento “Se han recopilado lecciones aprendidas” a la mayoría de las listas de verificación.

Otro resultado de estas entrevistas fue que los expertos sugirieron describir brevemente el camino que debe considerarse durante el uso de las ALPHAs, lo que ayudaría a los profesionales a iniciarse en la administración de proyectos de software.

El proceso de validación y verificación, se complementó mediante actividades que generaran una retroalimentación directa de profesionales involucrados activamente en proyectos y tareas de administración de éstos. Para llevar a cabo esta tarea se presentaron las ALPHAs a dos grupos de usuarios, de 22 y 21 personas respectivamente, para analizar la estructura, nombrado y suficiencia de los estados, y la pertinencia de las listas de verificación. La información obtenida de estos grupos permitió mejorar las ALPHAs y derivó en comentarios positivos.

De esta manera se validó y verificó que las ALPHAs cumplen con la estructura descrita en el capítulo 2, y que sus estados y listas de verificación son suficientes.

4.2. ALPHAs

El resultado del trabajo derivó en el conjunto de ALPHAs, conformadas por sus estados en la que cada una transita y las respectivas listas de verificación. Teniendo las características y beneficios siguientes:

- Al basarse en estándares internacionales, principalmente ESSENCE y PMBOK, se contemplaron los temas y áreas comúnmente utilizados en los proyectos de software.

- Están sustentadas en las 10 áreas de conocimientos del PMBOK, las cuales representan las aristas esenciales en la administración de proyectos.
- Se identificaron los estados que indican el punto en que se encuentra el desarrollo del proyecto, dentro de las diversas áreas que lo componen.
- El uso de las ALPHAs no se restringe a determinadas metodologías de desarrollo de software.
- Un adecuado uso de las ALPHAs podría ayudar que la administración de un proyecto de software sea más eficiente y práctico.
- Los profesionales poseen una base sólida que les permite conocer una perspectiva general para abordar de mejor manera un proyecto de software.

En la figura 4.3 se pueden observar las ALPHAs generadas.

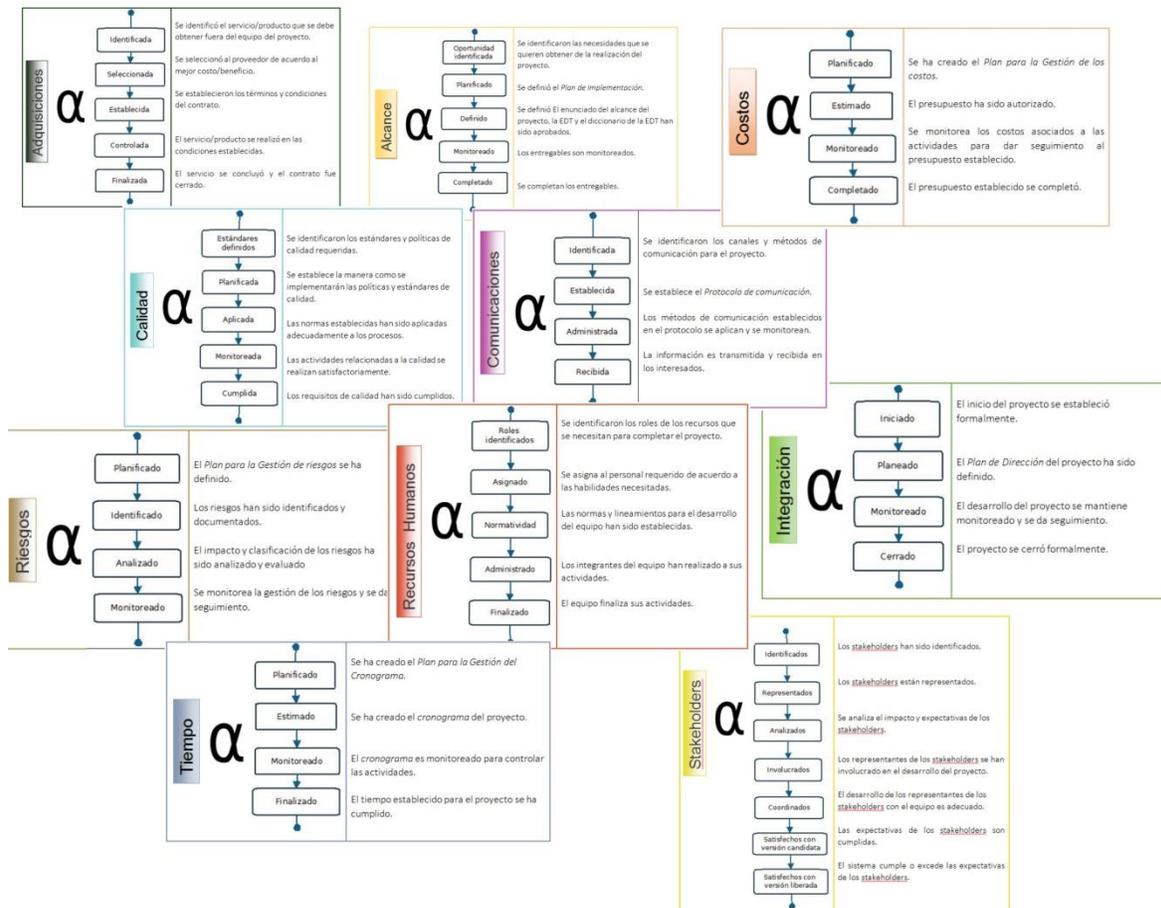


Figura 4.3. ALPHAs

4.3. Guía de uso de ALPHAs

El objetivo de las ALPHAs es el monitoreo y control de un proyecto de software, para tal efecto es indispensable que los profesionales que utilicen dichos elementos tengan conocimiento acerca de los conceptos centrales de la administración de proyectos, esencialmente del PMBOK, de tal forma que les sea posible utilizar con mayor facilidad las ALPHAs.

Con el propósito de proveer a los usuarios de esos elementos, a manera de guía, se dan a conocer los aspectos esenciales que deben considerarse.

4.3.1. Estructura del ALPHA

El conjunto de ALPHAs está conformado por 10 elementos, siendo estos:

- Integración
- Alcance
- Tiempo
- Costos
- Calidad
- Recursos Humanos
- Comunicaciones
- Riesgos
- Adquisiciones
- Stakeholders

Cada uno está asociado con un determinado color que favorece su ubicación y practicidad durante su uso.

Una ALPHA cuenta una representación gráfica compuesta por el **nombre, estados de transición** y la correspondiente descripción del estado, sea por ejemplo, el ALPHA Tiempo (figura 4.4) identificado por el color azul.

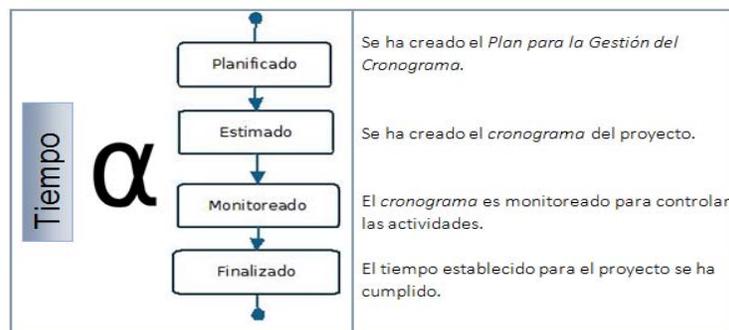


Figura 4.4. Estructura ALPHA Tiempo

Además cuenta con una **lista de verificación** para cada estado de transición del ALPHA y un conjunto de elementos que determinan el cumplimiento del estado (figura 4.5).

Estado	Lista de verificación
Planificado	<input type="checkbox"/> Se ha establecido una estrategia para gestionar el tiempo del proyecto. <input type="checkbox"/> El <i>Plan para la Gestión del Cronograma</i> es definido. <input type="checkbox"/> Los stakeholders están en común acuerdo con las fechas establecidas en el plan.
Estimado	<input type="checkbox"/> Las actividades requeridas para completar el proyecto han sido definidas. <input type="checkbox"/> Las actividades han sido secuenciadas. <input type="checkbox"/> Los recursos han sido asignados a cada actividad. <input type="checkbox"/> La duración ha sido estimada para cada actividad. <input type="checkbox"/> El <i>cronograma</i> ha sido desarrollado. <input type="checkbox"/> Los stakeholders conocen y han aceptado el cronograma. <input type="checkbox"/> El cronograma ha sido aprobado.
Monitoreado	<input type="checkbox"/> El <i>cronograma</i> es monitoreado y se da seguimiento <input type="checkbox"/> El equipo ha realizado sus actividades acorde al <i>cronograma</i> . <input type="checkbox"/> Se ha ajustado el <i>cronograma</i> de acuerdo a los cambios realizados al proyecto. <input type="checkbox"/> Los cambios hechos al <i>cronograma</i> son validados.
Finalizado	<input type="checkbox"/> Las actividades correspondientes para completar el proyecto han finalizado. <input type="checkbox"/> El tiempo establecido en el cronograma ha finalizado. <input type="checkbox"/> Se han recopilado lecciones aprendidas.

Figura 4.5. Listas de verificación

4.3.2. Recomendaciones

- Reconocer todas las ALPHAs previo a su utilización.
- Identificar la estructura y elementos que componen un ALPHA.
- En caso de necesitar mayor granularidad durante el desarrollo de cierta área de conocimiento, puede utilizarse un ALPHA más de una vez en el proyecto. Por ejemplo, si se identifican tres riesgos es posible crear tres instancias del ALPHA Riesgo, una para cada riesgo. Esto facilitará el monitoreo de cada uno de manera individual.
- En caso, de utilizar un ALPHA repetidamente en un proyecto, se recomienda identificarla de acuerdo a la etapa perteneciente con el fin de evitar confusiones futuras. Es decir, determinar claramente donde empieza y termina la vida de cada una.
- Utilizar el conjunto de ALPHAs alternándolo con herramientas y métodos de trabajo que se adapten al estilo de los equipos de desarrollo y metodología del proyecto.
- Un proyecto de software es único, por tal razón establecer un camino específico para el uso de ALPHAs sería inadecuado. Sin embargo es importante señalar que iniciar con

Integración y Alcance, da la pauta para ir abordando el proyecto y consecuentemente la utilización del resto de las ALPHAs.

4.4. Observaciones finales

El camino recorrido para la construcción de un ALPHA necesitó de innumerables análisis, revisiones y correcciones que derivaron en la versión previamente presentada de este conjunto de elementos esenciales en un proyecto de software. Para identificar cada ALPHA dentro del cuerpo de conocimiento que el PMBOK provee, se decidió basarlas en las áreas de conocimiento en que se agrupan los procesos involucrados en la administración de un proyecto, lo cual implicó profundizar en la mayoría de los apartados de ésta guía de fundamentos.

Construir cada ALPHA requirió el considerar, además del PMBOK, otros cuerpos de conocimiento de igual reconocimiento internacional que aportaran datos específicos sobre desarrollo de software, así una vez finalizada dichas ALPHAs se sometieron a revisiones antes de ser expuestas al juicio de expertos. Este proceso se concretó en un lapso considerable, debido a que implicó realizar actividades tales como entrevistas, encuestas, registro de errores y correcciones, pero la labor que desempeñó cada experto y el conocimiento que aportaron resultó benéfico para validarlas. De esta manera se obtuvo un producto preciso, completo y confiable, además de ser eliminados los defectos encontrados en cada ALPHA. Cabe resaltar que los expertos en todo momento mostraron accesibilidad durante tal proceso, y se lograron los resultados esperados.

Durante el desarrollo de ALPHAs, se hicieron presentes ciertos aspectos como:

- La identificación de estados conllevó un análisis exhaustivo para delimitarlos, nombrarlos y ordenarlos adecuadamente, ya que de no realizar esto con atención genera gran confusión por la inmensa información y los procesos asociados en las áreas de conocimiento del PMBOK.
- Agrupar e identificar cada lista de verificación para los estados del ALPHA, mereció de igual manera un análisis profundo en su clasificación para evitar ambigüedades en los elementos porque en caso contrario, un elemento de una lista de verificación podría indicar pertenecer a dos estados contiguos.

En el transcurso de este trabajo se fueron minimizando estos aspectos y se corrigieron los defectos encontrados, se logró así generar satisfactoriamente las ALPHAs que representan un apoyo a los profesionales de TI para gestionar sus proyectos.

El uso de ALPHAs en los proyectos de software es un punto de partida para abordarlo de mejor manera y dado que son lo suficientemente flexibles pueden ser utilizados con gran cantidad de metodologías de desarrollo de software y así aumentar las probabilidades de éxito en los proyectos.

Conclusiones

El fin primordial de la construcción de las ALPHAs fue proveer un conjunto de elementos a los profesionales de TI, específicamente aquellos que encabezan equipos dedicados al desarrollo de software, para ser utilizados en sus proyectos como herramientas para el monitoreo y control.

El desarrollo del presente trabajo pretende impulsar las buenas prácticas en los proyectos que día a día se realizan en el mundo del software. Estas ALPHAs, utilizadas correctamente, proveerán una alternativa en la disciplina que provee de un núcleo que representa la esencia en los proyectos de software sustentados en estándares reconocidos internacionalmente. Siendo éste un punto favorable en su uso, puesto que en la actualidad la industria de software está en pleno auge y es importante que se manejen conceptos y términos reconocidos globalmente.

De tal forma, las ALPHAs representan un punto de partida para guiar la gestión de un proyecto, además de proporcionar bases para monitorear su progreso con una perspectiva ágil, concisa e independiente a las prácticas utilizadas por los equipos de trabajo.

Durante el proceso de creación de las ALPHAs tuve un acercamiento con dos estándares en los cuales se basó mi trabajo: uno de ellos fue ESSENCE, que proporcionó los principios y conceptos acerca del significado y objetivo de un ALPHA en los proyectos de software; el segundo fue el PMBOK, que hoy en día es una de las guías más conocidas de gestión de proyectos. El conocer ambos estándares me dejó un gran aprendizaje en estos temas, clarificando su relevancia al aplicarlos en un proyecto y, a su vez, derivar en el éxito o fracaso del mismo.

Aunque aún falta mucho por hacer, la construcción de las ALPHAs es un paso más en el camino para proveer a la industria de mejores prácticas y cerrar brechas entre la teoría y la práctica, ya que actualmente se padece de una falta de madurez en la Ingeniería de Software. Esto implica un inmenso esfuerzo por parte de todos lo que estamos involucrados en esta profesión para erradicar malas prácticas y la percepción incorrecta que existe sobre la aplicación de los métodos y procesos, los cuales, en ocasiones, son considerados innecesarios por los mismos desarrolladores de software.

Realizar este trabajo de tesis conllevó una serie de actividades que evidenciaron la importancia que representa la carrera de Ciencias de la Computación. Fue indispensable tener bases teóricas y prácticas especialmente de Ingeniería de Software, además de experiencia desarrollando software. El ser una disciplina tan amplia, me permitió aplicar los conocimientos adquiridos en materias como ICC I y II, Lenguajes de Programación y sus Paradigmas, Sistemas Operativos, Bases de Datos, entre otras. Además de la capacidad de abstracción y análisis que proporciona el fundamento matemático presentado en el transcurso de la licenciatura.

Desde mi experiencia como egresada de esta carrera, el aprendizaje adquirido me ha permitido desarrollarme profesionalmente en diversas áreas y generar soluciones a problemas computacionales en proyectos del sector privado así como en proyectos realizados dentro de esta Universidad.

Anexos

Anexo 1

Nombre	<i>Nombre del revisor</i>	Fecha	<i>dd/mm/aaaa</i>
ALPHA	<i>Nombre del ALPHA</i>		
Defectos			
1.			
2.			
3.			
Observaciones			

Formato para el registro defectos encontrados en las ALPHAs.

Anexo 2

Encuesta para la validación de ALPHAs.

ENCUESTA	
1. ¿El nombre del ALPHA es representativo?	
2. ¿Los estados están ordenados correctamente?	
3. ¿Los nombres de los estados son representativos?	
4. ¿Agregarías/quitarías algún estado?	
5. ¿Las listas de verificación son suficientes?	
6. ¿Agregarías/quitarías algún elemento de las listas de verificación?	

Glosario

ALPHA	Acrónimo de “Abstract Level Progress and Health Attribute”
BABOK	Acrónimo de “Business Analysis Body of Knowledge”
CCMI	Acrónimo de “Capability Maturity Model Integration”
EDT	Acrónimo de “Estructura de Desglose de Trabajo”
ESSENCE	Estándar internacional que proporciona definiciones y descripciones, y un lenguaje de los métodos de ingeniería de software.
EITBOK	Acrónimo de “Enterprise Information Technology Body of Knowledge”
ISO	International Organization for Standardization. Organización que promueve el desarrollo de normas internacionales.
Kanban	Método para gestionar el trabajo mediante el uso de herramientas visuales para el equipo.
Kernel	Núcleo visto desde cualquier enfoque constituye parte fundamental de algo, en este caso es la esencia de la ingeniería de software.
PMBOK	Acrónimo de “Project Management Body of Knowledge”
PMI	Acrónimo de “Project Management Institute”
SEBOK	Acrónimo de “Systems Engineering Body of Knowledge”
SWEBOK	Guide to the Software Engineering Body of Knowledge
UML	Acrónimo de “Unified Modeling Language”

Bibliografía

- [1] Pressman, Roger S., *Ingeniería del software. Un enfoque práctico*, Mc Graw Hill, 2010, séptima edición.
- [2] Object Management Group, *ESSENCE – Kernel and Language for Software Engineering Methods 1.0*, 2014.
- [3] Unified Modeling Language (UML), Resource Page, <http://www.uml.org/> (Visitado: 02/09/2015).
- [4] Capability Maturity Model Integration, <http://cmmiinstitute.com> (Visitado: 02/09/2015)
- [5] K. Schwaber, J. Sutherland, *The Scrum Guide – The Definitive Guide to Scrum: The Rules of the Game*, 2011, <http://www.scrumguides.org/> (Visitado: 02/09/2015)
- [6] eXtreme Programming , <http://www.extremeprogramming.org/> (Visitado: 02/09/2015).
- [7] Anderson, David J., *Kanban. Succesful Evolutionary Change for Tecnology Organizations*. 2010
- [8] Jacobson, Ivar; Meyer Bertrand. SEMAT vision statement.
- [9] ISO/IEC 12207, *Systems and software engineering — software life cycle processes*, International Organization for Standardization, 2008.
- [10] ISO/IEC 15504, *Information technology — process assessment*, International Organization for Standardization, 2004.
- [11] *A Guide to the Project Management Body of Knowledge: PMBOK Guide*, Project Management Institute, Fifth Edition, Project Management Institute, Newtown Square, Pennsylvania, 2013.
- [12] *Guide to the Software Engineering Body of Knowledge*, version 3.0, SWEBOK, IEEE, 2014.

- [13] The University of Queensland, Australia. Types of information sources, <https://www.library.uq.edu.au/how-to-guides/types-information-sources> (Visitado: 10/10/2015).
- [14] University of Maryland. Primary, Secondary and Tertiary sources, <http://www.lib.umd.edu/tl/guides/primary-sources#primary> (Visitado: 10/10/2015).
- [15] ISO/IEC, 29110-5-1-2:2012 Software engineering — lifecycle profiles for very small entities (VSEs) — Management and engineering guide: generic profile group: basic profile, International Organization for Standardization, 2012.
- [16] International Organization for Standardization, ISO/IEC 15288:2008 Systems and software engineering – System life cycle processes, 2008.
- [17] International Organization for Standardization, ISO/IEC TR 24774:2010 Systems and software engineering – Life cycle management – Guidelines for process description, 2010.
- [18] Revista Nebrija. La validación por juicio de expertos. <http://www.nebrija.com/revista-linguistica/la-validacion-por-juicio-de-expertos-dos-investigaciones-cualitativas-en-linguistica-aplicada> (visitado: 15/12/2015).
- [19] Agile Software Development, Alistair Cockburn, 1st edition, 2001, Addison-Wesley Professional