

03063
3



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

POSGRADO EN CIENCIA E INGENIERIA DE LA COMPUTACION

**GUIA PARA LA ADMINISTRACION DE PROYECTOS
PEQUEÑOS DE SOFTWARE, CUMPLIENDO CMM-2
Y GUIADOS POR EL PMBOK**

T E S I S

QUE PARA OBTENER EL GRADO DE:
MAESTRA EN CIENCIAS DE LA COMPUTACION
P R E S E N T A
BLANCA LUCIA GIL CASTELLANOS

DIRECTORA DE LA TESIS: DRA. HANNA OKTABA

MEXICO, D.F.

0.82.137

2000



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.

**Dedicado a mis padres,
Teresa y Fidencio,**

**a mis hermanos, a mis
sobrinas y a mi sobrino.**

Agradecimientos:

En primer lugar, quiero agradecer a la Dra. Hanna Oktaba, por su tenacidad, su total disposición a ayudarme, y por el apoyo incansable que siempre me brindó, en la realización de este trabajo. Gracias Hanna.

También quiero agradecer a mis sinodales: M. en C. Guadalupe Ibarguengoitia, Dr. Manuel Romero Salcedo, Dr. Fernando Gamboa Rodríguez y Dr. Bárbaro Jorge Ferro Castro, por sus consejos, su tiempo y dedicación. A la Dra. María Garza por la gran confianza que siempre me brindó. A Lulú por su disposición, su amabilidad y ayuda, lo mismo que Rodrigo y a Gustavo.

Bianca Gil

ÍNDICE

1. Introducción	1
Objetivo Prefacio Antecedentes de este trabajo Descripción del contenido de los capítulos	
<hr/>	
PRIMERA PARTE. BASE TEÓRICA	5
2. Marco teórico	
2.1. Administración de Proyectos	5
Concepto de Proyecto Concepto de Administración	
2.1.1. Fases y ciclo de vida de un proyecto	6
Entregables Metodologías Involucrados Líneas base	
2.1.2. El proceso de la administración de proyectos	7
Concepto de Proceso Procesos de la Administración de Proyectos	
2.1.3. El proceso de software	10
Proceso de Software Componentes del Proceso de Software	
2.2. Cuerpo de Conocimiento de la Administración de Proyectos	11
Definición Áreas que lo componen	
2.2.1. Integración de los Procesos de la administración de proyectos con las áreas del PMBOK	14
Procesos de Iniciación Procesos de Planeación Procesos de Ejecución Procesos de Control Procesos de Finalización.	
2.2.2. Entradas, técnicas y salidas de los procesos del PMBOK	18
2.3. La administración de proyectos en el nivel 2 del CMM	24
El CMM La Capacidad del Proceso de Software	
2.3.1. Organizaciones de software maduras e inmaduras	24
Organizaciones inmaduras Organizaciones maduras	
2.3.2. Los niveles de maduración	25
Niveles de maduración Áreas claves del Proceso. Esquema general de las Áreas Clave del Proceso	
2.3.3. Los niveles 1 y 2 del CMM	28
Características generales del nivel Inicial. Características generales del nivel Repetible. Áreas claves del proceso del nivel 2. Repetible.	
2.3.4. Esquema para el nivel 2 del CMM, Repetible	30

2.4. Pautas para el proceso de desarrollo	36
El Paradigma estructurado. El Paradigma orientado a objetos.	
2.4.1. El Desarrollo Estructurado	38
2.4.2. El Proceso Unificado de Desarrollo de Software	40
2.4.3. Las Métricas en la Administración de Proyectos	41
Definiciones. Métricas Genéricas	
2.4.4. Evaluación y pruebas en la Administración de Proyectos	44
Definición del Proceso de Pruebas. Definición de Calidad. Definición de Evaluación. Diferencia entre corregir y probar. Pruebas Genéricas. Criterios de Validación.	
2.4.5. Marcos y patrones	49
Concepto de Patrón. Concepto de Marco (Frame). Patrones y Antipatrones de Proceso. Patrones de Organización.	

SEGUNDA PARTE. GUÍA PARA LA ADMINISTRACIÓN DE PROYECTOS DE SOFTWARE

3. Presentación de la Guía	51
3.1. Ciclo de Vida utilizado para esta Guía	51
3.1.1. Ciclo de Vida	51
Fases.	
3.1.2. Roles utilizados en esta Guía	52
Administrador Principal. Administrador del Proyecto. La Administración de la Configuración del SW. Administrador de la Configuración del SW (ACS). Grupo de Administración de la Configuración del SW (GACS). Administrador de la Calidad del SW (AQS). Grupo de Administración de la Calidad del SW (GAQS). Líder del Proyecto.	
3.1.3. Ciclo de Vida con procesos del PMBOK	53
3.2. Marcos y patrones utilizados en esta Guía	54
3.2.1. Descripción del Marco propuesto para la Guía	54
3.2.2. Descripción del Patrón propuesto para la Guía	55
3.2.3. Descripción de la Tabla Resumen de la Guía	56
4. Fase de Inicio	57
4.1. Marco para la Fase de Inicio	57
4.2. Patrones por rol para la Fase de Inicio	60
El Administrador Principal en la Fase de Inicio. El Administrador del Proyecto en la Fase de Inicio. El Administrador de la Configuración del Software en la Fase de Inicio. El Administrador de la Calidad en la Fase de Inicio. El Líder de Proyecto en la Fase de Inicio.	

5. Fase de Elaboración y de Construcción.	85
5.1. Marco para la Fase de Elaboración.	85
5.2. Marco para la Fase de Construcción.	88
5.3. Patrones por rol para la Fase de Elaboración y Construcción.	91
El Administrador Pncipal en la Fase de Elaboración El Administrador del Proyecto en la Fase de Elaboración El Administrador de la Configuración del Software en la Fase de Elaboración El Administrador de la Calidad en la Fase de Elaboración El Líder de Proyecto en la Fase de Elaboración	
6. Fase de Transición.	117
6.1. Marco para la Fase de Transición.	117
6.2. Patrones por rol para la Fase de Transición.	120
El Administrador Pncipal en la Fase de Transición El Administrador del Proyecto en la Fase de Transición	

Conclusiones.	129
Trabajos posteriores	
Bibliografía.	131
Anexo.	135
A Diagrama de Actividades	

CAPÍTULO 1. Introducción

OBJETIVO

Presentar una metodología para construir software de forma que su desarrollo sea planteado como un proyecto, y poder aplicar las mejores prácticas en la Administración de Proyectos que el "Project Management Body of Knowledge" (PMBOK) propone, incluyendo los requisitos para cumplir el nivel 2 del CMM. Esta metodología estará dirigida principalmente a proyectos pequeños y medianos

PREFACIO

Está de más resaltar la importancia que cada vez ganan los sistemas de software, dentro de las organizaciones

Los sistemas están presentes y los vemos como un hecho tecnológico de la vida. En muchos casos, dejamos nuestro trabajo, bienestar, seguridad, entretenimiento, decisiones y nuestras propias vidas en manos de un sistema de software. Aún ante la clara necesidad de construir programas fiables y de alta calidad, nuestra habilidad para construirlos no puede ir al ritmo de la demanda y al mismo tiempo cumplir con las necesidades del mercado

Existe un gran problema en mantener y mejorar los programas existentes a causa de diseños pobres y de recursos inadecuados y mal administrados. El incumplimiento a nuestros requerimientos de productos de software más complejos y de mayores capacidades, recae en el cómo desarrollamos software.

Un proyecto es un empeño o esfuerzo temporal cuyo objetivo es crear un producto o servicio único. Temporal significa que cada proyecto tiene un principio y un final bien definidos. Es único puesto que el producto o el servicio es notoriamente diferente de otros productos o servicios similares.

Los proyectos existen en todos los niveles de una organización. Involucran a sólo una persona o a muchas. Su cumplimiento puede requerir menos de 100 horas o hasta varios años. Es posible que en los proyectos colabore una sola unidad de la organización o bien, muchas de ellas cruzándose los límites del organigrama de una o varias organizaciones. Por lo general, son componentes críticos en el éxito estratégico del negocio de la empresa

Necesitamos productos de alta calidad. Para los fines de este trabajo, podemos decir que calidad es satisfacer en concordancia y enteramente, las necesidades del usuario en el menor tiempo posible y utilizando el menor número de recursos (cumplimiento, rapidez, economía y correctez). Para lograr esto, un camino posible a seguir y que ha mostrado sus logros en el mundo real, es hacer uso de la experiencia de aquéllos que nos anteceden. Las mejores prácticas agrupan este conocimiento, probado exhaustivamente en la industria, generado de las experiencias de aquéllos que están inmersos en los procesos de aseguramiento de calidad.

Existen, para la administración de proyectos, las mejores prácticas esquematizadas en varios modelos. Uno de ellos es el Project Management Body of Knowledge, mejor conocido como PMBOK, desarrollado por el Project Management Institute (PMI) [PMI96]. Estas prácticas representan la suma del conocimiento dentro de la profesión de la administración de proyectos. Este conocimiento es en forma general bien aceptado puesto que, junto con las prácticas que describe, aplican a la mayoría de los proyectos de la actualidad, según un amplio consenso.

Con respecto a la forma en cómo desarrollamos los productos de software, también existen esfuerzos para conjuntar las mejores prácticas o conocimientos, y proponer métodos cuya efectividad y eficacia estén avalados por un grupo importante de profesionales de la materia. Uno de estos métodos, es el "Unified Software Development Process" (JAC99), el cual conjunta el esfuerzo de varios especialistas que venían trabajando en forma independiente, presentando sus propuestas para mejorar el proceso de desarrollo del software.

Por otra parte, las metodologías tradicionales actualmente siguen teniendo una gran aplicación efectiva, razón por la cual no podemos ni debemos omitirlas dentro de esta propuesta.

El objetivo principal es el desarrollo de un sistema de software. Pero, el desarrollo de un sistema de software es un proyecto (o bien, un proyecto puede ser el desarrollo de un sistema de software).

Es decir, que el proceso de desarrollo de un sistema de cómputo es un proyecto (o viceversa), y por lo tanto podemos integrar a las mejoras prácticas de la administración de proyectos que el PMBOK nos ofrece, los métodos tradicionales o unificados de desarrollo de software.

Finalmente, este acercamiento a la administración de proyectos, debe estar regido por modelos de calidad. Varios grupos de expertos en la materia han propuesto esquemas, para asegurar ciertos niveles de calidad. Uno de estos modelos es el Capability Maturity Model, que presenta guías para mejorar el Proceso de Software. Buscaremos, entonces, aplicar estas guías de Calidad al Proceso de la Administración de Proyectos de Software, bajo los lineamientos que ya mencionamos.

ANTECEDENTES DE ESTE TRABAJO

Este trabajo se originó como resultado de la experiencia vivida, en el desarrollo de un proyecto escolar dentro de mis estudios de Maestría.

En la clase de "Ingeniería de Software Orientado a Objetos", impartida por la M. en C. Guadalupe Ibarquengoitia y por la Dra. Hanna Oktaba (en el segundo semestre de 1999), el objetivo planteado fue el de simular todo el proceso de creación de un producto de software, tan fiel como fuese posible en una empresa dedicada al desarrollo del software. Así que desde un principio se estableció un proyecto para un cliente real, el cual debería ser desarrollado siguiendo las propuestas de la Ingeniería de Software recientemente vistas en clase, y usando el "Proceso Unificado de Desarrollo de Software". Para realizar dicho proyecto, se crearon equipos de trabajo auxiliados por Técnicas de Grupo, se propusieron roles por etapas para ser realizados por uno o varios alumnos (de igual manera, un alumno podía ejecutar uno o más roles), y hubo libertad para escoger dichos roles.

En forma general, los roles iniciales fueron: Administrador del Proyecto, Administrador de la Configuración, Administrador de la Calidad, Líder de Proyecto y un Grupo de Ingeniería de Software. En nuestro equipo, 5 personas componían al Grupo de Ingeniería (primero analistas y diseñadores) y yo tuve el rol de Administrador del Proyecto. Las profesoras fungieron como Administradoras Principales del Proyecto y a las cuales semanalmente, se les informaba los avances, problemas, riesgos y logros del desarrollo del producto.

Además, otro de los objetivos del curso que se cubría al desarrollar el proyecto, era el de trabajar la creación del producto de tal forma que cumpliera con el estándar de calidad CMM en su nivel 2 (Repetible). Una de las alumnas que no pertenecía al equipo, fungiría como Auditor Externo para avalar nuestro apego a CMM-2.

El proyecto consistió en construir el "Sistema de Inscripciones al Posgrado en Ciencia e Ingeniería de la Computación de la UNAM", usando la tecnología de Internet para permitir que la inscripción y reinscripción de los alumnos del posgrado, se efectuara por dicho medio.

Una vez terminado exitosamente el producto, pude ver que los objetivos del curso fueron bastante ambiciosos pero alcanzables. Ambiciosos porque en el mismo tiempo que se desarrollaba un producto para el cual no existía una plataforma segura y confiable en las instalaciones del cliente, ni las bases técnicas suficientes en todos los que conformábamos al equipo, teníamos que crear toda la infraestructura de una compañía de consultoría de software. Esta infraestructura incluía: la presentación del equipo como una compañía, las políticas, estándares, procedimientos, guías y medios de comunicación necesarios para trabajar correctamente en equipo y cumplir con CMM-2. Fue alcanzable porque aún con los varios riesgos menores y

mayores que en su momento enfrentamos como equipo, de la capacitación en paralelo de los roles y sus responsabilidades, del aprendizaje en paralelo de los estándares de calidad y de las pautas para el proceso de desarrollo (como el Proceso Unificado, el PSP, las Métricas, Pruebas y Verificaciones), logramos cumplir totalmente y en tiempo, los objetivos pactados con el usuario avalados por un contrato. Además, cumplimos cabalmente con CMM-2 avalados por el Revisor Externo.

En mi calidad de Administrador Principal, tuve que documentarme con respecto de las funciones de mi rol y el de todo el grupo de administradores, así como auxiliarme de bibliografía para complementar las clases teóricas (impartidas en una hora y media a la semana). Fue a raíz de dicha investigación, aunado con la experiencia real de aplicar gran parte de lo revisado en la documentación en un desarrollo tangible, lo que dio forma a este trabajo de tesis.

Como ya mencioné en el prefacio, mi inquietud principal es poder condensar, complementar y embonar las propuestas más acordes a lo que utilizamos en el desarrollo real, (y que auxiliaron en el resultado satisfactorio del producto de software), en una guía dirigida principalmente al rol de Administrador del Proyecto del desarrollo de un producto de Software. Digo principalmente porque en el afán de presentar todas las relaciones que se crean con otros roles (en forma genérica, los casos particulares están en función de cada equipo de trabajo y de cada proyecto en especial), presento también los roles de los Administradores de la Configuración, de la Calidad, el del Líder del Proyecto (que puede ser más de una persona) y el del Administrador Principal, haciendo hincapié en la forma en cómo estos roles se relacionan con el Administrador del Proyecto.

Por otro lado, durante la vida del desarrollo del producto, también tuvimos problemas y modificaciones al calendario y al contrato pactado con el usuario (acciones aceptadas desde un principio con el cliente) Parte de ellos surgieron a raíz de las condiciones propias del cliente, de la naturaleza académica del proyecto, de los integrantes del equipo, y de no contar con experiencias anteriores (de hecho, este punto es básico para poder hablar de un nivel de CMM-2 "Repetible").

Esta guía está basada entonces, en la experiencia obtenida del proyecto académico, de donde tomo los elementos que considero ayudaron a organizar, controlar y a dar seguimiento dicho proyecto, y a resolver adecuadamente y en la medida de las posibilidades, los problemas que se suscitaron. Además que se ve respaldada con elementos teóricos y prácticos (como los propuestos en el PMBOK) que surgieron de la investigación bibliográfica previa.

DESCRIPCIÓN DEL CONTENIDO DE LOS CAPÍTULOS

Este trabajo consta de dos partes. La primera engloba todos los aspectos teóricos en que se basa la guía, así como la descripción en forma particular de la estructura de la misma, está formada por la unidad 2. En la segunda parte se expone propiamente la guía, y está formada por las unidades 3, 4, 5 y 6.

El capítulo 2 "Marco Teórico" presenta los cuatro ejes teóricos principales sobre los que gira esta guía. Estos son: La Administración de Proyectos, donde se definen los conceptos más importantes referente a este tema y se particulariza a los proyectos de software; El Cuerpo de Conocimiento de la Administración de Proyectos (PMBOK), aquí se habla de la propuesta del PMI (Project Management Institute) para efectuar, siguiendo las mejores prácticas, una administración exitosa de cualquier tipo de proyecto; La Administración de Proyectos en el nivel 2 del CMM, donde se habla de este modelo de calidad y en particular de cada una de las áreas clave que conforman el nivel 2, así como de los puntos necesarios a cubrir para cumplir con dicho modelo; y Las Pautas para el Proceso de Desarrollo, donde se exponen, en forma general, los elementos básicos que surgen en el desarrollo de un producto de Software: el Proceso Estructurado, el Proceso Unificado de Desarrollo, las Métricas, las Pruebas y Validaciones. Adicionalmente, se definen los conceptos de Marcos y patrones que serán las estructuras con las que se describe esta Guía.

El capítulo 3 "Presentación de la Guía", describe en forma particular la aplicación de los elementos expuestos en el capítulo 2 para conformar la Guía que da origen a este trabajo. En primer lugar se presenta el Ciclo de Vida utilizado para esta Guía, pues es uno de los primeros pasos para poder aplicar el PMBOK; posteriormente se habla de Marcos y Patrones, puesto que la presentación de la Guía está basada en el uso de patrones de procesos agrupados en un marco (frame).

Del capítulo 4 al 6, encontramos la Guía separada por fases según el Ciclo de Vida propuesto: Inicio, Elaboración y Construcción (unidas por tener la misma naturaleza desde el punto de vista de la Administración de Proyectos) y Transición. Para cada fase, se presentan los patrones para cada rol en tres partes: una breve descripción del rol en la fase, el flujo de actividades comunes del rol usando Diagramas de Actividad¹ y una tabla que resumen los aspectos más importantes que giran alrededor del rol.

Posteriormente se presentan las conclusiones a las que condujo la realización de esta guía, así como los trabajos posteriores sugeridos. En la parte final se encuentra la bibliografía organizada por temas vistos en el trabajo, y finalmente está el anexo, que describe en forma general, los elementos que componen a un Diagrama de Actividades.

¹ Ver el Anexo A

**PRIMERA PARTE
BASE TEÓRICA**

CAPÍTULO 2. Marco Teórico

En este capítulo se exponen los elementos principales de los cuatro ejes teóricos sobre los que gira este trabajo: la Administración de Proyectos, el Cuerpo de Conocimiento de la Administración de Proyectos (PMBOK), la Administración de Proyectos en el nivel 2 del CMM y las Pautas para el Proceso de Desarrollo.

Aquí se define lo que es un proyecto, lo que es la administración, los elementos básicos de un proyecto de software, la definición de proceso, qué es el PMBOK y cuáles procesos lo conforman, cuáles son los niveles del modelo del CMM y en particular, las características obligadas del nivel 2. También se presenta, en forma general, los elementos básicos necesarios de la Ingeniería de Software, que siempre deben estar presentes en el desarrollo de un producto de software: paradigmas de desarrollo (aquí se presentan sólo dos, el estructurado y el Proceso Unificado), las métricas, las pruebas y validaciones. Finalmente se definen los conceptos de Marcos y Patrones utilizados posteriormente en la estructura de esta Guía.

2.1. Administración de Proyectos.

En esta sección, se presentan los conceptos básicos dentro de la Administración de Proyectos, de los Ciclos de Vida de un Proyecto y de los Procesos de Software. Estos puntos conforman el primer eje teórico en los que se basa este trabajo. El conocimiento de estos elementos es fundamental para aplicarlos posteriormente en la estructuración de una Guía basada en procesos. Además, conceptos como entregables, involucrados, líneas base, etc., formarán el vocabulario básico de este trabajo.

CONCEPTO DE PROYECTO

En forma general, podemos decir que un proyecto es un esfuerzo limitado por el tiempo, realizado para crear, corregir o dar mantenimiento a un producto o servicio nuevo o ya existente (PMI96). Tradicionalmente, un proyecto se define en función del producto que se desea construir, y en función de este, el proyecto podría estar compuesto por varios subproyectos, si así fuese necesario. Un proyecto está coordinado por un administrador o líder de proyecto quien tiene la encomienda de completarlo cumpliendo las expectativas de tiempo, costo y beneficio.

Los subproyectos pueden ser realizados por empresas externas o por otra unidad funcional ajena al equipo original que atiende al proyecto general.

La mayoría de los proyectos tienen en su naturaleza las siguientes características:

- Al principio, el costo y el número de personas involucradas es muy bajo, pero este número se incrementa cercano al final y desciende abruptamente muy cerca del final.
- Al principio, la probabilidad de completar exitosamente el proyecto es muy baja y el riesgo de la incertidumbre es muy alto. Generalmente esta probabilidad va creciendo conforme se avanza en el desarrollo del proyecto.
- La capacidad de influencia que tienen las personas involucradas en el proyecto es mayor al inicio del proyecto y va disminuyendo conforme se acerca a su fin.

CONCEPTO DE ADMINISTRACIÓN

La administración de proyectos es la aplicación del conocimiento, de las destrezas, de las herramientas y de las técnicas a las actividades de un proyecto, pretendiendo cumplir e inclusive, exceder las necesidades y expectativas que los usuarios tienen [PMI96]. El cumplimiento de las necesidades de los usuarios invariablemente involucra el balance entre:

- El alcance, el tiempo, el costo y la calidad.
- Los usuarios con sus diferentes necesidades y requerimientos.
- Los requerimientos identificados (necesidades) y los no identificables (expectativas).

2.1.1. FASES Y CICLO DE VIDA DE UN PROYECTO

Como cada proyecto es único, solemos dividirlos en diferentes *fases* para lograr una mejor administración. Al conjunto de fases de un proyecto se le conoce como el *ciclo de vida del proyecto* [PRE98].

El ciclo de vida de un proyecto, en forma general, define:

- El trabajo técnico que debe realizarse en cada fase.
- Los roles que se ven envueltos en cada fase.

ENTREGABLES

Cada fase está delimitada por la realización de uno o más *entregables*. Un entregable es un producto de trabajo que puede ser verificado y además es tangible. Las fases son parte de una lógica secuencial más general, diseñada para asegurar la construcción correcta del producto, objetivo final del proyecto. Al término de cada fase, se revisan tanto los entregables clave como la calidad del proyecto, principalmente con dos fines:

- a) Determinar si el proyecto puede continuar a la siguiente fase.
- b) Detectar y corregir errores en forma efectiva.

Estas revisiones de fin de fase se conocen como *"terminación de fase"*. Cada fase incluye productos de trabajo bien definidos diseñados para establecer el nivel de control deseado de la administración. La mayoría de estos productos están relacionados a los entregables principales de la fase y las fases típicamente toman sus nombres de estos productos: requerimientos, análisis, diseño, construcción, etc.

Aunque los ciclos de vida de los proyectos puedan tener fases similares, muy pocos son en realidad, iguales.

El ciclo de vida de un proyecto sirve para definir el principio y el fin del proyecto. La definición del ciclo de vida de un proyecto determinará si el estudio de posibilidad será tratado como la primera fase del proyecto o debe ser un proyecto independiente. También ayuda a especificar cuáles acciones de transición al término de un proyecto deben estar incluidas y cuáles no.

"FAST TRACKING": Generalmente, los entregables de una fase predecesora son aprovechados en el inicio de la siguiente fase. Sin embargo, algunas veces conviene iniciar las siguientes fases sin haber terminado aún la antecesora, pues el riesgo involucrado es manejable. Al traslape de fases se le conoce como *fast tracking*.

METODOLOGÍAS

Las Metodologías para la Administración de Proyectos conforman el conjunto de medios para detallar la descripción de cada fase de un proyecto, a distintos niveles según el método. Estos medios pueden ser: gráficas, diagramas, listas de verificación (checklists), reportes, etc.

INVOLUCRADOS (STAKEHOLDERS)

Los colaboradores de un proyecto son todos los individuos y organizaciones que activamente están involucrados de una u otra manera en el proyecto, o bien, aquellos cuyos intereses pueden verse afectados, positiva o negativamente, según el resultado exitoso del término del proyecto. Entre los involucrados que generalmente encontramos en todos los proyectos tenemos:

- El administrador del proyecto (*Project manager*) La persona responsable de la administración del proyecto
- Clientes Individuos u organizaciones quienes usarán el producto del proyecto.
- Organización: La empresa cuyos empleados están directamente relacionados con el trabajo del proyecto
- Patrocinadores Individuos o grupos dentro de la organización quienes proveen los recursos financieros para realizar el proyecto

Existen muchas más categorías de involucrados del proyecto, que dependerán de la naturaleza propia de cada caso. El objetivo principal de nombrar y agrupar a los involucrados, es identificar a cada individuo, grupo de individuos u organización que interviendra en el proyecto y el grado de esa intervención

LÍNEAS BASE

Una línea base es un conjunto de elementos de software generados en una fase del ciclo de vida de un proyecto, revisados y aprobados que

- Representan una base acordada para continuar con el desarrollo.
- Pueden ser una especificación o un producto revisado, que posteriormente servirán como base en el desarrollo y mantenimiento del producto de software.
- Pueden ser modificadas mediante procedimientos de control formales tales como la Administración de la Configuración.

En forma general, existen cinco tipos de líneas base [RON92].

- De la definición. Las que conforman la definición de los requerimientos del producto de software
- Del diseño Las que se originan cuando se ha determinado el cómo construir o codificar y satisfacer los requerimientos del sistema
- De código y unidades de prueba Las que se originan de la terminación de la codificación de las unidades, y éstas son probadas para verificar la satisfacción de los requerimientos del diseño, las cuales a su vez, deben satisfacer los requerimientos del software
- De prueba Las que se originan cuando las pruebas se han efectuado y se demuestra que el producto reúne los requerimientos solicitados, y está listo para entregarse al cliente o usuario
- De mantenimiento Las que se originan cuando, después de entregar el producto al cliente, el producto puede ser adaptado, modificado, corregido y mejorado.

2.1.2. EL PROCESO DE LA ADMINISTRACIÓN DE PROYECTOS

CONCEPTO DE PROCESO

Un proceso es una serie de acciones llevadas a cabo para lograr un objetivo o realizar y entregar un resultado. Un proceso puede estar formado a su vez, por una serie de procesos. Un proceso integra personas, instrumentos y procedimientos. El proceso es aquello que la gente hace usando procedimientos, métodos, instrumentos y equipo, para transformar la materia prima (entradas) en un producto (salidas) valioso para los

clientes. Dada esta definición, la Administración de Proyectos puede verse como un proceso en sí, formado por un conjunto de sub-procesos.

Un proceso define *quién, qué y cómo* trabajar para lograr un objetivo dado. En la Ingeniería de Software, el objetivo principal es construir productos de software o modificar los ya existentes. Un proceso efectivo da los lineamientos para desarrollar eficientemente software de calidad.

PROCESOS DE LA ADMINISTRACIÓN DE PROYECTOS

Los proyectos también están formados por procesos, estos son los Procesos del Proyecto (*Project Processes*), que se clasifican en dos tipos (PMI96):

- **Procesos de la administración de proyectos** (*Project management processes*), que describen y organizan el trabajo que implica realizar un proyecto. (Aquí es donde el PMBOK hace su mayor aportación) y
- **Procesos de producción** (*Product-oriented processes*), que especifican y crean propiamente el producto del proyecto. (Generalmente se definen por el ciclo de vida del proyecto, en este trabajo mostraremos dos propuestas).

Dada su naturaleza, los procesos de la Administración de Proyectos y los de Producción se traslapan durante la vida de un proyecto.

Dada la gran variedad de proyectos que existen y sus características particulares, es muy factible que no todos los procesos definidos en la Administración del Proyecto o en la Producción se lleven a cabo. Es por esto que las relaciones que existen entre los procesos mencionados anteriormente, pueden adecuarse según las necesidades de quienes las usen.

Los Procesos de la Administración de Proyectos se organizan en cinco grupos, cada uno con uno o más procesos:

- **Procesos de iniciación** (*Initiating Processes*). Reconocen que un proyecto o una fase debe comenzar y realizan las acciones para esto.
- **Procesos de planeación** (*Planning Processes*). Crean y mantienen un esquema de trabajo para cumplir con las necesidades del negocio, origen del proyecto actual. Formados por:
 - **Procesos comunes** (*Core Processes*): aquellos que son requeridos todos o casi todos y esencialmente en el mismo orden, en la mayoría de los proyectos.
 - **Procesos de facilitación** (*Facilitating Processes*): aquellos cuyas interacciones están dadas principalmente por la naturaleza del proyecto que se realiza.
- **Procesos de ejecución** (*Executing Processes*). Coordinan a los recursos humanos y materiales para cumplir el plan.
- **Procesos de control** (*Controlling Processes*). Aseguran el cumplimiento de los objetivos del proyecto monitoreando su progreso y tomando acciones correctivas en su caso.
- **Procesos de finalización** (*Closing Processes*). Formalizan en conformidad, la aceptación de la fase y posteriormente la aceptación del proyecto.

Dentro de cada grupo, sus procesos se encuentran ligados a otros por las entradas que reciben y por las salidas que entregan. En función de estas ligas, podemos describir los procesos en términos de sus:

- **Entradas**. Documentos o elementos que puedan ser documentados.
- **Técnicas**. Mecanismos efectuados sobre las entradas para generar salidas.
- **Salidas**. Documentos o elementos que resultan de los procesos.

La figura 2.1 muestra que los grupos de procesos se encuentran unidos por el resultado que producen. El proceso de planeación entrega al de ejecución el plan del proyecto inicial y las actualizaciones posteriores bajo la observación de los procesos de control. También, esta interacción entre los procesos, obliga que durante el desarrollo del proyecto, se traslapen, tal como lo muestra la figura 2.2. Además, esta interacción cruza los límites de las fases como cuando una fase de cierre entrega información a una fase inicial siguiente, como lo muestra la figura 2.3. (Aplicación del *fast tracking*).

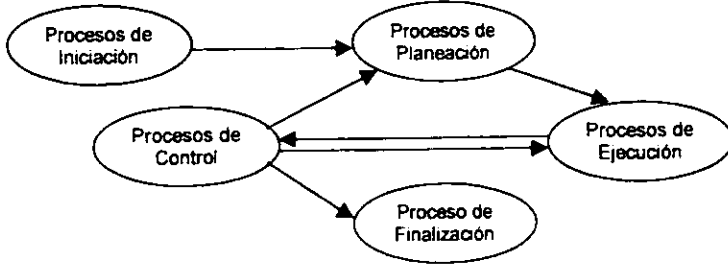


Figura 2.1 Comunicación entre los Procesos¹

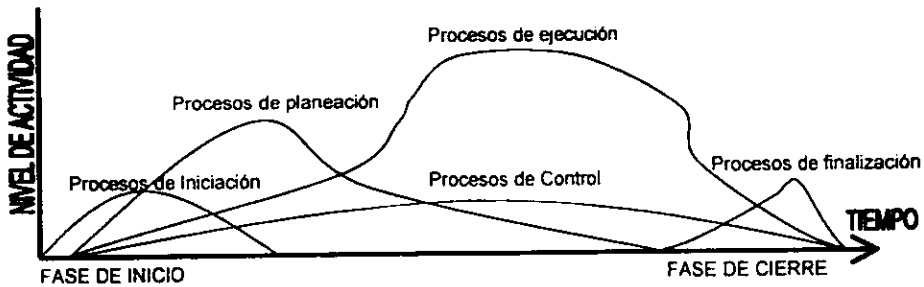


Figura 2.2 Traslape o paralelismo entre Procesos²

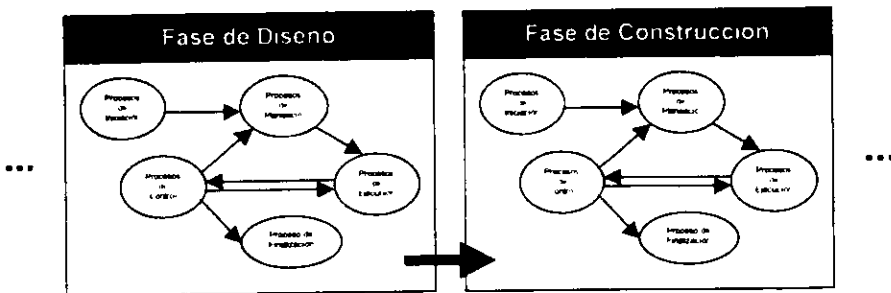


Figura 2.3 Secuencia entre Procesos³

¹ PMI, A Guide to the PMBOK, [PM196], pág. 28

² idem; pág. 29

³ idem

2.1.3. EL PROCESO DE SOFTWARE

PROCESO DE SOFTWARE

El Proceso de Software se define como el conjunto de actividades, métodos, prácticas y transformaciones que la gente emplea para desarrollar y mantener el software, así como sus productos asociados (como los planes del proyecto, documentos de diseño, código, casos de prueba, y manuales de usuario).

En [FUG95] encontramos que el Proceso de Software es un grupo de personas, estructuras de organización, reglas, políticas, actividades y procedimientos, componentes de software, metodologías y herramientas usadas o creadas específicamente para conceptualizar, desarrollar, ofrecer un servicio, innovar y extender un producto de software.

COMPONENTES DEL PROCESO DE SOFTWARE

El Proceso de Software es una composición de fases, actividades, entregables y recursos (incluyendo los humanos) [OKT98].

Las fases, mejor conocidas como fases del ciclo de vida del software, constituyen pasos significativos del Proceso de Software. Cada fase contiene varias actividades que tienen el objetivo de generar productos (entregables) importantes.

Las actividades o tareas son las piezas claves en el proceso. Definen acciones (procedimientos) que deben realizarse en un momento dado dentro del desarrollo. En general, una actividad requiere de uno o más productos de entrada y genera los entregables de salida. Una actividad también requiere recursos, particularmente humanos, que se vinculan al proceso bajo el concepto de rol. La accesibilidad de los productos de entrada y de los recursos, impone un cierto orden temporal en la ejecución de las actividades.

Los entregables, como ya se mencionó, son las entradas y las salidas de las actividades. Pueden ser un conjunto variado de documentos, como diagramas de diseño, código, planes de pruebas, manuales, calendarios, etc.

Los roles representan a los recursos más importantes en el Proceso de Software, quienes se encargan de realizar las actividades del proceso. Los roles se asignan a personas o bien, a instrumentos automáticos. Los recursos pueden ser: horas de trabajo, dinero, computadoras, etc. En la figura 2.4 se presenta el diagrama de clases con las relaciones pertinentes, propuesto en [OKT96] siguiendo la notación UML, del Proceso de Software.

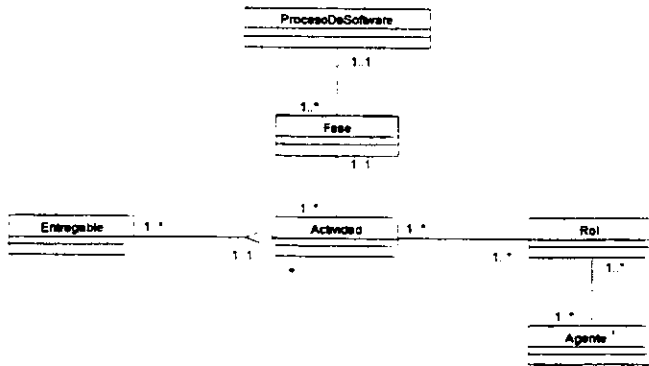


Figura 2.4 El Proceso de Software²

¹ En este diagrama, un agente es una persona o una herramienta automática que ejecuta una actividad [OKT96]
² Oktaba e Ibarjúngoitia: Software Process Modeled with Objects: Static View; [OKT98]; pág. 230.

2.2. Cuerpo de Conocimiento de la Administración de Proyectos.

En esta sección se habla del segundo eje técnico en el que se basa este trabajo. Una vez que establecimos que el desarrollo de un producto de software puede verse como un proyecto (o viceversa), podemos entonces recurrir a la experiencia de quienes llevan tiempo trabajando en el área de proyectos, y utilizar su conocimiento documentado en guías. Una de estas, es el PMBOK. En esta sección se define y se presentan sus elementos constitutivos, que posteriormente serán usados como modelo a seguir para la Guía que propone este trabajo.

DEFINICIÓN.

El Project Management Body of Knowledge (PMBOK, Cuerpo de conocimiento de la Administración de Proyectos), describe el conjunto de información requerida en el área de la Administración de Proyectos. El PMBOK incluye el conocimiento de prácticas probadas y tradicionales que ya han sido aplicadas durante mucho tiempo, así como el conocimiento de prácticas nuevas y avanzadas que comienzan a ser utilizadas.

Estas guías representan una aproximación a lo que idealmente podrían ser las mejores prácticas que hasta el momento existen. Como consecuencia de su uso, se reduce el riesgo y se incrementa la capacidad de predicción. En general, promueven una visión y una cultura común.

ÁREAS QUE LO COMPONEN.

La descripción del conocimiento aplicable a la administración de proyectos, se encuentra organizada en nueve áreas con uno o más procesos que a continuación se resumen y que se presentan en la figura 2.5:

1. Administración de la Integración del Proyecto (*Project Integration Management*). Describe los procesos requeridos para asegurar que los elementos de un proyecto, se encuentren correctamente coordinados. Está formado por:
 - 1.1. Desarrollo del plan del proyecto (*Project Plan Development*).
 - 1.2. Ejecución del plan del proyecto (*Project Plan Execution*).
 - 1.3. Control integral de cambios (*Overall Change Control*)
2. Administración del Alcance del Proyecto (*Project Scope Management*). Describe los procesos requeridos para asegurar que el proyecto incluya todo el trabajo y sólo el trabajo necesario para realizar exitosamente el proyecto. Está conformado por:
 - 2.1. Iniciación (*Initiation*).
 - 2.2. Planeación del alcance (*Scope Planning*)
 - 2.3. Definición del alcance (*Scope Definition*)
 - 2.4. Verificación del alcance (*Scope Verification*)
 - 2.5. Control de los cambios al alcance (*Scope Change Control*)
3. Administración del Calendario del Proyecto (*Project Time Management*). Describe los procesos requeridos para asegurar que el proyecto termine en los tiempos estipulados. Está conformado por:
 - 3.1. Definición de las actividades (*Activity Definition*)
 - 3.2. Establecimiento de la secuencia de las actividades (*Activity Sequencing*).
 - 3.3. Estimación de la duración de las actividades (*Activity Duration Estimating*).
 - 3.4. Desarrollo del calendario (*Schedule Development*)
 - 3.5. Control del calendario (*Schedule Control*)
4. Administración de los Costos del Proyecto (*Project Cost Management*). Describe los procesos requeridos para asegurar que el proyecto termine dentro del presupuesto aprobado. Está conformado por:
 - 4.1. Planeación de los recursos (*Resource Planning*).
 - 4.2. Estimación de los costos (*Cost Estimating*).
 - 4.3. Presupuesto de costos (*Cost Budgeting*).
 - 4.4. Control de costos (*Cost Control*).

5. Administración de la Calidad del Proyecto (*Project Quality Management*). Describe los procesos requeridos para asegurar que el proyecto satisfaga las necesidades que le dieron origen. Está conformado por:
 - 5.1. Planeación de la calidad (*Quality Planning*).
 - 5.2. Aseguramiento de la calidad (*Quality Assurance*).
 - 5.3. Control de la calidad (*Quality Control*).

6. Administración de los Recursos Humanos del Proyecto (*Project Human Resource Management*). Describe los procesos requeridos para lograr el uso más efectivo de las personas relacionadas con el proyecto. Está conformado por:
 - 6.1. Planeación de la organización (*Organization Planning*).
 - 6.2. Selección de personal (*Staff Acquisition*).
 - 6.3. Desarrollo del equipo de trabajo (*Team Development*).

7. Administración de la Comunicación en el Proyecto (*Project Communications Management*). Describe los procesos que aseguran que la información del proyecto se genere, reúna, disemine y almacene apropiadamente. Está conformado por:
 - 7.1. Planeación de la comunicación (*Communications Planning*).
 - 7.2. Distribución de la información (*Information Distribution*).
 - 7.3. Reporte de la realización o seguimiento (*Performance Reporting*).
 - 7.4. Control de la terminación administrativa (*Administrative Closure*).

8. Administración de Riesgos del Proyecto (*Project Risk Management*). Describe los procesos que identifican, analizan y trabajan los riesgos del proyecto. Formado por:
 - 8.1. Identificación de riesgos (*Risk Identification*).
 - 8.2. Medición de los riesgos (*Risk Quantification*).
 - 8.3. Desarrollo de respuestas a los riesgos (plan de contención) (*Risk Response Development*).
 - 8.4. Control de las respuestas a los riesgos (plan de contingencia) (*Risk Response Control*).

9. Administración de Abastecimiento del Proyecto (*Project Procurement Management*). Describe los procesos requeridos para adquirir bienes y servicios del exterior de la organización. Está conformado por:
 - 9.1. Planeación del abastecimiento (*Procurement Planning*).
 - 9.2. Planeación de la cotización (*Solicitation Planning*).
 - 9.3. Cotización (*Solicitation*).
 - 9.4. Selección del proveedor (*Source Selection*).
 - 9.5. Administración de contratos (*Contract Administration*).
 - 9.6. Terminación de contratos (*Contract Close-out*).

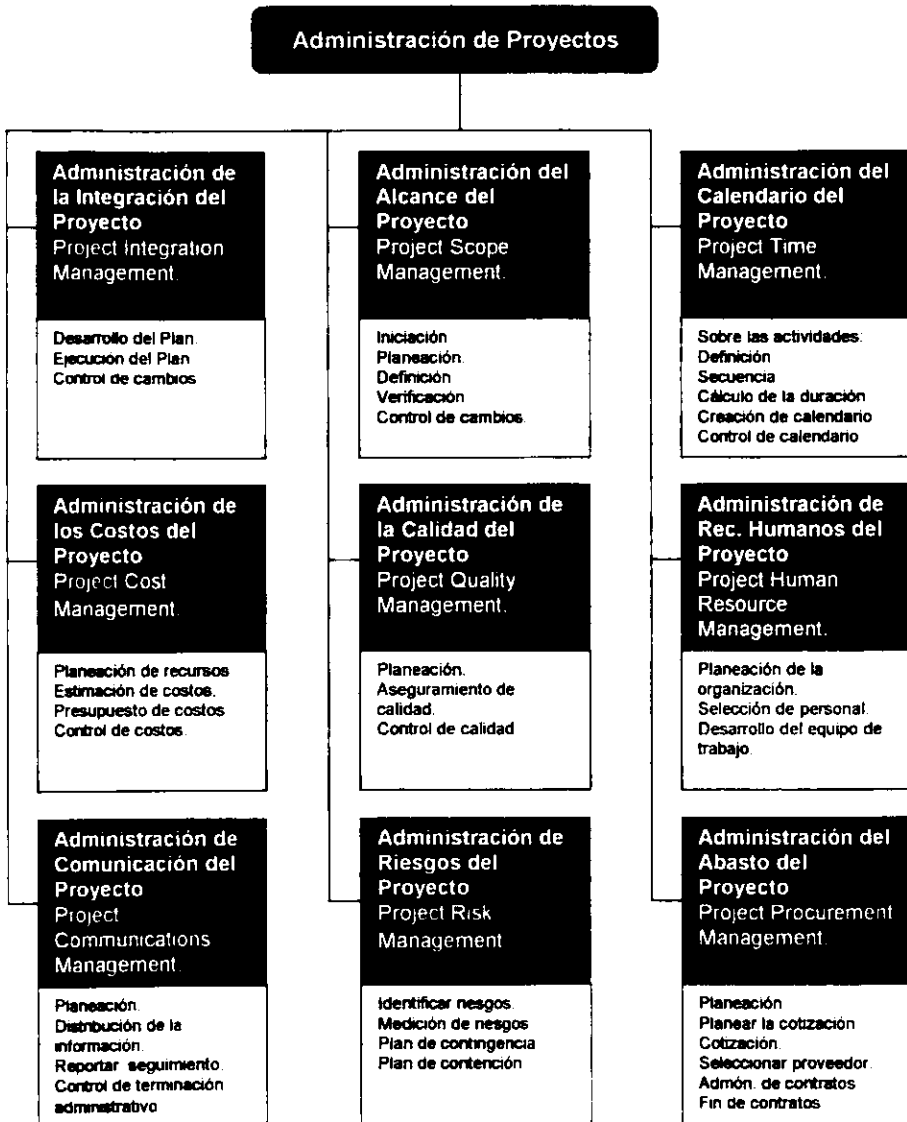


Figura 2.5 Áreas del PMBOK¹

¹ PMI. A Guide to the PMBOK: [PMI96]; pág. 7

2.2.1. INTEGRACIÓN DE LOS PROCESOS DE LA ADMINISTRACIÓN DE PROYECTOS CON LAS ÁREAS DEL PMBOK.

Como ya se mencionó en la sección anterior, las áreas del PMBOK cuentan con uno o más procesos. También vimos, en la sección 2.1.2, los procesos propios de la Administración de Proyectos. En esta sección se integran ambos conjuntos de procesos para establecer la interacción típica de la mayoría de los proyectos y de sus aplicaciones, al englobar los procesos de las áreas del PMBOK dentro de los procesos de la Administración de Proyectos.

Como ya se mencionó, los procesos se ligan unos a otros por las entradas que reciben y las salidas que entregan:

PROCESOS DE INICIACIÓN.

- **Iniciación (2.1).** Compromete a la organización a empezar la siguiente fase del proyecto.

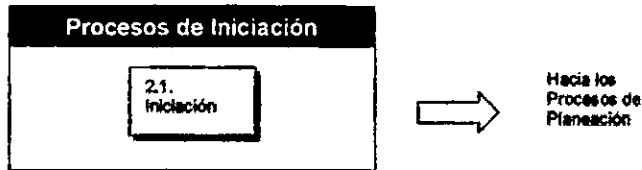


Figura 2.6 Procesos de Iniciación⁴

PROCESOS DE PLANEACIÓN.

- **Procesos comunes.**
 - **Planeación del alcance (2.2).** Desarrolla en forma escrita, el alcance como base para las decisiones futuras del proyecto.
 - **Definición del alcance (2.3).** Subdivide los entregables principales del proyecto en componentes más pequeños y manejables.
 - **Definición de las actividades (3.1).** Identifica las actividades específicas que deben realizarse para producir los diversos entregables del proyecto.
 - **Establecimiento de la secuencia de las actividades (3.2).** Identifica y documenta las dependencias interactivas entre las actividades.
 - **Desarrollo del calendario (3.3).** Estima el número de períodos de trabajo que se necesitarán para realizar actividades individuales.
 - **Planeación de los recursos (4.1).** Determina cuáles recursos (gente, equipo, materiales) y en qué cantidad de cada uno podrían usarse para realizar las actividades del proyecto.
 - **Estimación de los costos (4.2).** Desarrolla una aproximación (estimación) del costo de los recursos necesarios para realizar las actividades del proyecto.
 - **Presupuesto de costos (4.3).** Refleja el costo integral estimado a los elementos de trabajo individuales.
 - **Desarrollo del plan del proyecto (1.1).** Toma el resultado de los otros procesos de planeación y los documenta en forma organizada, consistente y coherente.
- **Procesos de Facilitación.**
 - **Planeación de la calidad (5.1).** Identifica cuáles estándares de calidad son relevantes al proyecto y determina cómo satisfacer dicho estándar.
 - **Planeación de la organización (6.1).** Identifica, documenta y asigna roles, responsabilidades y reporta sus relaciones.

⁴ PMI: A Guide to the PMBOK; [PMI96]; pág. 30

- Selección del personal (6.2). Obtiene los recursos humanos necesarios en la asignación del trabajo en el proyecto.
- Planeación de la comunicación (7.1). Determina la información y la comunicación necesaria para los involucrados: quién necesita la información, cuándo la necesitarán y cómo se les entregará dicha información.
- Identificación de riesgos (8.1). Determina cuáles riesgos podrían afectar el proyecto y documenta las características de cada uno.
- Medición de los riesgos (8.2). Evalúa los riesgos y sus interacciones para establecer el rango de los resultados posibles del proyecto
- Plan de contención (8.3). Define los pasos principales para prever oportunidades y respuestas a las amenazas
- Planeación de abastecimiento (9.1). Determina qué abastecer y cuándo.
- Planeación de la solicitud (9.2). Documenta requerimientos de productos e identifica proveedores potenciales

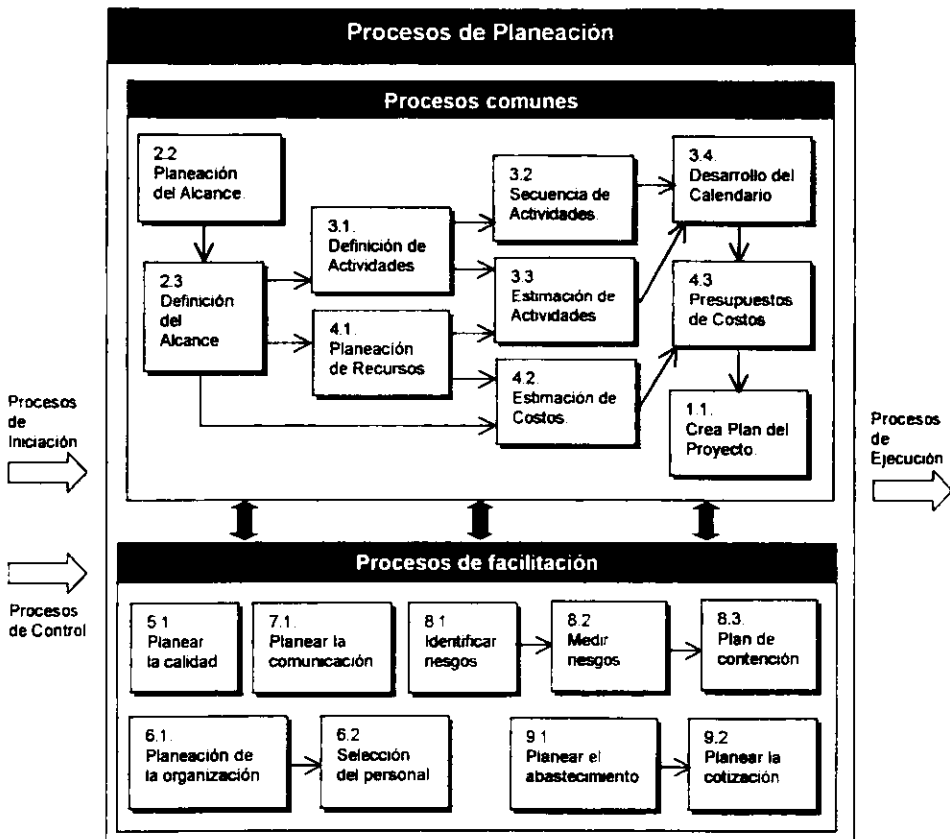


Figura 2.7 Procesos de Planeación⁵

⁵ PMI. A Guide to the PMBOK; [PMI96]; pág. 31

PROCESOS DE EJECUCIÓN.

- Ejecución del plan del proyecto (1.2). Realiza el plan del proyecto al ejecutar las actividades que lo conforman.
- Verificación del alcance (2.4). Formaliza la aceptación del alcance del proyecto.
- Aseguramiento de la calidad (5.2). Evalúa la realización del proyecto en forma integral, para dar confianza de que el proyecto cumplirá con los estándares relevantes de calidad.
- Desarrollo del equipo de trabajo (6.3). Desarrolla habilidades individuales y de grupo para mejorar la realización del proyecto.
- Distribución de la información (7.2). Permite que la información necesaria esté disponible para los involucrados durante el tiempo que se requiera.
- Cotización (9.3). Obtiene cotizaciones, ofertas, propuestas, etc.
- Selección del proveedor (9.4). Escoge de entre todos los vendedores potenciales.
- Administración de contratos (9.5). Administra las relaciones con el vendedor.

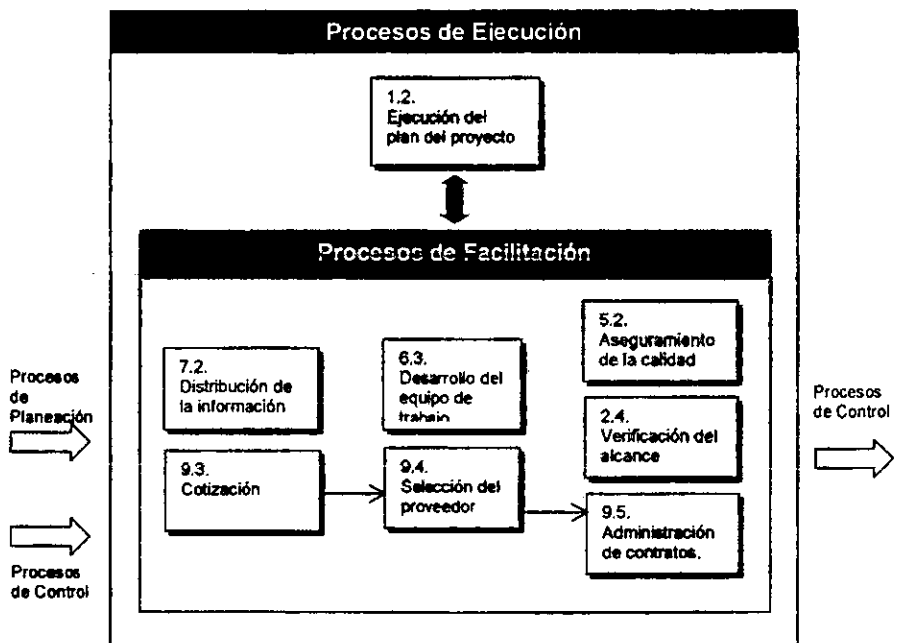


Figura 2.8 Procesos de Ejecución⁶

PROCESOS DE CONTROL.

- Control integral de cambios (1.3). Coordina los cambios en el proyecto entero.
- Control de cambios al alcance (2.5). Controla cambios al alcance del proyecto.
- Control del calendario (3.5). Controla cambios al calendario del proyecto.
- Control de costos (4.4). Controla cambios al presupuesto del proyecto.
- Control de calidad (5.3). Monitorea resultados específicos del proyecto para determinar si cumplen con los estándares de calidad relevantes e identifica medios para eliminar las causas de una realización insatisfactoria.
- Reporte del seguimiento (7.3). Colecta y disemina la información de la realización. Incluye el reporte del estado, medidas de progreso y pronósticos.

⁶ PMI: A Guide to the PMBOK; [PMI96]; pág. 33

- Plan de contingencia (8.4). Responde a los cambios dados por los riesgos durante el curso del proyecto

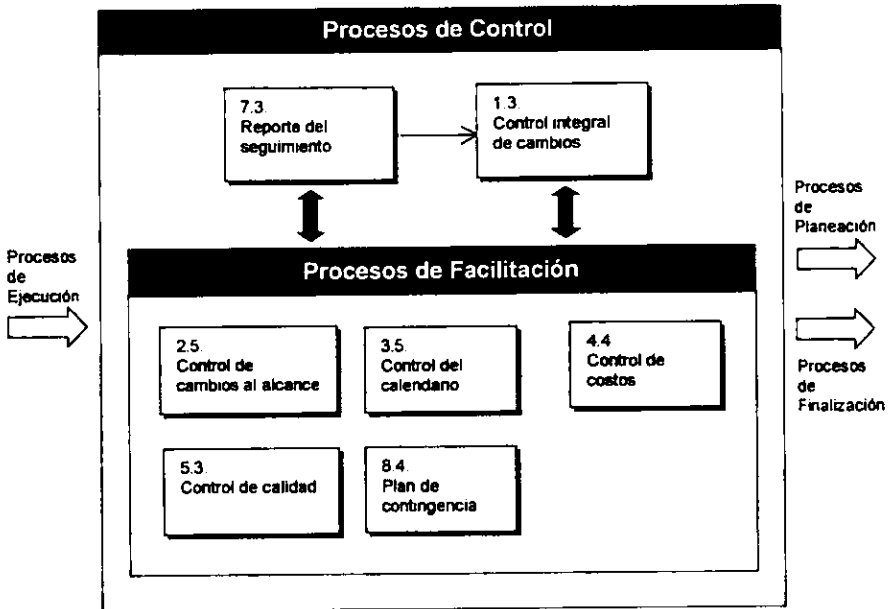


Figura 2.9 Procesos de Control⁷

PROCESOS DE FINALIZACIÓN.

- Control de la terminación administrativa (7.4). Genera, reúne y disemina la información para formalizar la terminación de una fase o del proyecto.
- Terminación de contratos (9.6). Se establece en el contrato, la terminación, así como la resolución de tópicos del proyecto aun no concluidos.

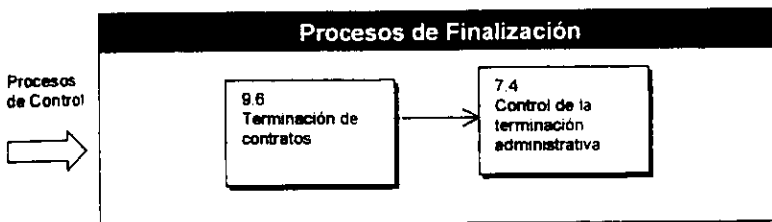


Figura 2.10 Procesos de Finalización⁸

⁷ PM; A Guide to the PMBOK, [PM196]; pág. 34

⁸ idem; pág. 35

2.2.2. ENTRADAS, TÉCNICAS Y SALIDAS DE LOS PROCESOS DEL PMBOK.

Los procesos de la Administración de Proyectos reciben entradas externas, usan las técnicas internas y producen salidas que a su vez serán entradas de otros procesos. A continuación describimos estos elementos de los procesos de las áreas del PMBOK.

1. Administración de la Integración del Proyecto.

1.1. Desarrollo del Plan del Proyecto.		
Entradas	Técnicas	Salidas
Entradas específicas del área de aplicación del proyecto. Políticas de la organización. Descripción de restricciones (2.1). Descripción de supuestos (2.1,8.1).	Metodología de planeación de proyectos. Conocimientos y habilidades de los involucrados. Sistemas de información de administración de proyectos (SIAP).	El plan del proyecto. Detalles del soporte.
1.2. Ejecución del Plan del Proyecto.		
Entradas	Técnicas	Salidas
El plan del proyecto (1.1). Detalles del soporte (1.1). Políticas de la organización (1.1). Acciones correctivas (1.3,2.5,3.5).	Habilidades de administración general. Conocimientos y habilidades del producto. Sistemas de autorización de trabajo. SIAP. Procedimientos de la organización.	Resultados del trabajo. Solicitud de cambios.
1.3. Control integral de los cambios.		
Entradas	Técnicas	Salidas
El plan del proyecto (1.1). Reportes de desempeño (7.3). Solicitud de cambios (1.2, 7.3). Sistemas de control de cambios.	Administración de la configuración. Medidas de rendimiento. Planeación adicional. SIAP.	Actualización del plan del proyecto. Acciones correctivas. Experiencia aprendida.

2. Administración del Alcance del Proyecto.

2.1. Iniciación		
Entradas	Técnicas	Salidas
Descripción del producto. Plan estratégico. Criterios de selección del proyecto. Información histórica.	Métodos de selección. Experiencia en toma de decisiones. Medidas de rendimiento. Planeación adicional. SIAP.	Descripción oficial del proyecto. Asignación del administrador del proyecto. Descripción de restricciones. Descripción de supuestos.
2.2. Planeación del Alcance.		
Entradas	Técnicas	Salidas
Descripción del producto. Descripción oficial del proyecto (2.1). Descripción de restricciones (2.1). Descripción de supuestos (2.1).	Análisis del producto. Análisis de costo / beneficio. Identificación de alternativas. Experiencia en la toma de decisiones.	Documentación del alcance. Detalles del soporte. Plan de administración del alcance.

2.3. Definición del Alcance.		
Entradas	Técnicas	Salidas
Documentación del alcance (2.2) Descripción de restricciones (2.1) Descripción de supuestos (2.1) Información histórica.	Patrones para estructurar la división del trabajo. Descomposición.	Estructura de la división del trabajo
2.4. Verificación del Alcance.		
Entradas	Técnicas	Salidas
Resultados del trabajo (1.2) Documentación del producto	Inspección Descomposición.	Aceptación Formal.
2.5. Control de los Cambios del Alcance.		
Entradas	Técnicas	Salidas
Estructura de la división del trabajo (2.3). Reportes de desempeño (7.3). Solicitud de cambios (1.2, 7.3). Plan de administración del alcance (2.2).	Sistema de control de cambios del alcance Medidas de rendimiento Planeación adicional.	Cambios al alcance Acciones correctivas Experiencia aprendida.

3. Administración del Calendario del Proyecto.

3.1. Definición de la Actividades.		
Entradas	Técnicas	Salidas
Estructura de la división del trabajo (2.3) Documentación del alcance (2.2) Información histórica Descripción de restricciones (2.1) Descripción de supuestos (2.1)	Descomposición Patrones	Lista de actividades Detalles de soporte Actualización de la estructura de la división del trabajo
3.2. Establecimiento de la Secuencia de las Actividades.		
Entradas	Técnicas	Salidas
Lista de actividades (3.1). Descripción del producto. Dependencias obligatorias. Dependencias discrecionales. Dependencias externas. Descripción de restricciones (2.1) Descripción de supuestos (2.1)	Métodos de diagramas de precedencia. Métodos de diagramas de flujos Métodos de diagramas de condiciones. Patrones de redes de conexión.	Diagrama reticular del proyecto Actualización de lista de actividades.
3.3. Estimación de la Duración de las Actividades.		
Entradas	Técnicas	Salidas
Lista de actividades (3.1). Descripción de restricciones (2.1). Descripción de supuestos (2.1) Requerimientos de recursos (4.1) Capacidades de los recursos. Información histórica.	Experiencia en la toma de decisiones. Estimación análoga. Simulación.	Estimación de la duración de las actividades. Bases de la estimación Actualización de la lista de actividades.

3.4. Desarrollo del Calendario.		
Entradas	Técnicas	Salidas
Diagrama reticular del proyecto (3.2). Estimación de la duración de las actividades (3.3). Requerimientos de recursos (4.1). Descripción del conjunto de recursos. Calendario. Descripción de restricciones (2.1). Descripción de supuestos (2.1). Avances y retrasos.	Análisis matemático. Compresión de tiempos. Simulación. Heurísticas de nivelación de recursos. Software de administración de proyectos.	Calendario del proyecto. Detalles del soporte. Plan de administración del calendario. Actualización de los requerimientos de recursos.
3.5. Control del Calendario.		
Entradas	Técnicas	Salidas
Calendario del proyecto (3.4). Reportes de rendimiento (7.3). Solicitud de cambios (1.2, 7.3). Plan de administración del calendario (3.4).	Sistema de control de cambios del calendario. Medidas de rendimiento. Planeación adicional. Software de administración de proyectos.	Actualizaciones al calendario. Acciones correctivas. Experiencia aprendida.

4. Administración de los Costos del Proyecto.

4.1. Planeación de los Recursos.		
Entradas	Técnicas	Salidas
Estructura de la división del trabajo (2.3). Documentación del alcance (2.2). Información histórica. Descripción de recursos (3.4). Políticas de la organización.	Toma de decisiones. Identificar alternativas.	Requerimiento de recursos.
4.2. Estimación de los Costos.		
Entradas	Técnicas	Salidas
Estructura de la división del trabajo (2.3). Requerimientos de recursos (4.1). Precios de recursos. Estimación de la duración de las actividades (3.3). Información histórica. Cuentas contables.	Estimaciones análogas. Modelado paramétrico. Estimación en reversa. Programas de cómputo.	Estimación de costos. Detalle del soporte. Plan de administración de costos.
4.3. Presupuesto de Costos.		
Entradas	Técnicas	Salidas
Estructura de la División del trabajo (2.3). Estimación de costos (4.2). Calendario del proyecto (3.4).	Técnicas de estimación de costos.	Línea base de costos.
4.4. Control de Costos.		
Entradas	Técnicas	Salidas
Línea base de costos (4.3). Reportes de rendimiento (7.3). Solicitudes de cambios (1.2, 7.3). Plan de administración de costos (4.2).	Sistema de control de cambios de costos. Medidas de rendimiento. Planeación adicional. Programas de SW.	Estimación de costos revisados. Presupuesto actualizado. Acciones correctivas. Estimación al término. Experiencia aprendida.

5. Administración de la Calidad del Proyecto.

5.1. Planeación de la Calidad.		
Entradas	Técnicas	Salidas
Políticas de calidad. Documentación del alcance (2.2) Descripción del producto. Estándares y regulaciones. Salidas de otros procesos.	Análisis de costo / beneficio Métodos para comparar Diagramas de flujo Diseño de experimentos	Plan de administración de la calidad. Definición de operaciones Listas de verificación. Entradas a otros procesos
5.2. Aseguramiento de la Calidad.		
Entradas	Técnicas	Salidas
Plan de administración de la calidad (5.1) Resultados de las mediciones del control de la calidad. Definición de operaciones (5.1)	Técnicas de planeación de la calidad Auditorías de calidad	Mejoramiento de la calidad.
5.3. Control de la Calidad.		
Entradas	Técnicas	Salidas
Resultado del trabajo Plan de administración de la calidad (5.1) Definición de operaciones (5.1) Listas de verificación (5.1)	Inspección. Gráficas de control. Diagramas de Pareto Estadísticas básicas. Diagramas de flujos. Análisis de tendencias.	Mejoramiento de la calidad. Decisiones de aceptación. Realizar el trabajo de nuevo Listas finales de verificación Ajustamiento de procesos

6. Administración de los Recursos Humanos del Proyecto.

6.1. Planeación de la Organización.		
Entradas	Técnicas	Salidas
Interfaces del proyecto. Requerimientos de personal. Descripción de restricciones (2.1)	Patrones. Prácticas de recursos humanos Teoría de organizaciones. Análisis de involucrados.	Asignación de roles y responsabilidades. Plan de administración del personal. Diagrama de la organización Detalle del soporte
6.2. Selección de Personal.		
Entradas	Técnicas	Salidas
Plan de administración del personal (6.1). Descripción del personal. Prácticas de reclutamiento.	Negociaciones Asignación previa Tácticas de obtención	Equipo de trabajo asignado Directorio del equipo de trabajo.
6.3. Desarrollo del Equipo de Trabajo.		
Entradas	Técnicas	Salidas
Personal del proyecto Plan de administración del personal (6.1). Reportes de rendimiento (7.3). Retroalimentación externa.	Actividades de integración Habilidades de administración general Sistemas de reconocimiento y motivación Colación Entrenamiento	Mejoramiento del rendimiento Entradas a la valoración del rendimiento.

7. Administración de la Comunicación en el Proyecto.

7.1. Planeación de la Comunicación.		
Entradas	Técnicas	Salidas
Requerimientos de comunicación. Tecnología de comunicaciones. Descripción de restricciones (2.1). Descripción de supuestos (2.1).	Análisis de involucrados.	Plan de administración de la comunicación.
7.2. Distribución de la Información.		
Entradas	Técnicas	Salidas
Resultados del trabajo. Plan de administración de la comunicación. Plan del proyecto (1.1).	Habilidades de comunicación. Sistemas de obtención de información. Sistemas de distribución de información.	Registros del proyecto.
7.3. Reporte del Seguimiento.		
Entradas	Técnicas	Salidas
Plan del proyecto (1.1). Resultados del trabajo. Otros registros de proyectos.	Revisión del seguimiento. Análisis de varianza. Análisis de tendencia. Análisis del valor ganado. Técnicas de distribución de información.	Reportes de seguimiento. Solicitud de cambios.
7.4. Control de la Terminación Administrativa.		
Entradas	Técnicas	Salidas
Documentación de mediciones del seguimiento. Documentación del producto. Registros de otros proyectos.	Técnicas y repórtes de seguimiento.	Logros del proyecto. Aceptación formal. Experiencia aprendida.

8. Administración de la Comunicación en el Proyecto.

8.1. Identificación de Riesgos.		
Entradas	Técnicas	Salidas
Descripción del producto. Otras salidas de planeación. Información histórica.	Listas de verificación. Diagramas de flujo. Entrevistas.	Origen de los riesgos. Eventos potenciales de riesgo. Síntomas de riesgos. Entradas a otros procesos.
8.2. Medición de Riesgos.		
Entradas	Técnicas	Salidas
Tolerancia a riesgos de los involucrados. Origen de los riesgos (8.1). Eventos potenciales de riesgo (8.1). Estimación de costos (4.2). Estimación de la duración de las actividades (3.3).	Valor monetario esperado. Sumas estadísticas. Simulación. Árboles de decisión. Experiencia en la toma de decisiones.	Riesgos a resolver. Riesgos a ignorar.
8.3. Plan de Contención.		
Entradas	Técnicas	Salidas
Riesgos a resolver (8.2). Riesgos a ignorar (8.2).	Técnicas de obtención. Planeación de contención. Estrategias alternativas. Seguros.	Plan de admón. de riesgos. Entradas a otros procesos. Planes de contención. Reservas. Acuerdos contractuales.

8.4. Plan de Contingencia.		
Entradas	Técnicas	Salidas
Plan de administración de riesgos (8.3). Eventos de riesgos actuales. Identificación de riesgos adicionales	Aproximaciones Desarrollo de respuestas a riesgos adicionales	Acciones correctivas Actualizaciones al plan de administración de riesgos

9. Administración del Abastecimiento del Proyecto.

9.1. Planeación del Abastecimiento.		
Entradas	Técnicas	Salidas
Documentación del alcance (2.2) Descripción del producto. Recursos de abastecimiento. Condiciones del mercado Otras salidas de planeación Descripción de restricciones (2.1) Descripción de supuestos (2.1)	Análisis hacer o comprar. Toma de decisiones. Selección del tipo de contrato.	Plan de administración del abastecimiento Requerimientos del trabajo.

9.2. Planeación de la Cotización.		
Entradas	Técnicas	Salidas
Plan de administración del abastecimiento (9.1). Requerimientos del trabajo (9.1) Otras salidas de planeación.	Formas estándares. Toma de decisiones.	Documentos de abastecimiento. Criterios de evaluación. Actualizaciones a los requerimientos del trabajo

9.3. Contratación.		
Entradas	Técnicas	Salidas
Documentos de abastecimiento (9.2). Lista de vendedores calificados.	Conferencias de postores. Anuncios y publicidad.	Cotizaciones.

9.4. Selección del Proveedor.		
Entradas	Técnicas	Salidas
Cotizaciones (9.3). Criterios de evaluación (9.2) Políticas de la organización	Negociación de contratos Sistema de toma de decisiones Estimaciones independientes	Contrato.

9.5. Administración de Contratos.		
Entradas	Técnicas	Salidas
Contrato (9.4). Resultados del trabajo Solicitud de cambios (1.2.7.3) Facturas del vendedor.	Sistema de control de cambios en los contratos. Reportes de seguimiento Sistema de pagos	Correspondencia Cambios en el contrato. Solicitud de pagos

9.6. Terminación de Contratos.		
Entradas	Técnicas	Salidas
Documentación del contrato.	Auditorías de abastecimiento	Archivo del contrato. Aceptación y terminación formal

Figura 2.11 Entradas, salidas y técnicas de los procesos del PMBOK.

2.3. La administración de proyectos en el nivel 2 del CMM.

En esta sección se presenta el tercer eje teórico, base de este trabajo. Al desarrollar un producto de software, buscamos que este satisfaga las necesidades del usuario en compromisos pactados, rapidez, economía y correctez, es decir, que cumpla con ciertos criterios de calidad. El CMM se presenta como un modelo de referencia, que puede auxiliarnos para establecer las políticas y los procedimientos que garanticen la calidad necesaria de los procesos y de los productos. En esta sección se hace un resumen de las características básicas de este modelo, detallando el nivel 2 (Repetible) que es el nivel mínimo que intentamos cumplir con esta Guía para la Administración de Proyectos.

EL CMM

El CMM [PAU95] (*Capability Maturity Model for Software*) fue desarrollado por el SEI (*Software Engineering Institute*) de la Universidad de Carnegie Mellon. Describe los elementos claves de un proceso efectivo de software. También describe el mejoramiento evolutivo de las organizaciones dedicadas al desarrollo de software, desde un nivel rudimentario hasta llegar a uno disciplinado y maduro. Esta evolución está formada por cinco niveles de madurez.

El CMM representa un camino de mejoras recomendadas, para las organizaciones de software que desean incrementar su madurez.

Junto con la madurez de la organización, el Proceso de Software va mejorando en su definición, y su puesta en marcha crece consistentemente. Un Proceso de Software efectivo integra tanto a las personas como a los instrumentos y a los métodos.

El CMM es abstracto en su esencia, pues no indica cómo debería estar implementado el Proceso de Software en una organización, sino que describe lo que normalmente se espera como resultado de este proceso y sus elementos.

LA CAPACIDAD DEL PROCESO DE SOFTWARE

La Capacidad del Proceso de Software (*Software Process Capability*), describe el rango de resultados esperados al que podemos llegar siguiendo dicho proceso. La capacidad del Proceso de Software de una organización, provee un medio de predicción de la mayoría de los entregables esperados en el siguiente proyecto de software, que dicha organización realice.

Los cinco niveles de maduración mencionados en el primer párrafo, definen una escala ordenada de medición de la madurez de la capacidad del Proceso del Software de una organización, asignando prioridades a los esfuerzos de mejoramiento.

2.3.1. ORGANIZACIONES DE SOFTWARE MADURAS E INMADURAS.

Podemos clasificar a las organizaciones dedicadas a la construcción de software, en inmaduras y maduras, dependiendo del grado de disciplina que exista en el proceso integral de desarrollo de productos de software.

ORGANIZACIONES INMADURAS

- El Proceso de Software generalmente es improvisado por practicantes y administradores durante el curso del proyecto.
- Aún si existe un Proceso de Software especificado, este no es apoyado ni se toma en cuenta rigurosamente.
- Las organizaciones son reactivas, donde los administradores generalmente están enfocados en resolver crisis inmediatas.
- Estas organizaciones generalmente sobrepasan los presupuestos y los calendarios y tiempos de entrega, puesto que no están basados en estimados reales.

- En muchos de los casos, comprometen la funcionalidad y la calidad del producto en aras de cumplir con los calendarios y tiempos cuando estos se aproximan
- No existen bases objetivas para juzgar la calidad del producto ni para resolver los problemas del proceso y aún menos los problemas del producto.
- No se toma en cuenta el cómo los pasos del Proceso de Software, afectan la calidad dificultando su predicción
- Cuando un proyecto se retrasa, generalmente se recortan o incluso se eliminan las tareas de revisión y pruebas
- Los clientes o usuarios finales tienen muy poca participación en el proceso hasta que el producto es liberado

ORGANIZACIONES MADURAS

- La organización posee la habilidad para manejar el desarrollo de software y procesos de mantenimiento.
- Puede comunicar el Proceso de Software tanto al equipo actual como a los nuevos empleados
- Realiza las tareas y actividades de acuerdo con el proceso planeado
- El proceso obligatorio está documentado, es utilizable y consistente con el trabajo que en realidad se realiza
- La definición de los procesos se actualiza cada vez que es necesario y las mejoras se realizan en forma controlada mediante pilotos de prueba o con análisis de costo – beneficio
- Existe en la organización una participación activa durante las actividades de mejora
- Los roles y las responsabilidades dentro del proceso son claros durante el proyecto y dentro de la organización
- Los administradores monitorean la calidad de los productos de software y los procesos que producen dichos productos.
- Existen bases objetivas y cuantitativas para juzgar la calidad del producto y para analizar los problemas del producto y del proceso.
- Los presupuestos, calendarios y tiempos están basados en información histórica y son realistas
- Generalmente se alcanzan los resultados esperados en costos, calendarios, tiempos, funcionalidad y calidad del producto.
- En forma general, una organización madura sigue procesos disciplinados consistentemente, puesto que cada participante entiende el valor de lo que se hace además que existe la infraestructura necesaria.

2.3.2. LOS NIVELES DE MADURACIÓN.

NIVELES DE MADURACIÓN

Un nivel de maduración es un estado bien definido, que puede evolucionar, al que se intenta llegar en un Proceso de Software maduro [PAU95]. Cada uno contiene un conjunto de objetivos que, cuando se satisfacen, estabilizan un componente importante del Proceso. A continuación se presentan estos niveles.

- | | |
|-----------------|--|
| 1. Inicial | El Proceso de Software es informal y ocasionalmente caótico, existen muy pocos procesos definidos y el éxito de un proyecto depende del esfuerzo individual de las personas. |
| 2. Repetible | Existen procesos básicos de administración de proyectos en el seguimiento de los costos, calendarios, tiempos y funcionalidad; existe la disciplina necesaria en el proceso, de tal forma que pueden repetirse los éxitos de proyectos anteriores. |
| 3. Definido | El Proceso de Software, tanto para las actividades de administración y de ingeniería, se encuentra documentado, estandarizado e integrado al Proceso de Software de la organización. Todos los proyectos utilizan la última versión aprobada del Proceso de Software de la organización en el desarrollo y mantenimiento de software |
| 4. Administrado | Se recolectan métricas importantes del Proceso y del producto de Software, además que estos procesos están cualitativamente entendidos y controlados |
| 5. Optimizado | Existe un mejoramiento continuo del Proceso, basado en la retroalimentación cuantitativa del Proceso y de nuevas ideas y tecnologías piloto |

La figura 2.12, muestra la tabla con los tipos de mejoras esperadas por cada nivel:

PROCESO				
Nivel 1	Nivel 2	Nivel 3	Nivel 4	Nivel 5
Existen o se usan pocos procesos estables.	Existen procesos documentados y estimaciones y planes estables a nivel del proyecto.	Se usan procesos integrados, administrados y de ingeniería en la organización.	Los procesos se encuentran cuantitativamente entendidos y estabilizados.	Los procesos se mejoran constante y sistemáticamente.
Se trabaja improvisadamente.	Los problemas se detectan y se corrigen al momento de ocurrir.	Se anticipan los problemas y su impacto se previene o se minimiza.	Se conoce el origen de los problemas individuales y se elimina.	Se conoce el origen común de los problemas y se elimina.
PERSONAS				
El éxito depende de los esfuerzos heroicos individuales.	El éxito depende de las personas apoyadas por el sistema de administración.	Los equipos de los proyectos trabajan juntos y a veces lo hacen como un grupo de trabajo integrado.	En cada proyecto existe un sentido fuerte de equipo de trabajo.	En toda la organización existe un sentido fuerte de equipo de trabajo.
Son comunes las modificaciones de última hora.	Los compromisos se entienden y están administrados.	El entrenamiento se planea y se imparte según el rol.		Todos están inmersos en el mejoramiento del Proceso.
No existe coordinación interdisciplinaria y en ocasiones existe rivalidad.	La gente está entrenada.			
TECNOLOGÍA				
Se corren riesgos al introducir nueva tecnología	Se establece apoyo tecnológico y existe estabilidad en las actividades.	Las nuevas tecnologías se evalúan con bases cualitativas.	Las nuevas tecnologías se evalúan con bases cualitativas.	Las nuevas tecnologías son apoyadas e implantadas proactivamente.
MÉTRICAS				
Se improvisa la colección y análisis de datos.	En los proyectos individuales se usan datos planeados y administrados.	Se recolectan los datos y se usan en todos los procesos definidos.	Se estandariza la definición y recolección de datos en la organización.	Los datos se usan para evaluar y seleccionar el mejoramiento del Proceso.
		Los datos se comparten sistemáticamente en los proyectos.	Los datos se usan para entender y estabilizar el proceso cuantitativamente.	

Figura 2.12 Mejoras esperadas por niveles del CMM⁹

⁹ Paulk, Webber et al; *The Capability Maturity Model: Guidelines for Improving the Software Process*; (PAU96); págs. 24, 25

La figura 2.13 muestra los cinco niveles de maduración del CMM. La flecha indica el tipo de Capacidad del Proceso de Software que se institucionaliza en la organización en cada nivel de madurez.

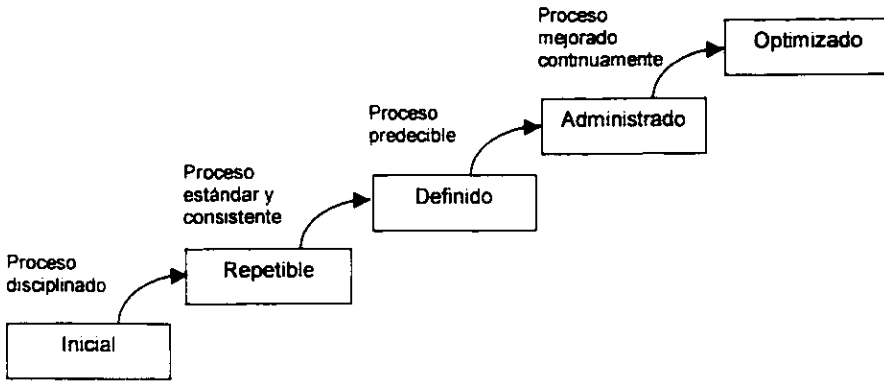


Figura 2.13 Niveles de maduración del CMM¹⁰

ÁREAS CLAVE DEL PROCESO

Las áreas claves del proceso engloban los puntos que deben cumplirse para lograr un nivel de maduración. Contempla todos los atributos esenciales que necesitan ser caracterizados en las organizaciones, para poder determinar el nivel de maduración y así, detectar aquellos elementos o sectores del Proceso de Software a los que se les debe prestar mayor atención en la búsqueda del mejoramiento. Estas áreas son:

- | | |
|------------------------|---|
| 1. Inicial | Ninguna. |
| 2. Repetible | Administración de Requerimientos.
Planeación del Proyecto de Software.
Supervisión y Seguimiento del Proyecto de Software.
Administración de Subcontratistas de Software.
Aseguramiento de la Calidad del Software.
Administración de la Configuración del Software. |
| 3. Definido | Atención a los Procesos de la Organización.
Definición de los Procesos de la Organización.
Programa de Entrenamiento
Administración de la Integración del Software.
Ingeniería del Producto de Software
Coordinación entre grupos
Revisiones entre pares. |
| 4. Administrado | Administración del Proceso Cuantitativamente
Administración de la Calidad del Software. |
| 5. Optimizado | Prevención de Errores y Defectos.
Administración del Cambio Tecnológico
Administración del Cambio del Proceso. |

¹⁰ Peuk, Webwe et al. *The Capability Maturity Model*, pág. 16.

ESQUEMA GENERAL DE LAS ÁREAS CLAVE DEL PROCESO

El CMM describe cada una de las áreas clave del proceso con el siguiente formato, de tal forma que se mantiene la consistencia en la semántica y en la estructura.

- Objetivos:** Motivaciones principales de esta área.
- Compromisos:** Acciones que la organización debe llevar a cabo para asegurar que el proceso exista y se encuentra estable.
- Condiciones:** Precondiciones y habilidades esperadas o que deben realizarse en el proyecto o en la organización para poder llevar a cabo el proceso.
- Actividades:** Actividades, roles y procedimientos necesarios para realizar el proceso.
- Métricas:** Medidas básicas necesarias para determinar el estado del proceso.
- Verificaciones:** Aseguramiento que las actividades se realizaron conforme al proceso.

2.3.3. LOS NIVELES 1 Y 2 DEL CMM.

CARACTERÍSTICAS GENERALES DEL NIVEL INICIAL

- Las organizaciones no proveen un ambiente estable para desarrollar y mantener software.
- Las organizaciones frecuentemente tienen dificultades para lograr los objetivos planteados, dando como resultado una serie de problemas mayores; objetivos que el equipo de trabajo podría lograr con un proceso de ingeniería ordenado.
- El éxito depende de la existencia de un administrador excepcional y un equipo de software apto y efectivo.
- Ocasionalmente, administradores capaces, pueden sortear presiones y tomar caminos cortos durante el proceso, pero cuando dejan el proyecto pueden desestabilizarlo.
- No se puede corregir la inestabilidad creada por la ausencia de prácticas administrativas, ni con procesos de ingeniería robustos.
- Las organizaciones generalmente desarrollan productos que cumplen con la funcionalidad, aún excediendo el presupuesto y el calendario y en contra de los procesos improvisados y caóticos de la organización.
- El éxito de las organizaciones depende de la competencia y heroísmo de la gente, y no puede repetirse a menos que las mismas características individuales se asignen al próximo proyecto.
- En este nivel, la capacidad está en función de los individuos y no de la organización.

CARACTERÍSTICAS GENERALES DEL NIVEL REPETIBLE

- Existen en la organización políticas y procedimientos para la administración del proyecto de software.
- La planeación y la administración de nuevos proyectos, están basadas en la experiencia con proyectos similares.
- Al establecer una disciplina de administración del proceso en un nivel básico, se establece y mejora la capacidad del proceso de un proyecto.
- Los proyectos implementan procesos efectivos, los cuales se encuentran definidos y documentados, además, estos procesos realmente se practican, se enseñan, se miden y se mejoran.
- Existen controles básicos de administración del software en cada proyecto.
- Los proyectos proponen objetivos realistas, basados en los resultados observados en proyectos anteriores y en los requerimientos del proyecto actual.
- Los administradores del proyecto de software llevan el seguimiento de los costos, funcionalidad, calendarios y tiempos.
- Los riesgos en lograr los objetivos se identifican al momento en que suceden.
- Los requerimientos del software y el desarrollo de los productos de trabajo que los satisfacen, constituyen las líneas base controladas.

- Se encuentran definidos los estándares del proyecto de software y las organizaciones aseguran su seguimiento fidedigno
- Existe una relación efectiva usuario – proveedor, que el proyecto de software tiene con sus subcontratistas, si los hay
- Los procesos pueden ser diferentes para proyectos distintos.
- Las organizaciones son disciplinadas puesto que son estables tanto la planeación como el seguimiento del proyecto, y pueden repetirse éxitos anteriores.
- El proceso está bajo el control efectivo de un sistema de administración de proyectos, siguiendo planes realistas basados en el desempeño de proyectos anteriores.

ÁREAS CLAVES DEL PROCESO DEL NIVEL 2: REPETIBLE

Las áreas claves del proceso en el Nivel 2 se enfocan al establecimiento de controles básicos de administración del proyecto.

Administración de Requerimientos.

- El objetivo es establecer los requerimientos que debe satisfacer el producto de software, de común acuerdo entre el cliente y el equipo de software.
- Los acuerdos logrados entre el cliente y el equipo de trabajo, formarán la base para la planeación y la administración del proyecto de software
- La comprensión de estos requerimientos forma la base para la construcción de un producto que satisfaga las expectativas del cliente
- Debe existir una revisión continua de estos requerimientos donde intervengan tanto el equipo de trabajo como el cliente.
- Los requerimientos del usuario deben estar necesariamente documentados y controlados, para que puedan formar parte básica de los próximos estimados, planes, desarrollos y seguimiento de las actividades del proyecto, durante todo el ciclo de vida del software. Para esto, hay que tomar en cuenta que dichos requerimientos evolucionan y cambian, principalmente en las primeras etapas del proyecto.

Planeación del Proyecto de Software.

- El objetivo es establecer los planes razonables para efectuar la ingeniería de software y la administración del proyecto
- Estos planes deben basarse en estimados realistas del trabajo a realizarse, y establecido las acciones necesarias para lograr este trabajo.
- Estos planes deben contener el enunciado del trabajo a realizarse, las restricciones y los éxitos que definen y limitan el proyecto de software
- El proceso de planeación del proyecto incluye la estimación del tamaño de los productos de trabajo de software, y de los recursos necesarios para producir, calendarizar, identificar y solventar riesgos del software, y para negociar la naturaleza de los objetivos
- El plan del proyecto se documenta y se le da mantenimiento pues constituye el elemento básico y necesario para la administración del proyecto de software.

Supervisión y Seguimiento del Proyecto de Software.

- El objetivo es establecer la visibilidad adecuada y en todo momento, del progreso del proyecto, esto es para que la administración pueda tomar las acciones efectivas cuando existan desviaciones significativas con respecto a lo planeado.
- La administración del proyecto de software debe estar basada en el PDS¹¹.
- La administración debe incluir el seguimiento y la revisión de los resultados logrados confrontados con el plan del proyecto. En caso necesario, se toman las acciones correctivas
- Las acciones de seguimiento pueden incluir la revisión del PDS, para reflejar los logros actuales y en su caso, hacer una nueva planeación del trabajo pendiente o bien, plantear nuevas actividades para mejorar la realización y el desempeño del proyecto.

¹¹ PDS *Plan de Desarrollo del Software*

Administración de Subcontratistas de Software.

- El objetivo es seleccionar proveedores calificados de software, y administrar el trabajo conjunto de manera efectiva.
- La selección del contratista debe basarse principalmente en sus habilidades para desarrollar el trabajo, pero también pueden contribuir otros factores como la capacidad del proceso y consideraciones técnicas.
- Conviene seleccionar subcontratistas buscando alianzas estratégicas de negocio.
- Se documenta el trabajo y los planes que realizarán los subcontratistas.
- Debe existir un seguimiento y monitoreo constante para que el trabajo de los subcontratistas se realice conforme a la planeación.

Aseguramiento de la Calidad del Software.

- El objetivo es administrar con la debida supervisión, los procesos que se usan en el proyecto de software y los productos construidos.
- La supervisión se logra con la revisión y auditoría de las actividades y de los productos de software, para verificar que cumplen con los estándares y procedimientos adecuados.
- Se revisan en primer lugar, los elementos acordados, los cuales se corrigen si así es necesario.
- El Grupo de Aseguramiento de la Calidad del Software, deberá escalar apropiadamente la solución de los elementos que aún no han sido corregidos dentro del proyecto de software.

Administración de la Configuración del Software.

- El objetivo es establecer y mantener la integridad de los productos del proyecto de software durante su ciclo de vida.
- La integridad de los productos de trabajo se logra con la identificación de la configuración del software (es decir, de los productos de trabajo del software y de sus descripciones), en un momento dado, sistemáticamente controlando los cambios de la configuración y manteniendo la integridad y el seguimiento de la configuración durante el ciclo de vida del software.
- Mientras se desarrollan las líneas base del software, éstas se encontrarán en sus respectivas bibliotecas.
- Los cambios y las liberaciones de las líneas base contenidas en sus respectivas bibliotecas, están sistemáticamente controladas vía un sistema de control de cambios y de auditorías de la configuración, funciones que lleva a cabo la Administración de Configuración del Software.

2.3.4. ESQUEMA PARA EL NIVEL 2 DEL CMM, REPETIBLE.

ÁREA CLAVE: ADMINISTRACIÓN DE REQUERIMIENTOS.

Objetivos:

1. Controlar los requerimientos del sistema para establecer las líneas base de IS y de uso administrativo.
2. Mantener la consistencia entre los planes del software, los productos y las actividades, con los requerimientos del sistema.

Compromisos:

1. El proyecto seguirá una política escrita y propia de la organización, para administrar los requerimientos del sistema.

Condiciones:

1. Para cada proyecto, se establecerán las responsabilidades para analizar los requerimientos del sistema, y estos últimos se asignarán a componentes de hardware, software u otros.
2. Los requerimientos se deben documentar.
3. Deben existir recursos y fondos necesarios para administrar los requerimientos.
4. Los elementos del GIS¹² y otros grupos relacionados, deberán estar debidamente capacitados para realizar las actividades de la administración de requerimientos.

¹² GIS Grupo de Ingeniería de Software.

Actividades

- 1 Los requerimientos serán revisados por el GIS antes de que sean incorporados al PYS¹³
- 2 El GIS usará los requerimientos como base para los planes, productos de trabajo y actividades.
- 3 Deben revisarse e incorporarse al proyecto, todos los cambios que se hayan hecho a los requerimientos

Métricas

- 1 Deben tomarse medidas que determinen el estado de las actividades de administración de requerimientos

Verificaciones

1. Revisar periódicamente, las actividades de administración de requerimientos junto con el Administrador Principal
- 2 Revisar periódicamente o según los eventos que se presenten, las actividades de administración de requerimientos junto con el Administrador del Proyecto
- 3 El GAQS¹⁴ revisará o auditará y reportará, las actividades y los productos de trabajo de la administración de los requerimientos.

ÁREA CLAVE: PLANEACIÓN DEL PROYECTO DE SOFTWARE.

Objetivos:

- 1 Documentar los estimados del software para usarlos en la planeación y seguimiento del PYS.
- 2 Planear y documentar las actividades y obligaciones del PYS
- 3 Acordar con los grupos e individuos afectados, los compromisos relacionados con el PYS

Compromisos:

- 1 Designar un Administrador del Proyecto de Software, responsable de negociar los compromisos y de crear el plan de desarrollo del PYS.
- 2 Seguir una política escrita de la organización para planear el PYS.

Condiciones:

1. Debe existir un documento aprobado con la planeación del PYS.
2. Asignar las responsabilidades para realizar el PDS.
- 3 Deben existir recursos y fondos adecuados para planear el PYS.
4. Los administradores, IS¹⁵ y otros individuos relacionados con la planeación del PYS, deberán estar debidamente entrenados para estimar y planear las actividades de las áreas bajo su responsabilidad

Actividades:

- 1 Integrar al GIS en el equipo del proyecto
- 2 La planeación del PYS debe iniciar en las primeras etapas y en paralelo con la planeación total del proyecto
- 3 Integrar al GIS para que participe con otros grupos involucrados en la planeación, durante toda la vida del proyecto
- 4 La administración principal revisará, según los procedimientos documentados, los compromisos hechos con individuos y grupos externos a la organización.
- 5 Definir o identificar el ciclo de vida del software con sus etapas predefinidas y manejables
- 6 Crear el plan de desarrollo del PYS, conforme a los procedimientos documentados
7. Documentar el plan del PYS.
- 8 Identificar los productos de trabajo de software, necesarios para establecer y controlar el PYS
- 9 Seguir los procedimientos documentados para estimar el tamaño de los productos de software o los cambios a estos tamaños.
- 10 Seguir los procedimientos documentados para estimar los costos y esfuerzos del PYS.

¹³ PYS Proyecto de Software

¹⁴ GAQS Grupo de Aseguramiento de la Calidad del Software.

¹⁵ IS Ingenieros de Software

11. Seguir los procedimientos documentados para estimar los recursos de cómputo críticos del proyecto.
12. Seguir los procedimientos documentados para crear el calendario del PYS.
13. Identificar, medir y documentar los riesgos del software asociados con los costos, recursos, calendario y aspectos técnicos del proyecto.
14. Preparar los instrumentos de apoyo y de ayuda, para la IS del proyecto.
15. Registrar los datos de la planeación del software.

Métricas:

1. Deben tomarse medidas que determinen el estado de las actividades de la planeación del software.

Verificaciones:

1. Revisar periódicamente, las actividades de la planeación del PYS, junto con el Administrador Principal.
2. Revisar periódicamente o según los eventos que se presenten, las actividades de la planeación del PYS, junto con el Administrador del Proyecto.
3. El GAQS revisará o auditará y reportará, las actividades y los productos de trabajo de la planeación del PYS.

ÁREA CLAVE: SUPERVISIÓN Y SEGUIMIENTO DEL PROYECTO DE SOFTWARE.

Objetivos:

1. Corroborar que los resultados actuales y el desarrollo de las actividades siguen los planes del proyecto.
2. Tomar y administrar hasta el final, las acciones correctivas, cuando los resultados actuales se desvían significativamente de los planes.
3. Acordar con los individuos y grupos afectados, los cambios a los compromisos del PYS.

Compromisos:

1. Designar un Administrador del Proyecto de Software responsable de negociar las actividades y resultados del PYS.
2. Seguir una política escrita de la organización, para la administración del PYS.

Condiciones:

1. Documentar el PDS aprobado, del PYS.
2. El administrador del PYS debe asignar explícitamente, las responsabilidades sobre las actividades y productos de trabajo de software.
3. Deben existir recursos y fondos adecuados para planear el seguimiento del PYS.
4. Los administradores del software deberán estar debidamente entrenados, personal y técnicamente, en los aspectos del PYS.
5. Los administradores de primera línea, deberán recibir orientación en los aspectos técnicos del PYS.

Actividades:

1. Usar el PDS documentado, para dar seguimiento y comunicar el estado de las actividades de software.
2. Revisar, según los procedimientos documentados, el PDS del proyecto.
3. La administración principal revisará, según los procedimientos documentados, los compromisos del PYS y los cambios que se hayan acordado con individuos y grupos externos a la organización.
4. Comunicar a los miembros del GIS y a otros grupos relacionados. La aprobación de los cambios a los compromisos que afectan al proyecto.
5. Seguir y tomar las acciones correctivas, si así fuese necesario, sobre los tamaños de los productos de software o sobre los cambios de estos tamaños.
6. Seguir y tomar las acciones correctivas, si así fuese necesario, sobre los costos y esfuerzos del PYS.
7. Seguir y tomar las acciones correctivas, si así fuese necesario, sobre los recursos de cómputo críticos para el proyecto.
8. Seguir y tomar las acciones correctivas, si así fuese necesario, sobre el calendario del PYS.

- 9 Seguir y tomar las acciones correctivas, si así fuese necesario, sobre las actividades técnicas de ingeniería de software.
- 10 Seguir los riesgos asociados con los costos, recursos, calendarios y aspectos técnicos
- 11 Registrar métricas y datos de los cambios a la planeación.
- 12 El GIS debe realizar periódicamente, revisiones internas para dar seguimiento al progreso técnico, a los planes, realización y contratiempos del PDS.
- 13 Revisar formalmente que los resultados hasta el momento, cumplen con lo fijado en el PYS, según los procedimientos documentados

Métricas:

- 1 Deben tomarse medidas que determinen el estado del seguimiento y de la supervisión de las actividades.

Verificaciones:

1. Revisar periódicamente, las actividades de seguimiento y supervisión del PYS junto con el Administrador Principal.
2. Revisar periódicamente o según los eventos que se presenten, las actividades de seguimiento y supervisión del PYS junto con el Administrador del Proyecto
3. El GAQS revisará o auditará y reportará, las actividades y los productos de trabajo del seguimiento y supervisión del PYS.

ÁREA CLAVE: ADMINISTRACIÓN DE SUBCONTRATISTAS DE SOFTWARE.

Objetivos:

1. Seleccionar subcontratistas calificados en el área de software.
2. Acordar, el contratista y el subcontratista, los compromisos a realizar.
3. Mantener la comunicación entre el contratista y el subcontratista.
4. Supervisar, conforme a los compromisos acordados, la realización y los resultados actuales que entregue el subcontratista.

Compromisos:

1. Seguir una política escrita de la organización, para la administración de subcontratistas
2. Designar un Administrador de Subcontratistas, responsable de establecer y administrar a los subcontratistas de software.

Condiciones:

1. Deben existir recursos y fondos adecuados para seleccionar y administrar subcontratistas.
2. Los administradores del software, y otras personas involucradas en el establecimiento y administración de los subcontratistas, deberán estar debidamente entrenadas para realizar estas actividades.
3. Los administradores del software, y otras personas involucradas en el establecimiento y administración de los subcontratistas, deberán recibir orientación en los aspectos técnicos de la contratación.

Actividades

- 1 Definir y planear el trabajo de los subcontratistas, según los procedimientos documentados
- 2 Seguir los procedimientos documentados de selección del contratista, según la evaluación de su capacidad para realizar el trabajo
- 3 Usar el contrato firmado entre el contratista y el subcontratista, como base para la administración de este último.
- 4 Revisar el plan de desarrollo de software del subcontratista, documentado y aprobado por el contratista
- 5 Usar el plan de desarrollo de software del subcontratista, documentado y aprobado por el contratista, para supervisar las actividades y comunicar su estado
- 6 Seguir los procedimientos documentados para realizar los cambios a los compromisos acordados en el contrato con el subcontratista, así como sus términos y condiciones.
7. El administrador del proyecto revisará periódicamente, el estado y la coordinación con el administrador de subcontratistas.
8. Revisar y retroalimentar periódicamente con el subcontratista.
9. Revisar formalmente que los resultados del subcontratista hasta el momento, cumplen con lo fijado en el PYS según los procedimientos documentados.

10. El GAQS revisará las actividades de aseguramiento de calidad del software del subcontratista, según los procedimientos documentados.
11. El GACS¹⁶ revisará las actividades de administración de configuración del subcontratista, según los procedimientos documentados.
12. Realizar pruebas de aceptación como parte de los entregables de los productos de software del subcontratista, según los procedimientos documentados.
13. Evaluar periódicamente, el rendimiento del subcontratista.

Métricas:

1. Deben tomarse medidas que determinen el estado de las actividades de la administración de los subcontratistas.

Verificaciones:

1. Revisar periódicamente, las actividades de administración de subcontratistas junto con el Administrador Principal.
2. Revisar periódicamente o según los eventos que se presenten, las actividades de administración de subcontratistas junto con el Administrador del Proyecto.
3. El GAQS revisará o auditará y reportará las actividades y los productos de trabajo de la administración de subcontratistas.

ÁREA CLAVE: ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE.

Objetivos:

1. Planear las actividades de aseguramiento de calidad del software (AQS).
2. Verificar objetivamente, que las actividades y los productos de software sigan los requerimientos, estándares y procedimientos.
3. Informar las actividades y resultados de las acciones tomadas en el aseguramiento de calidad.
4. El administrador general resolverá los conflictos del PYS que no hayan sido corregidos.

Compromisos:

1. Seguir una política escrita de la organización para la implantación del AQS¹⁷.

Condiciones:

1. Debe existir un grupo responsable de coordinar e implantar el AQS en el proyecto.
2. Deben existir recursos y fondos adecuados para realizar las actividades del AQS.
3. Los miembros del GAQS deberán estar debidamente entrenados para realizar las actividades del AQS.
4. Los miembros del PYS deberán estar debidamente orientados en sus roles, responsabilidades, autoridad e importancia en el GAQS.

Actividades:

1. Prepara un plan de AQS (PAQS) para el PYS, según los procedimientos documentados.
2. El GAQS debe realizar sus actividades según el PAQS¹⁸.
3. El GAQS debe participar en la preparación y revisión del plan de desarrollo del PYS, de los estándares y de los procedimientos.
4. El GAQS debe revisar que se realicen las actividades de IS.
5. El GAQS debe auditar los productos de trabajo del software diseñado.
6. El GAQS debe reportar periódicamente los resultados de sus actividades al GIS.
7. Documentar y manejar, según los procedimientos escritos, las desviaciones identificadas en las actividades y en los productos de trabajo de software.
8. El GAQS debe realizar periódicamente, revisiones de sus auditorías y resultados, con el personal de AQS del cliente.

Métricas:

1. Deben tomarse medidas que determinen el costo y estado del calendario de las actividades de AQS.

¹⁶ GACS Grupo de Administración de la Configuración del Software.

¹⁷ AQS Aseguramiento de la Calidad del Software.

¹⁸ PAQS Plan de Aseguramiento de la Calidad del Software.

Verificaciones:

1. Revisar periódicamente, las actividades de AQS junto con el Administrador Principal
2. Revisar periódicamente o según los eventos que se presenten, las actividades de AQS junto con el Administrador del Proyecto
3. Periódicamente, un grupo independiente de expertos en AQS revisará las actividades y los productos de trabajo del GAQS del proyecto

ÁREA CLAVE: ADMINISTRACIÓN DE LA CONFIGURACIÓN DEL SOFTWARE.

Objetivos:

1. Planear las actividades de la administración de la configuración del software (ACS)
2. Identificar, controlar, seleccionar y ofrecer los productos de software
3. Controlar los cambios identificados a los productos de trabajo de software
4. Informar a los grupos e individuos involucrados, del estado y contenido de las líneas base del software

Compromisos:

1. Seguir una política escrita de la organización para la implantación de I ACS¹⁹

Condiciones:

1. Establecer un Plan de la Configuración del Software, que contenga las autorizaciones para la administración de las líneas base del proyecto
2. Debe existir un grupo responsable de coordinar e implantar la ACS en el proyecto
3. Deben existir recursos y fondos adecuados para realizar las actividades de la ACS.
4. Los miembros del GACS deberán estar debidamente entrenados en los objetivos, procedimientos y métodos para realizar las actividades del ACS.
5. Los miembros del GIS y otros grupos relacionados, deberán estar debidamente entrenados para realizar sus actividades de ACS.

Actividades:

1. Preparar un plan de ACS (PACS) para cada PYS, según los procedimientos documentados.
2. Usar el PACS²⁰ documentado y aprobado, para realizar las actividades de ACS.
3. Establecer un sistema de Biblioteca de la Configuración del Software, que funcione como repositorio para las líneas base del software
4. Identificar los productos de trabajo del software bajo la responsabilidad del GAC.
5. Supervisar el inicio, registro, revisión, aprobación y seguimiento, de las solicitudes de cambios y de los problemas reportados en todos los elementos o unidades de la configuración, según los procedimientos documentados.
6. Controlar, según los procedimientos documentados, los cambios a las líneas base
7. Crear los productos de las bibliotecas de las líneas base de software, y controlar sus versiones, según los procedimientos documentados
8. Registrar el estado de los elementos y unidades de la configuración, según los procedimientos documentados
9. Generar y proveer los reportes estándares que documentan las actividades de la ACS y el contenido de las líneas base del software
10. Realizar auditorías a las líneas base según los procedimientos documentados

Métricas:

1. Deben tomarse medidas que determinen el estado de las actividades de la ACS.

Verificaciones:

1. Revisar periódicamente, las actividades de la ACS junto con el Administrador Principal.
2. Revisar periódicamente o según los eventos que se presenten, las actividades de la ACS junto con el Administrador del Proyecto
3. El GACS debe auditar periódicamente las líneas base para verificar que sean consistentes con la documentación.
4. El GAQS revisará o auditará y reportará las actividades y los productos de trabajo de la ACS.

¹⁹ ACS Administración de la Configuración del Software

²⁰ PACS Plan de la Configuración del Software

2.4. Pautas para el proceso de desarrollo.

Un aspecto básico en la administración de un proyecto de software, es delimitar y especificar la estructura técnica en la cual se basará el trabajo de Ingeniería de Software. Esto es fundamental puesto como lo indica [PRE97], marca el inicio de una serie de tareas de modelado que llevan a una especificación completa de los requisitos y a una representación del diseño general del producto de software a construir. Actualmente existen, entre otras, dos tendencias que dominan el modelado, el paradigma estructurado y el orientado a objetos.

En esta sección se presentan estas propuestas, así como elementos fundamentales en el desarrollo de un producto de software: las métricas, evaluaciones y pruebas; conformando así el cuarto eje teórico de este trabajo de tesis. Adicionalmente se definen los conceptos de marcos y patrones, pues con estos hará la descripción de cada uno de los elementos que abarca esta Guía para la Administración de Proyectos.

EL PARADIGMA ESTRUCTURADO.

El paradigma estructurado es la propuesta donde creamos modelos que representan el contenido y el flujo de la información (datos y control); dividimos el sistema dependiendo de su función y su comportamiento y con esto, representamos la esencia del producto que queremos construir. Este paradigma no es un método sencillo que se aplica siempre de la misma forma por todos los que lo usan, más bien es un amalgama que ha evolucionado durante los últimos veinte años [PRES97], [PRE98].

Dentro del paradigma estructurado, los datos se encuentran separados de los procesos que los transforman. El comportamiento dinámico del sistema, aunque importante, generalmente queda fuera del análisis estructurado, pues principalmente este análisis se basa en la descomposición funcional estática.

La figura 2.14 muestra las técnicas de modelado más importantes usadas dentro del paradigma estructurado y la relación que guardan entre ellas. Dependiendo de cada metodología en particular, se utilizarán estos modelos para representar una parte del mundo real con su respectivo contexto dentro del negocio [BAR90].



Figura 2.14 Técnicas de modelado principales en un paradigma estructurado²¹.

²¹ Barker; CASE Method: tasks and deliverables; [BAR90]; pág. 1-5.

EL PARADIGMA ORIENTADO A OBJETOS.

Como su nombre lo indica, el principal componente de este paradigma y el que lo caracteriza es el de objeto. Un objeto es todo aquello que en forma única, puede distinguirse dentro del universo de una aplicación. Los objetos tienen propiedades que definen sus características propias; para manipularlos, los objetos contienen mecanismos, métodos o funciones que describen su comportamiento, formado estos, el único medio factible para afectar sus propiedades y estados. La generalización de un objeto es la clase.

Una clase es el medio para caracterizar en forma precisa, las propiedades de la estructura y del comportamiento que una colección de entidades (objetos) comparte entre sí [BOO91]. Las características básicas y principales del paradigma orientado a objetos son:

Abstracción

- Es un modelo que sobresalta los aspectos relevantes de un fenómeno e ignora sus detalles irrelevantes [GHE97].
- Denota las características esenciales de un objeto, que permiten distinguirlo de todas las demás clases de objetos y con esto, fija las fronteras conceptuales definidas en relación con la perspectiva de quien lo observa [BOO91].
- Es la forma en cómo tratamos los objetos en el mundo real según la perspectiva que tomemos [KAP95].
- Surge de un reconocimiento de las similitudes entre ciertos objetos, situaciones o procesos del mundo real, y la decisión de concentrarse en esas similitudes e ignorar por el momento las diferencias [DAH72].
- Es una descripción simplificada o especificación de un sistema que enfatiza algunos de los detalles o propiedades del mismo mientras suprime otros [SHA84].
- Un concepto merece el calificativo de abstracción sólo si se puede describir, comprender y analizar independientemente del mecanismo que vaya a utilizarse eventualmente para realizarlo [BER86].

Encapsulación

- Es el mecanismo que permite agrupar los componentes de un programa que combinados proveen un servicio, y permitir la visibilidad a quien desee usar este servicio de sólo sus aspectos relevantes [GHE97].
- Es el proceso donde se esconden todos los detalles de un objeto que no contribuyen a sus características principales [BOO91].
- Es la característica de los objetos de incluir toda su información en una entidad única, lista a ser invocada cada vez que sea requerida [KAP95].
- Es el proceso de almacenar en un mismo compartimiento los elementos de una abstracción que constituyen su estructura y su comportamiento. [BOO96].
- Permite que ninguna parte de un sistema complejo dependa de los detalles internos de otras partes [IGA76].

Herencia²²

- Es la estructuración por categorías u ordenación de las abstracciones [BOO91].
- Es la habilidad de los objetos para tomar toda la funcionalidad y características de otros objetos, según una estructura previa [KAP95].
- Define una relación entre clases, en la que una clase comparte la estructura de comportamiento definida en una o más clases. Una subclase aumenta o redefine la estructura y el comportamiento de sus superclases.
- Semánticamente, la herencia denota una relación "es un".

Polimorfismo

- Mecanismo que permite que un objeto pueda pertenecer a más de un tipo. El tipo de un objeto dependerá de la asociación que dinámicamente se presente [GHE97].
- Es la habilidad de los objetos de la misma clase, de comportarse en forma diferente dependiendo las diversas situaciones en las que se encuentre [KAP91].

²² También llamada Generalización [CAN98]. En [BOO96] este concepto lo maneja bajo el nombre de Jerarquía, y dice que la Herencia es la Jerarquía "de clases" más importante.

Al igual que para el paradigma estructurado, para el orientado a objetos existe una gran variedad de métodos. Cada uno de ellos introduce un proceso para el análisis de un producto o sistema, un conjunto de modelos que evoluciona fuera del proceso, y una notación que posibilita la consistencia de los modelos. Entre los más importantes cronológicamente, tenemos el Método de Coad y Yourdon, el Método de Booch, el Método de Jacobson y el Método de Rumbaugh. Finalmente, estos tres últimos autores fusionaron en un solo método sus propias propuestas, creando el Proceso Unificado que se explica brevemente en las secciones siguientes.

A continuación, describiremos brevemente una metodología estándar estructurada así como el Proceso Unificado basado en el paradigma de la orientación a objetos, con el fin de preparar la guía, razón de este trabajo, de tal manera que pueda seguir cualquiera de ambos paradigmas.

2.4.1. EL DESARROLLO ESTRUCTURADO.

Este proceso conjunta las principales propiedades de la Ingeniería de Software propuestas por Yourdon principalmente, y cubren totalmente el ciclo de vida de un sistema. Está dividido en diferentes fases, cada una con un conjunto de tareas y productos específicos que llevados a cabo, garantizan un mejor entendimiento y desarrollo del sistema. El proceso une cada fase de tal manera que los resultados de una fase serán la información de entrada de las siguientes [GIL98].

A lo largo de la evolución del paradigma estructurado, han existido diversas propuestas para aplicar sus conceptos, dando lugar a un gran número de metodologías. Como ya se mencionó, todas estas tienen una estructura común (figura 2.15) y un ciclo de vida semejante. Las diferencias recaen, principalmente, en cada una de las actividades que definen a las fases del Proceso de Software, los entregables y los roles en particular. Por citar sólo algunos de los más importantes están: la Metodología de Yourdon, la Metodología SSADM, la Metodología Métrica V2, la Metodología Merise y la Case*Method [GIL98].

Podemos generalizar, entonces, sus características principales:

- Orientado fundamentalmente a las funciones.
- Basado en la funcionalidad del producto, utiliza técnicas de diseño descendentes (top-down) o ascendentes (bottom-up) para fragmentar y disminuir la complejidad funcional.
- La información es independiente de la funcionalidad.
- El medio más utilizado para relacionar la información con la funcionalidad es el uso de los Diagramas de Flujos de Datos.
- No existe actualmente, un estándar único para la nomenclatura utilizada en los modelos y diagramas.

Las fases más comunes dentro del desarrollo estructurado son [GIL98]:

- Estrategia (*Strategy*). Donde se produce en conjunto con el usuario, el modelo del negocio, un conjunto de recomendaciones y un plan de desarrollo del sistema, cubriendo los requerimientos actuales y futuros del negocio, tomando en cuenta las restricciones organizacionales, financieras y técnicas.
- Análisis (*Analysis*). Donde se verifican los resultados de la estrategia, detallándolos y refinando a un nivel suficiente para especificar los requerimientos de la organización, planteando modelos que reflejen adecuadamente la situación de la organización y desarrollando alternativas válidas de solución.
- Diseño (*Design*). Donde se toman y se verifican los resultados de la etapa de análisis y se encuentra el mejor camino para cubrir los requerimientos, dado el ambiente técnico y los niveles requeridos de automatización. El objetivo es producir un diseño apropiado a las necesidades del negocio, dentro de las restricciones técnicas y considerando el enorme impacto que tiene dicho diseño, en el rendimiento final del sistema.
- Construcción (*Build*). Donde se codifica y prueban los programas utilizando las herramientas adecuadas al problema, escogidas éstas dependiendo del ambiente técnico y del tipo de programa.
- Documentación (*Documentation*). Donde se crean los manuales del usuario y la documentación de cada fase que ha quedado registrada en documentos, reportes y diagramas

- Transición (*Transition*). Donde se realizan todas las tareas necesarias para la instalación del sistema en su ambiente de operación, proporcionando un periodo inicial de soporte y apoyo
- Producción (*Production*). Donde se asegura que el sistema funciona adecuadamente, con la mínima intervención del personal de operación y soporte, se monitorea su rendimiento

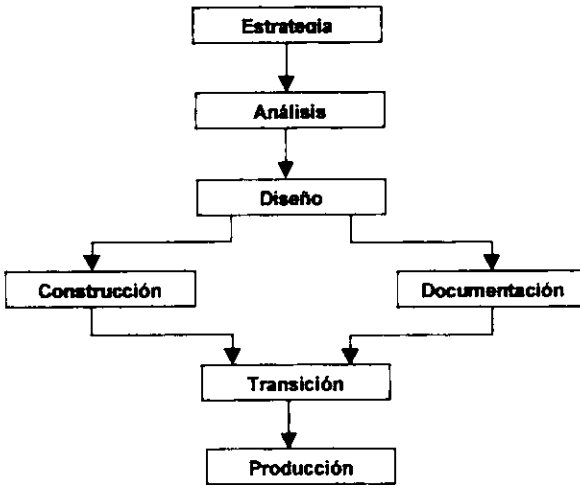


Figura 2.15 Ciclo de Vida en un desarrollo estructurado²³

Para cada fase, los productos generados en la mayoría de los desarrollados estructurados, son:

Estrategia	Análisis	Diseño
<ul style="list-style-type: none"> • Términos de referencia. • Diagramas de contexto. 	<ul style="list-style-type: none"> • Diagrama Entidad – Relación. • Diagrama de descomposición funcional, es decir, Diagrama Jerárquico Funcional • Diagrama de Flujo de Datos • Matrices de validación • Volúmenes de datos. • Plan de desarrollo de Producto 	<ul style="list-style-type: none"> • Arquitectura del sistema. • Diseño de los módulos • Diseño de la Base de Datos. • Procedimientos operativos o manuales

Construcción	Documentación	Transición	Producción
<ul style="list-style-type: none"> • Diseño de programas • Base de Datos afinada. • Programas trabajando y probados • Plan de transición. • Resultados de pruebas del sistema. • HW y SW instalado • Indicadores de rendimiento para la aplicación. 	<ul style="list-style-type: none"> • Documentación del usuario. • Documentación técnica 	<ul style="list-style-type: none"> • Material de entrenamiento • Instalación del sistema. • Conversión de datos, de sistemas viejos o archivos necesarios. 	<ul style="list-style-type: none"> • Archivos de respaldo y recuperación • Registro de control de cambios. • Reporte de fallas. • Cambios – adiciones al sistema. • Estadísticas de rendimiento • Nuevos requerimientos. Resultados de la auditoría al sistema.

Figura 2.16 Entregables por cada fase (Diseño Estructurado).

²³ Barker. CASE Method: tasks and deliverables, [BAR90]: pág. 3-1

2.4.2 EL PROCESO UNIFICADO DE DESARROLLO DE SOFTWARE. (THE UNIFIED SOFTWARE DEVELOPMENT PROCESS).

El Proceso Unificado de Desarrollo de Software (*The Unified Software Development Process*), al igual que cualquier otra metodología, especifica el orden en que deben realizarse las actividades del equipo de trabajo; conjunta las tareas de los desarrolladores individuales y las del equipo como un todo; especifica cuáles entregables deben construirse y ofrece criterios para monitorear y medir los productos y las actividades de los productos. Sus principales características son [JAC99]:

- Está basado en componentes, es decir, los sistemas de software están contruidos con base en componentes de software interconectados via interfaces bien definidas.
- Se basa completamente en el concepto de Casos de Usos.
- Está centrado en la arquitectura.
- Es un proceso iterativo e incremental.
- Se modela utilizando el Lenguaje Unificado de Modelado (*Unified Modeling Language UML*)

El Proceso Unificado consta de cinco áreas o flujos de trabajos (*workflows*) que conforman el eje central de la parte iterativa e incremental de los sistemas. Estas áreas son:

- **Requerimientos (*Requirements*)**. Donde se describen las características y funciones que debe conformar un sistema.
- **Análisis (*Analysis*)**. Donde se estudian los requerimientos, se refinan y se estructuran para su total comprensión.
- **Diseño (*Design*)**. Donde se detecta la forma (arquitectura) a partir de los requerimientos (incluyendo los no funcionales y los condicionales).
- **Implementación (*Implementation*)**. Donde, a partir del diseño, se construye el sistema en términos de componentes (código fuente, guiones o scripts, y demás archivos).
- **Prueba (*Test*)**. Donde se verifica el resultado de la implementación al probar cada producto construido antes de ser liberado el sistema, al exterior.

Estas áreas, generan a su vez, una serie de entregables donde podemos distinguir, en el caso del modelado los siguientes:

Requerimientos	Análisis	Diseño
<ul style="list-style-type: none"> • Modelo de Casos de Usos con sus actores, casos de uso y relaciones. • Descripción de la arquitectura (desde el punto de vista del modelo de Casos de Usos). • Glosario de términos. 	<ul style="list-style-type: none"> • Modelo de Análisis. • Clases del análisis. • Análisis de la realización de los Casos de Uso. • Paquete de análisis. • Descripción de la arquitectura (desde el punto de vista del modelo de Análisis) 	<ul style="list-style-type: none"> • Modelo de Diseño. • Clases del diseño. • Diseño de la realización de los Casos de Uso. • Subsistemas del diseño. • Interfaces • Descripción de la arquitectura (desde el punto de vista del modelo del Diseño). • Modelo de Implantación. • Descripción de la arquitectura (desde el punto de vista del modelo de Implantación).
Implementación	Prueba	
<ul style="list-style-type: none"> • Modelo de Implementación. • Componentes. • Subsistemas de implementación. • Interfaces. • Descripción de la arquitectura (desde el punto de vista del modelo de Implementación). • Plan de la integración de los pasos incrementales. 	<ul style="list-style-type: none"> • Modelo de Pruebas. • Prueba de Casos • Prueba de procedimientos. • Prueba de componentes. • Prueba del plan. • Defectos encontrados. • Prueba de evaluación. 	

Figura 2.17 Entregables por cada Fase (Proceso Unificado).

Los flujos de trabajo (workflows) se repiten sobre una serie de ciclos durante la vida del sistema. Estos ciclos terminan con la liberación de un producto al cliente. Cada ciclo consiste de cuatro fases, que a continuación se mencionan y que se pueden entender mejor en la figura 2.18.

- Inicio (*Inception*). Establece la regla del negocio para el proyecto.
- Elaboración (*Elaboration*). Establece el plan del proyecto y la arquitectura.
- Construcción (*Construction*). La creación del sistema.
- Transición (*Transition*). La liberación del sistema a los usuarios.

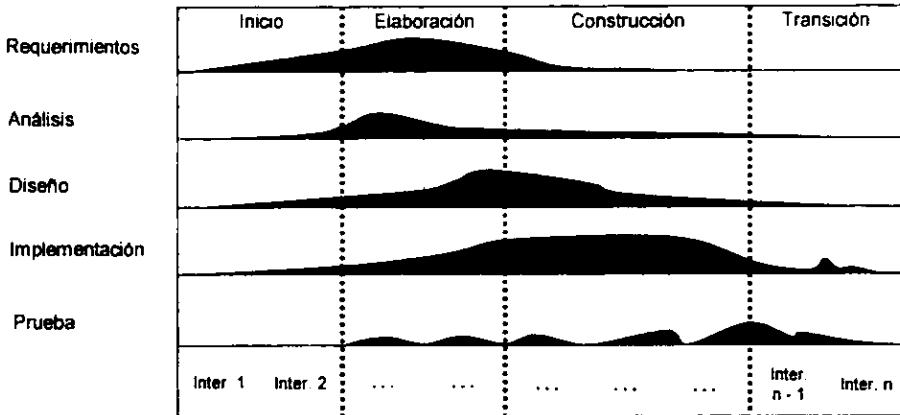


Figura 2.18 Fases y flujos de trabajo (Proceso Unificado)²⁴

2.4.3. LAS MÉTRICAS EN LA ADMINISTRACIÓN DE PROYECTOS.

DEFINICIONES

Dentro del contexto de la ingeniería de software, una *medida* proporciona una indicación cuantitativa de la extensión, cantidad, dimensión, capacidad o tamaño de algún atributo de un producto o proceso. La *medición* es el acto que se realiza para determinar una medida. Una *métrica* es una medida cuantitativa del grado en que un sistema, componente o proceso, tenga un atributo dado [PRE97].

Un *indicador* es una métrica o combinación de métricas que proveen una vista del proceso de software, de un proyecto de software o de un producto dado. Las *diferencias (variance)* son aquellas diferencias entre los gastos reales y los planeados, y los *costos proyectados* son estimaciones del costo actual basado en la razón de gasto actual [CAN98].

MÉTRICAS GENÉRICAS

El objetivo de las métricas de software es proveer al GIS y al grupo de administradores, de la información básica sobre el estado del proceso de software, por lo que es fundamental que la calidad de los productos de software y del progreso del proyecto, sean medidos durante el desarrollo del ciclo de vida del software. Dentro de la información que encontramos está [ROY99]:

- Una valoración exacta del progreso a la fecha.
- Una vista de la calidad de los productos de software.
- Una base para estimar el costo y los tiempos necesarios para completar el producto incrementando, con el tiempo, la exactitud.

²⁴ Jacobson, Ives et al. *The Unified Software Development Process*; [JAC99]; pág. 11

Podemos identificar siete métricas generales [ROY99]:

- Indicadores de Administración:
 1. Trabajo y progreso (trabajo realizado en el tiempo).
 2. Costo presupuestado y gastos (costos incurridos en el tiempo).
 3. Equipo de trabajo y dinámica del equipo (cambios de personal en el tiempo).
- Indicadores de Calidad:
 4. Tránsito de cambios y estabilidad (tránsito de cambios en el tiempo).
 5. Estropicios y modularidad (estropicios promedio por cambios en el tiempo).
 6. Duplicación de trabajo y adaptabilidad (duplicación de trabajo promedio por cambios en el tiempo).
 7. Lapso entre fallos (MTBF *mean time between failures*) y maduración (defectos en el tiempo).

En forma resumida, la siguiente tabla muestra las anteriores métricas con sus propósitos fundamentales y las mediciones más utilizadas, las cuales varían sólo en el enfoque con el que se desea realizar el seguimiento.

MÉTRICA GENÉRICA	PROPÓSITO	MEDICIONES POSIBLES
Trabajo y progreso	Compara lo planeado contra lo actual. Permite realizar una planeación iterativa. Es un indicador administrativo.	SLOC. Puntos de función, Puntos de objeto. Escenarios. Casos de prueba. SCO. BCWS. BCWP. ACWP.
Costo presupuestado y gastos	Compara lo planeado contra lo actual. Ofrece una visión financiera. Es un indicador administrativo.	Costos por mes. Equipo de trabajo de tiempo completo por mes. Porcentaje utilizado del presupuesto. BCWS. BCWP. ACWP. BAC. RC.
Equipo de trabajo y su dinámica	Realiza la comparación entre los recursos planeados y los actuales.	Personal adicional por mes. Personal liberado por mes.
Cambios y estabilidad	Permite realizar una planeación iterativa. Es un indicador administrativo sobre la convergencia del calendario.	SCO actuales contra los SCO realizados por tipo ²⁵ , versión, componente y subsistema.
Estropicios y modularidad ²⁶	Permite revisar la convergencia. Controla el software desechado. Es un indicador de calidad.	SLOC rehechas a solicitud de cambios por tipo, versión, componente y subsistema.
Trabajo duplicado y adaptabilidad	Permite revisar la convergencia. Controla el trabajo duplicado en la construcción del software. Es un indicador de calidad.	Número de horas promedio a solicitud de cambios por tipo, versión, componente y subsistema.
MTBF y maduración	Permite realizar pruebas cobertura / adecuación. Controla la robustez de uso. Es un indicador de calidad.	Conteo de errores. Horas de pruebas hasta encontrar errores por versión, componente y subsistema.

Figura 2.19 Métricas genéricas.

²⁵ Error crítico, error, mejoras, nueva característica y otro caso.

²⁶ Breakage and modularity [ROY99].

Donde [CAN98]

- ACWP** *Actual Cost of Work Performed*. Costo actual del trabajo realizado.
- BCWP** *Budget Cost of Work Performed*. Costo presupuestado del trabajo realizado.
- BCWS** *Budget Cost of Work Scheduled*. Costo presupuestado del trabajo calendarizado.
- SCO** *Software Change Order*. Orden de cambios al software
- SLCO** *Software Lines of Code*. Líneas de código del software.
- BAC** *Budgeted at Completion*. Presupuesto al terminar el proyecto
- RC** *Remaining Cost*. Presupuesto restante.

De las medidas anteriores, se desprende el análisis del valor ganado, donde, en forma adicional podemos calcular los siguientes valores

- CV** *Cost Variance*. Diferencia de costos. Diferencia entre los gastos planeados y los reales. $CV = BCWP - ACWP$.
- SV** *Schedule Variance*. Diferencia de lo calendarizado. Diferencia de los costos del trabajo realizado hasta la fecha y el costo de trabajo calendarizado hasta la fecha. $SV = BCWP - BCWS$.
- PC** *Percent Complete*. Porcentaje terminado. Porcentaje del trabajo calendarizado hasta la fecha y que se ha terminado. $100 \times BCWP / BCWS$
- CVI** *Cost Variance Index*. Índice de la diferencia de costos. Es la diferencia de costos normalizada por el costo presupuestado del trabajo realizado. $CVI = CV / BCWP = (BCWP - ACWP) / BCWP$.
- SVI** *Scheduled Variance Index*. Índice de la diferencia de lo calendarizado. Es la diferencia calendarizada normalizada por el costo presupuestado del trabajo calendarizado. $SVI = SV / BCWS = (BCWP - BCWS) / BCWS$
- CPI** *Cost Performance Index*. Índice del costo del rendimiento. Es la razón del valor ganado y del costo actual. $CPI = BCWP / ACWP$. Si el $CPI < 1.0$ entonces el costo actual es mayor que el valor ganado, si sigue así, no se terminará dentro del presupuesto
- EAC** *Estimate at Complete*. Proyección del costo final estimado del proyecto. Indica el costo final más acertado a la realidad. $EAC = BAC / CPI = (BAC \times ACWP) / BCWP$, o bien, $EAC = (ACWP / BCWP) \times BCWP + RC$
- VAC** *Variance at Complete*. Diferencia al terminar. $VAC = EAC - BAC$. Si el $VAC > 0$ entonces hay que realizar ajustes

Entre otros análisis que podemos hacer con la información anterior, podemos, detectar el estado actual de los costos y tiempos del proyecto reflejados en la siguiente tabla, donde se relacionan el CV y el SV.

	SV >= 0	SV < 0
CV >= 0	En o bajo el presupuesto y en o sobre lo calendarizado	En o bajo el presupuesto y retrasados conforme al calendario.
CV < 0	Sobre el presupuesto y en o sobre lo calendarizado.	Sobre el presupuesto y retrasados conforme al calendario.

Figura 2.20 Relación entre diferencias de costos y de calendario (CV SV).

2.4.4. EVALUACIÓN Y PRUEBAS EN LA ADMINISTRACIÓN DE PROYECTOS.

DEFINICIÓN DEL PROCESO DE PRUEBAS

Dentro del contexto de la ingeniería de software, el *Proceso de pruebas* es la preparación, planeación y medición dirigidos a establecer las características de los sistemas de información y demostrar las diferencias entre el estado actual y el requerido. Las pruebas reducen el nivel de incertidumbre en la calidad del producto de Software. Los productos más importantes del proceso de pruebas son [KOO99]:

- *Defectos detectados*: Las diferencias entre los resultados esperados y los reales.
- *Información y sugerencias de la calidad del objeto probado*: Basados en los defectos encontrados, se puede tener una visión de la calidad del producto y mejoras posibles para la siguiente versión o fase.
- *Testware*: El testware es el conjunto de productos realizados en el proceso de pruebas, como los planes, casos de prueba específicos, bases de datos de prueba, escenarios y prototipos de pruebas.
- *Documentación restante*: Otros documentos aparte de los de testware, como balances y reportes.
- *Indicadores*: Indicadores que muestran el progreso, los tamaños y la calidad del proceso de pruebas.

DEFINICIÓN DE CALIDAD

El objetivo principal que busca el proceso de pruebas, es lograr los niveles de calidad de los productos de software que existen en el desarrollo de un proyecto de software. Es así que, usamos la definición que el estándar ISO 8402 propone: la *calidad* como la totalidad de características y propiedades de un producto o servicio en las que se apoya para satisfacer necesidades tangibles o implícitas. Para esto, existen una serie de acciones orientadas a lograr calidad en los productos de software: acciones preventivas, acciones de detección y acciones correctivas.

DEFINICIÓN DE EVALUACIÓN

Las pruebas pertenecen a las acciones de detección de defectos en un sistema. Están relacionadas a las revisiones, simulación, inspección, auditoría, ejemplificación, pruebas de escritorio, recorridos y demás. *Evaluación*, también llamada verificación, es la examinación de productos intermedios de las fases de desarrollo, en la mayor parte por inspección o revisión de documentos. Normalmente la evaluación no interviene en la ejecución del software. Las pruebas o la validación, inspeccionan los productos finales y lo logran al ejecutar el software. La evaluación, por ejemplo, implica también la inspección del testware [KOO99].

DIFERENCIA ENTRE CORREGIR Y PROBAR

Un aspecto crítico para los administradores, desarrolladores y usuarios es el entender, desde el principio, la diferencia entre corregir defectos (debugging) y probar [SOF98].

- *Corregir defectos (debuggin)* es el proceso de aislar e identificar la causa de un problema del software, y entonces modificarlo para corregir dicho problema.
- *Probar* es el proceso de encontrar defectos en función de un conjunto de criterios y especificaciones predeterminados. El propósito de este proceso es probar que un sistema, software o configuración de software, *no funciona en vez de que sí lo haga*.

Hay dos formas de probar: *usando pruebas de caja blanca* y *usando pruebas de caja negra*. Un ambiente de pruebas ideal alterna entre las actividades de pruebas de caja blanca y de caja negra, primero probando el diseño y luego demostrando que realiza la funcionalidad requerida, de manera confiable, consistente con las restricciones del usuario, de desempeño y de operación.

Las *pruebas de caja blanca* se realizan en los componentes de código, que pueden ser unidades de software, componentes de software de computadora (CSC), o elementos de la configuración de software de computadora (ECSC). Estas pruebas ejercitan la estructura interna de un componente de código, e incluyen:

- La ejecución de cada instrucción en un componente de código, al menos una vez.
- La ejecución de cada camino de una condicional en el componente de código.

- La ejecución de los caminos con valores de entrada de frontera y con valores más allá de estas fronteras.
- La verificación de la integridad de las interfaces internas
- La verificación de la integridad de la arquitectura a través de un rango de condiciones.
- La verificación del diseño y estructura de la base de datos.

Las pruebas de caja negra se realizan a los componentes funcionales, cuya integridad del diseño ha sido verificada hasta la terminación del rastreo de las pruebas de caja blanca. Como en las pruebas de caja blanca, estos componentes incluyen unidades de software, CSC o ECSC.

Las pruebas de caja negra rastrean los requerimientos enfocados en los aspectos externos del sistema. Validan que el software cumpla con los requerimientos sin tomar en cuenta los caminos de ejecución llevados a cabo para cumplir cada uno de ellos. Este es el tipo de prueba conducida de software, esto es, la integración de unidades de código. El proceso de pruebas de caja negra incluye:

- La validación de la integridad funcional en relación con los estímulos externos.
- La validación de todas las interfaces externas (incluyendo humanas) a través de un rango de condiciones normales y anómalas
- La validación de la capacidad del sistema, software o hardware para recuperarse o minimizar el efecto de imprevistos, aspectos externos anómalos o condiciones ambientales
- La validación de la capacidad del sistema para direccionar datos de entrada fuera de las fronteras, recuperación de errores, comunicación y condiciones de estrés

Idealmente, cada prueba de caja negra debería estar precedida por una prueba de caja blanca que establezca el diseño. La figura 2.21 ilustra el *Proceso de Pruebas* [HET00].

Nivel	Actividad de la Prueba	Tipo de Prueba	Base Documental de la Prueba	Responsabilidad de la Prueba	Enfoque de la Prueba
0	Inspecciones Estructuradas	Pruebas No Basadas en Computadora	Varios	Equipo de Inspección	Varios
1	Pruebas de Unidad de Software de Computadora	Pruebas de Caja Blanca	ADS	Desarrollador	Diseño de Unidad de Software
2	Pruebas de Integración ECSC	Pruebas de Caja Blanca	DDSW	Prueba independiente	Diseño / Arquitectura ECSC
3	Pruebas de Calificación ECSC	Pruebas de Caja Negra	ERS	Prueba independiente	Requerimientos ECSC
4	Pruebas de Integración ECSC / ECHW	Pruebas de Caja Blanca	DDS	Prueba independiente	Diseño / Arquitectura del Sistema
5	Pruebas del Sistema	Pruebas de Caja Negra	ESS	Prueba independiente	Requerimientos del Sistema
6	Pruebas de PDyE	Pruebas de Caja Negra	Manuales de Usuario	Grupo de Pruebas	Conformidad del Manual de Usuario
7	Pruebas de POyE	Pruebas de Caja Negra	DRO o Documentos de Requerimientos de Usuarios	Pruebas Operacionales	Requerimientos Operacionales
8	Pruebas de Sitio	Pruebas de Caja Negra	Plan de Transición (Configuración de Sitio)	Equipo de Instalación de Sitio	Requerimientos de Sitio

Donde:

PDyE	Pruebas de Desarrollo y Evaluación	ERS	Especificación de Requerimientos de Software.
ECHW	Elemento de la Configuración del Hardware.	DDS	Documento de Diseño de Subsistemas.
DRO	Documento de Requerimientos Operacionales.	ESS	Especificación de Segmento del Sistema.
POyE	Pruebas Operacionales y Evaluación	DDSW	Documento de Diseño del Software.
ADS	Archivo de Diseño del Software.		

Figura 2.21 Proceso de Pruebas²⁷

Nivel 0 – Estas pruebas consisten de un conjunto de inspecciones estructuradas ligadas a cada producto colocado bajo la administración de la configuración. El propósito de las pruebas de Nivel 0 es eliminar los defectos en el momento cuando ocurren, y antes de que afecten a otros productos.

Nivel 1 – Estas pruebas de caja blanca califican el código contra los estándares y la especificación de diseño de unidades. Este es el único nivel de prueba que se enfoca al código.

Nivel 2 – Estas pruebas de caja blanca integran CSCs calificados en una configuración ejecutable de ECSC. Estas pruebas se enfocan a las interfaces internas de los CSC.

Nivel 3 – Estas pruebas de caja negra ejecutan los ECSC integrados para asegurar que los requisitos de la Especificación de Requerimientos de Software (ERS) hayan sido implementados y que el ECSC se ejecute de manera aceptable. Los resultados de las pruebas de Nivel 3 son revisados y aprobados por quien adquirirá el producto.

Nivel 4 – Estas pruebas de caja blanca integran los ECSC, en una configuración del sistema ejecutable al conectar ECSC independientes y así integrar la configuración del software ejecutable con el hardware en el que se instalará.

Nivel 5 – Estas pruebas de caja negra califican una configuración del sistema, para asegurar que los requerimientos del sistema se han cumplido y que el concepto básico del sistema fue satisfecho. Este nivel de prueba usualmente resulta en la aceptación o al menos, en la aprobación del sistema bajo las pruebas basadas en el cliente.

Nivel 6 – Estas pruebas de caja negra, planeadas y conducidas por la persona que adquiere el producto, rastrean los manuales del usuario y del operador así como a las especificaciones de interfaces externas.

Nivel 7 – Las pruebas independientes de caja negra rastrean los requerimientos y especificaciones operacionales. Las pruebas de nivel 7 son conducidas por agentes del usuario para asegurar que los requerimientos del ambiente, tanto los operacionalmente críticos, de seguridad y otros, hayan sido satisfechos y que el sistema esté listo para implantarse.

Nivel 8 – Estas pruebas de caja negra son conducidas por el equipo de instalación para asegurar que el sistema trabaje correctamente cuando se instale, y se ejecute correctamente cuando se conecte a las interfaces reales del sitio. Las pruebas de nivel 8 rastrean los manuales de instalación y utilizan hardware y software para realizar el diagnóstico.

PRUEBAS GENÉRICAS

La figura 2.22 muestra las pruebas genéricas que se realizan dentro el desarrollo del Proyecto de Software, en función del nivel donde se clasifican dentro del Proceso de Pruebas del punto anterior.

Tipo de Revisión. Las revisiones en forma general, pueden ser:

- | | |
|--------------------------|--|
| <p>Informales</p> | <p>Son las revisiones que se realizan en forma personal o entre gente de confianza con conocimiento en la información, proceso o entregable que se desea revisar. Se conocen como <i>walkthrough</i>, dando la idea que se realiza un "paseo" o una revisión informal según el criterio de quien lo realiza.</p> |
|--------------------------|--|

²⁷ Hetzel, Bill. *The little book of testing: vol. II [HET00]*

- Entre colegas Estas revisiones se llevan a cabo entre elementos del equipo de trabajo con el conocimiento necesario y suficiente para poder corregir y mejorar el trabajo de un compañero. Las personas involucradas en este tipo de revisiones deben estar trabajando conjuntamente en el mismo entregable o en productos muy relacionados, de tal forma que sus aportaciones puedan llegar a un nivel de detalle provechoso para el proyecto.
- Entre pares Estas revisiones se llevan a cabo entre elementos del equipo de trabajo con el conocimiento básico para poder corregir y mejorar el trabajo de un colega. Este tipo de revisiones es muy útil para que un elemento experimentado pueda llevar el control del trabajo de un equipo con poca experiencia, y poder rastrear la capacitación y el nivel de madurez del personal [ROY99]. Por lo tanto, no es necesario que las personas involucradas en este tipo de revisiones estén trabajando en el mismo entregable, y esto permite que se tengan diferentes perspectivas de revisión.
- Formales Estas revisiones se llevan a cabo por un grupo especialmente asignado para esta tarea (pueden ser elementos del mismo equipo de trabajo), siguiendo una planeación establecida. Para este tipo de revisiones existe el material previo y elementos auxiliares (listas de verificación, guías, cuestionarios), así como un proceso bien establecido para registrar los defectos, tiempos de revisión y corrección, clasificación de los defectos, perfil y capacidad del equipo de trabajo.

Elementos de prueba. Los productos, procesos y entregables que se deben revisar generan modelos de prueba que pueden clasificarse en los siguientes conjuntos [ROY99]:

- Administrativo En este conjunto de elementos se encuentran todas las especificaciones y descripciones relevantes que capturan los objetivos, alcances, lineamientos, políticas, organización, y todo aquello que tenga que ver con el control, planeación, y las negociaciones resultantes entre los elementos del equipo de trabajo interno y del seguimiento del proyecto. El modelo de prueba debe contemplar los criterios de evaluación de los resultados de los puntos esenciales (*milestones*) del Plan de Desarrollo. Como resultado de las pruebas encontramos defectos, cambios, ambigüedades entre los requerimientos, mejoras, así como criterios de aceptación asociados con los cambios discretos a las líneas base.
- Requerimientos En este conjunto de elementos se encuentran todos los productos que dan origen a un proyecto de software. El modelo de prueba debe agrupar los conceptos operacionales del nivel funcional del sistema así como la descripción de casos de pruebas, incluyendo el comportamiento esperado del sistema y sus atributos de calidad. El conjunto completo de requerimientos es un elemento de prueba porque es la base de todas las actividades de evaluación durante el ciclo de vida.
- Diseño Un modelo de prueba para los componentes entregables necesita revisar los productos de las líneas base y esto se logra en el diseño. Estos componentes incluyen los entregables del análisis y del diseño tomando la simulación de eventos para crear datos realistas; un "operador virtual" que pueda soportar casos de pruebas sobre productos no ejecutables; con instrumentos específicos para realizar demostraciones tempranas (antes de la construcción) del uso de los recursos, de las velocidades de transacción, de tiempo de respuestas y del resultado de las pruebas dirigidas de casos de uso sobre componentes aislados.
- Implementación En este modelo de pruebas, los códigos fuente documentados, constituyen los componentes de las pruebas dirigidas. Estos códigos que se encuentran dentro de los archivos fuente, proveen los procedimientos y los guiones de las pruebas a realizarse. También pueden incluir archivos con datos legibles que representen conjuntos específicos de datos definidos estáticamente. Ambos tipos de archivos conforman los archivos fuentes de pruebas, y los archivos resultantes de las pruebas, conformarían el equivalente a los reportes de pruebas.

Implantación En este conjunto de elementos se encuentran las versiones ejecutables de los componentes de pruebas, guiones y procedimientos de las pruebas dirigidas, y todos los archivos de datos generados en las etapas anteriores.

Prueba de	Nivel	Tipo de Revisión	Elementos de prueba
Estrategia	0	Informal	Administrativo
Modelo de Ciclo de Vida	0	Informal y entre colegas	Administrativo
Planeación y Estimación	0	Entre colegas	Administrativo
Compromisos, motivación y capacitación	0	Entre colegas	Administrativo
Comunicación e información	0	Entre colegas, entre pares	Administrativo
Requerimientos	0, 3	Entre colegas, entre pares y Formales	Requerimientos
Análisis y Diseño	0	Entre colegas, entre pares y Formales	Diseño
Unidad de Software	1	Formales	Implementación
Integración	2, 4	Formales	Implementación
Sistema	5, 6, 7	Formales	Implementación e implantación
Instalación	7, 8	Formales	Implantación

Figura 2.22 Mapeo del Proceso de Pruebas con las Pruebas Genéricas.

CRITERIOS DE VALIDACIÓN

La figura 2.23 muestra los criterios que surgen de las métricas y de las validaciones (pruebas) que en conjunto evalúan la calidad de un producto dado. Estos criterios sirven para poder conocer si el estado actual de los entregables de una fase, están en un nivel adecuado de calidad, como para poder ser integrados a las bibliotecas de líneas base con ese status y poder continuar con el ciclo de vida del proyecto.

Criterio	Descripción
Involucrados	Que todos los involucrados intervengan en la definición de objetivos y estimados en alcances, costos y tiempos (<i>fase de Inicio</i>). Que todos los involucrados coincidan que la arquitectura seleccionada cubre sus requerimientos y expectativas (<i>fase de Elaboración</i>).
Requerimientos	Que todos los requerimientos iniciales queden perfectamente entendidos y modelados.
Estimados	Que los estimados de costos y tiempos, prioridades, riesgos y procesos de desarrollo sean creíbles y cercanos a la realidad.
Arquitectura inicial	Que el alcance de la arquitectura propuestas demuestre su viabilidad para ser generada por el equipo de trabajo sobre la plataforma técnica sugerida.
Seguimiento	Que los recursos y gastos reales sean comparables con los planeados.
Estabilidad de visibilidad	Que los objetivos y estimados se mantengan estable.

Criterio	Descripción
Estabilidad de solución	Que la arquitectura seleccionada se mantenga estable
Riesgos	Que se demuestre realmente que los riesgos principales detectados están bajo control.
Congruencia	Que el plan de la construcción conserve la fidelidad y congruencia con base en estimados y resultados anteriores.
Defectos	Que los defectos actuales en los productos de las líneas base no sean obstáculos para lograr los propósitos de la siguiente liberación.
Estabilidad de construcción	Que el producto de la línea base sea suficientemente estable para que sea instalado y probado en el equipo del usuario.
Transición	Que todos los involucrados estén listos para la transición en las instalaciones del usuario.
Satisfacción	Que el usuario se encuentre satisfecho con el producto final

Figura 2.23 Criterios de Evaluación.

2.4.5. MARCOS Y PATRONES.

CONCEPTO DE PATRÓN

Un patrón es un molde, modelo u horma, altamente recomendable para ser emulado. Es el plano de un producto más que el producto mismo. También puede verse como la representación perfecta de un ejemplo de la vida real, aplicado como una regla genérica a todos aquellos casos que tengan el mismo comportamiento. Un patrón es un conjunto de información instructiva con un nombre, que captura la estructura y el conocimiento esenciales de soluciones probadas exitosamente, a un problema recurrente que aparece dentro de cierto contexto del sistema [BRO98].

El patrón contesta, en su propio lenguaje o especificación, a las preguntas fundamentales del comportamiento de lo que se quiere generalizar. Por lo tanto, debe tener una estructura común, lógica y consistente [COA95]

CONCEPTO DE MARCO (FRAME)

Un marco es un conjunto de estructuras abstractas y concretas que colaboran y que pueden servir de plantilla para resolver una familia afín de problemas. Para este trabajo, describirá la estructura fundamental de la Guía para la Administración de Proyectos Pequeños de Software, y estará basada en la especificación de patrones de proceso y de organización.

PATRONES Y ANTIPATRONES DE PROCESO

Un patrón de proceso, es un patrón que describe una serie de acciones probadas exitosamente para desarrollar software. Existen al menos tres tipos importantes de patrones de proceso [AMB99]

1. **De tareas:** Este tipo de patrón describe los pasos detallados para realizar una tarea específica.
2. **De etapas:** Este tipo de patrones describe los pasos que generalmente se realizan iterativamente, dentro de una sola etapa del proyecto. Un patrón de este tipo está presente por cada etapa del proyecto.
3. **De fases:** Este tipo de patrón describe las interacciones entre los patrones de proceso de etapas para una fase única del proyecto.

También existen los antipatrones de proceso. Un antipatrón describe ciertas soluciones a un problema común, que en su momento prueban estar equivocadas o son muy ineficaces en detrimento de la organización.

PATRONES DE ORGANIZACIÓN

Son patrones que describen soluciones probadas exitosamente para la organización y administración de la gente involucrada en el Proceso de Software. Los patrones de proceso van de la mano con los patrones de organización, e igual que en estos, existen al menos tres tipos importantes de patrones de organización [AMB99]:

1. De estructura: Los patrones de organización estructurales establecen patrones de interacción y coordinación tecnológica y humana en la organización.
2. De proceso: Los patrones de organización de procesos son patrones relacionados al flujo de la información, comunicación y toma de decisiones dentro de las organizaciones.
3. De comportamiento: Los patrones de organización de comportamiento tratan con la delegación de responsabilidades dentro de las organizaciones, cubriendo roles y algunos patrones de administración.

SEGUNDA PARTE
GUÍA PARA LA ADMINISTRACIÓN DE
PROYECTOS DE SOFTWARE

CAPÍTULO 3. Guía para la Administración de Proyectos

Este capítulo presenta toda la estructura y la forma en cómo está organizada la guía. Proponemos representarla auxiliándonos del concepto de patrones y en especial, de los patrones de procesos (sección 2.4.5). En forma gráfica, la guía se esquematiza con ayuda de los Diagramas de Actividad¹ y, en una Tabla Descriptiva (sección 3.2.3) se resumen los aspectos básicos con que consta la guía, haciendo alusión a los elementos teóricos vistos en el capítulo anterior, como lo referente al CMM-2. Por otro lado, se plantea la forma en cómo se realizó el mapeo de los procesos propuesto por el PMBOK, a la administración en particular de un proyecto de software, proponiendo un ciclo de vida que conjunte las propuestas vistas en el capítulo 2.

Para cada fase del Ciclo de Vida entonces, la guía presentará un Marco con la estructura que se explica en la sección 3.2.1. Cada Marco (fase) está formado por cada Patrón de los roles que intervienen en las actividades de administración y, haciendo hincapié en aquéllos que tienen una relación directa con el rol del Administrador del Proyecto. Cada patrón estará formado por:

- El patrón en sí, con la estructura que se explica en la sección 3.2.2.
- Una gráfica con el flujo de procesos que realiza el rol usando los Diagramas de Actividad².
- Una Tabla Descriptiva con la estructura que se especifica en la sección 3.2.3.

3.1 Ciclo de Vida utilizado para esta Guía.

Como ya mencionamos en el punto 2.1.1, decimos que el conjunto de fases de un proyecto se le conoce como el *ciclo de vida del proyecto*. En el punto 2.4.1, se presentó en forma general, el ciclo de vida de un proyecto de software basado en el paradigma estructurado y en el punto 2.4.2, presentamos el ciclo de vida de un proyecto de software basado en el paradigma orientado a objetos. A continuación, se propone una combinación de ambos ciclos, con las fases y los traslapes entre ellas, necesarias para poder aplicar los conceptos vistos del PMBOK (sección 2.2).

3.1.1. CICLO DE VIDA.

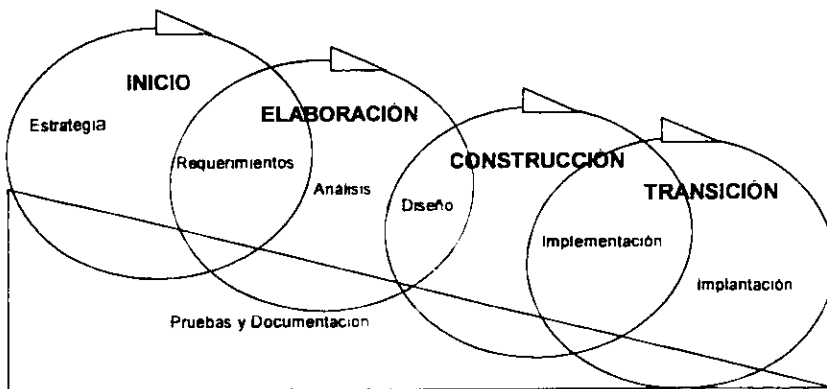


Figura 3.1 Ciclo de vida propuesto para Guía.

¹ Ver el anexo A.

² Idem.

FASES

- I. **Inicio.** En esta fase se obtiene una visión del producto final y del modelo del negocio del usuario. En forma general, se especifica, junto con el usuario, los requerimientos, la arquitectura del sistema, y un plan de desarrollo, todos desde un punto de vista general. Esta fase está formada a su vez por:
 - **Estrategia.** Donde se produce el modelo del negocio y un plan de desarrollo del sistema, cubriendo los requerimientos actuales y futuros del negocio, tomando en cuenta las restricciones de la organización, financiera y técnicas. Representa la justificación del proyecto y describe y define la infraestructura básica de trabajo.
 - **Requerimientos.** Donde se describen las características, recomendaciones y funciones iniciales que debe conformar un sistema, tanto requerimientos actuales como futuros.
 - **Pruebas y Documentación.** Donde se definen los documentos de administración iniciales y los resultados de esta fase se corroboran con los usuarios.

- II. **Elaboración.** Aquí se detallan cada uno de los requerimientos iniciales, y en forma iterativa se analizan y se diseñan, usando las técnicas que previamente se hayan seleccionado. También se afina el plan de desarrollo con sus respectivos calendarios y cotizaciones. Esta fase está formada a su vez por:
 - **Requerimientos.** Aquí, la recopilación, selección y estudio de los requerimientos se hace en forma aún más detallada, y se conforma la única base necesaria para el análisis.
 - **Análisis.** Aquí se verifican los resultados de los requerimientos, se detallan, refinan y estructuran para su total comprensión, y poder desarrollar alternativas válidas de solución.
 - **Diseño.** Donde se toman y se verifican los resultados de la etapa de análisis y se propone el mejor camino para cubrir los requerimientos, dado el ambiente técnico y los niveles requeridos de automatización. Además, se detecta la arquitectura que reside sobre los requerimientos incluyendo los no funcionales los condicionales.
 - **Pruebas y Documentación.** Donde todos los resultados del análisis de los requerimientos son validados y autorizados por el usuario.

- III. **Construcción.** En esta fase: basándose en el diseño, se procede a la construcción del producto, utilizando las herramientas adecuadas al problema, escogidas dependiendo del ambiente técnico y del tipo de programa. Esta fase está formada a su vez por:
 - **Diseño.** A partir de un diseño maduro, este se termina, detalla, o corrige según así lo requieran las características propias de la construcción, que no pudieron ser previstas por su finura técnica o por lo específico del propio diseño.
 - **Implementación.** Donde, a partir del diseño, se construye el sistema en términos de componentes (código fuente y demás archivos).
 - **Pruebas y Documentación.** Donde todos los resultados de la construcción se validan y verifican (según las variadas técnicas que para esto existen), para asegurar el cumplimiento cabal de los requerimientos.

- IV. **Transición.** Esta fase cubre el proceso de liberación del producto a los usuarios, siguiendo un plan de transición previamente pactado.
 - **Implementación.** Donde se realizan los ajustes finales conforme al resultado de las pruebas, sobre los códigos y archivos relacionados.
 - **Implantación.** Donde se realizan todas las tareas necesarias para la implantación del sistema proporcionando un periodo inicial de soporte y apoyo. Se asegura que el sistema funcione adecuadamente en su ambiente final.
 - **Pruebas y Documentación.** Donde se verifica que el resultado de la construcción y de la implantación, según las diversas técnicas que existen para esto. Se formalizan los manuales del usuario y la documentación de cada fase que ha quedado registrada en documentos, reportes y diagramas.

3.1.2. ROLES UTILIZADOS EN ESTA GUÍA.

ADMINISTRADOR PRINCIPAL.

El administrador principal realiza la administración en los altos niveles de la organización atendiendo principalmente en los intereses a largo plazo de la organización. En general, puede estar controlando múltiples proyectos, proveyéndoles de los recursos necesarios según su propia planeación. En las organizaciones pequeñas, este puede ser uno de los gerentes de la empresa.

ADMINISTRADOR DEL PROYECTO.

El administrador del proyecto, es quien tiene total responsabilidad de llevar a cabo un proyecto entero. El administrador del proyecto dirige, controla, administra y regula un proyecto para construir un producto o sistema de software. Dependiendo de la estructura de la organización, la gente que trabaja en un proyecto podría reportar al administrador del proyecto, o bien, solo le reportaría el equipo de administradores específico. El grupo de Ingeniería de Software podría tener una relación indirecta con el administrador del proyecto via el Líder del Proyecto.

ADMINISTRADOR DE LA CONFIGURACIÓN DEL SOFTWARE (ACS).

La administración de la configuración del software (ACS) identifica la conformación de un sistema en puntos discretos del tiempo con el propósito de controlar los cambios para mantener la integridad y el seguimiento de la configuración a través del ciclo de vida del sistema [BER80]

El administrador de la configuración es la figura responsable de la planeación, de la coordinación y de la comunicación de las actividades referentes a la administración de la configuración. Es quien, junto con el resto de los administradores, identifica y propone las líneas base del proyecto, así como los instrumentos y procedimientos para lograr la integridad de los productos del proyecto de software durante su ciclo de vida. El administrador puede verse apoyado por un grupo de personas (grupo de administración de la configuración del software (GACS)), que también podría fungir como el administrador de la configuración.

Para esta guía, las actividades del rol del Grupo de Administración de la Configuración del Software podrán ser realizadas por el propio Administrador de la Configuración, por tratarse de proyectos de tamaño reducido donde el principal interés, y por lo tanto, el mayor de los esfuerzos y recursos, se ve reflejado en la elaboración del Producto de Software.

ADMINISTRADOR DE LA CALIDAD DEL SOFTWARE (AQS).

El administrador de la calidad es la figura responsable de la planeación, de la coordinación y de la comunicación de las actividades referentes a la administración de la calidad. Es quien, de manera paralela y junto con el resto de los administradores identifica y evalúa el estado de los procesos que llevan todos los involucrados en el Proyecto de Software para garantizar el cumplimiento de los objetivos pactados dentro y fuera de la organización. El administrador puede verse apoyado por un grupo de personas (GAQS, grupo de administración de la calidad del software) que también podría fungir como el administrador de la calidad.

Igual que para la Administración de la Configuración, para esta guía, las actividades del rol del Grupo de Administración de la Calidad del Software podrán ser realizadas por el propio Administrador de la Calidad, y opcionalmente, por un tercero que funja como ente externo al proyecto y aporte una evaluación imparcial.

LÍDER DEL PROYECTO.

El Líder del proyecto es la figura responsable de llevar el liderazgo de los aspectos técnicos y del proceso de producción. Selecciona al personal que formará parte del equipo de desarrollo (Grupo de Ingeniería de Software, GIS), quien los motiva, les asigna sus tareas y lleva el seguimiento y supervisión de sus actividades para que el calendario de desarrollo planeado conjuntamente con los demás administradores, se lleve a cabo exitosamente. Resuelve los problemas técnicos y del equipo de trabajo. Es responsable también de la organización moral y técnica del equipo. A este respecto, toma las decisiones junto con el GIS, de las estrategias posibles, asegurando que se entreguen a tiempo los productos requeridos en cada fase. Trabaja en forma directa con el administrador del proyecto en la planeación y seguimiento y le reporta los avances.

3.1.3. CICLO DE VIDA CON PROCESOS DEL PMBOK.

A continuación, para cada fase del ciclo de vida del software que se presentó en el punto 3.2.1. y siguiendo el esquema de los procesos propuestos por el PMBOK del punto 2.1.2, figura 2.3, presentamos gráficamente la selección de los procesos que la conforman.

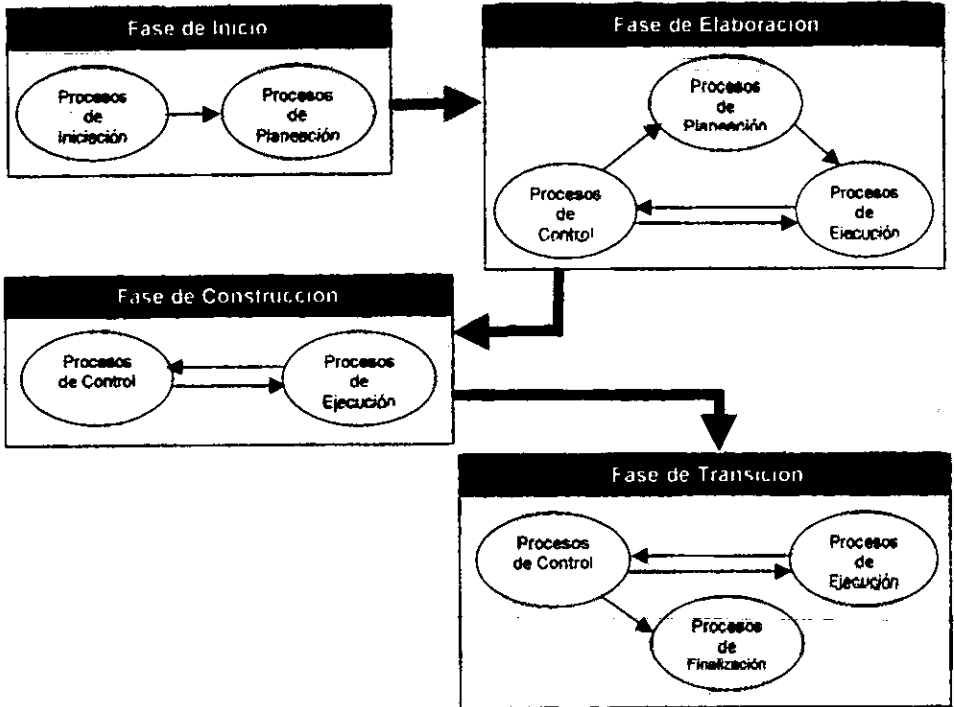


Figura 3.2 Mapeo de los procesos del PMBOK y del Ciclo de vida propuesto.

Además, los procesos de la Administración de Proyectos (sección 2.1.2), cuentan con una serie de subprocesos (figuras de la 2.6 a la 2.10) que, dependiendo la fase de la que se trate, podrán o no estar todos ellos. Para esta Guía, el detalle de dichos subprocesos se hará dentro del patrón de cada fase.

3.2. Marcos y patrones propuestos para la Guía.

En la sección 2.4.5 se presentó la definición de marcos y patrones. En este punto se hace uso de estos conceptos para proponer la estructura fundamental, que le da cuerpo a esta Guía para la Administración de Proyectos.

3.2.1. DESCRIPCIÓN DEL MARCO PROPUESTO PARA LA GUÍA

Proceso	Es el nombre genérico del conjunto de acciones llevadas a cabo para lograr un objetivo o realizar y entregar un resultado. Para este trabajo, usaremos la propuesta dada por el PMBOK.
Objetivo:	Es la motivación principal que da origen a la necesidad de crear un proceso independiente y relacionado, dentro del Proceso de Software. Describe la naturaleza y función principal de este proceso

Fase compuesta por	Detalla cada una de las partes en las que esta fase está compuesta, según el modelo del Ciclo de Vida que se haya seleccionado
Tiempo utilizado	Propuesta del tiempo promedio que esta fase deberá absorber del total del PYS
Factores críticos de Éxito	Lista todos los elementos de riesgo que podrían en un momento dado, dilatar, detener o reducir las posibilidades de éxito, de la fase en cuestión
Roles	La lista de recursos, particularmente humanos encargados de ejecutar las actividades. Estos roles también pueden ser ejecutados por sistemas externos y herramientas de apoyo
Entregables de la Fase	El o los productos de trabajo del proceso, tangibles y a los que se les pueden aplicar métodos de verificación. Aquí tenemos una primera clasificación general
Entradas	Documentos o elementos necesarios para iniciar el proceso
Salidas	Documentos o elementos que resultan del proceso
Subprocesos	En caso de contar con ellos, marca la configuración del proceso en otros tantos procesos. Esta referencia marcará la estructura general de la Guía
Métricas	Medidas básicas necesarias para determinar el estado del proceso
Verificaciones	Aseguramiento que las actividades se realizaron conforme al proceso
Procedimientos, Métodos, Técnicas e Instrumentos	Descripción detallada de métodos, técnicas, diagramas, instrumentos y demás elementos necesarios para llevar a cabo las actividades que anteriormente se listaron
Patrones aplicables.	Para cada proceso, pueden existir soluciones previamente validadas por la comunidad de IS, de tal forma que garanticen su efectividad. Con esto, podemos hacer uso de la experiencia y tomar las mejores prácticas en la aplicación de la administración de procesos de software. Aquí se listan aquellos patrones aplicables a los objetivos del proceso.
Antipatrones posibles.	Lista los posibles problemas potenciales que la comunidad de IS ha detectado en la administración de procesos de software. Este aspecto es de gran utilidad, pues nos permite aprender de los errores ajenos para evitarlos en el Proceso de Software.
Area clave del CMM	Se identifica el area clave según la clasificación que el CMM hace del proceso de software. Además, se explica como se cubren cada uno de los aspectos que marca el estándar

3.2.2. DESCRIPCIÓN DEL PATRÓN PROPUESTO PARA LA GUÍA

Nombre del Patrón	El patrón debe tener un nombre con significado. Esto nos permite usar una palabra o una frase corta para referirnos al patrón, al contenido y a la estructura que describe. Los nombres de los patrones forman el vocabulario de las discusiones de los conceptos abstractos [APP97]
Alias	Algunas veces, los patrones pueden tener más de un nombre. En este caso debemos documentar los alias o sinónimos en el rubro "También conocido como" [APP97]
Problema	Es el enunciado del problema que intentamos resolver, los objetivos que deseamos alcanzar dadas las fuerzas y dentro del contexto [APP97]

Contexto:	Describe las precondiciones bajo las cuales el problema y su solución parecen recurrir y para el cual la solución es deseable. Esto nos indica las posibilidades de aplicar dicho patrón. Podemos pensarlo como la configuración inicial del sistema antes de que el patrón se aplique [APP97]. <i>Este elemento del patrón, puede ser omitido en los roles cuyo contexto quedó explicado en fases anteriores.</i>
Relacionado con:	Lista los patrones con los que comparte relaciones dinámicas o estáticas. Puede compartir fuerzas comunes y contextos (iniciales o finales) con otros patrones. Pueden ser predecesores de otros patrones, patrones alternativos con soluciones diferentes para el mismo problema pero con fuerzas y restricciones diferentes, o bien, patrones codependientes que podrían (o deberían) aplicarse simultáneamente [APP97].
Solución:	Describe las relaciones estáticas y las reglas dinámicas que describen cómo lograr el resultado deseado. Es equivalente a dar instrucciones de cómo construir los productos de trabajo necesarios. Puede estar acompañada de dibujos, diagramas y texto que identifiquen la estructura del patrón, sus participantes, y sus colaboradores. La solución puede describir no sólo la estructura del comportamiento estático sino también la dinámica. La descripción de la solución del patrón puede indicar consejos y errores a evitar. Pueden existir variantes o soluciones especializadas [APP97].

3.2.3. DESCRIPCIÓN DE LA TABLA RESUMEN DE LA GUÍA

Actividad	Nombre de la actividad que involucra uno o varios procesos a ser realizados por el rol, para la fase en la que se encuentra.
Procesos del PMBOK	Relación de los procesos con los que se está cumpliendo según la propuesta del PMBOK. Para detallar aún más acerca del contenido y naturaleza de estos procesos, puede revisarse la sección 2.2 de este trabajo.
Actividades de CMM-2	Se relacionan aquí, los compromisos (C), las Actividades (A) las Verificaciones (V), que deben ser cubiertas y satisfechas durante el desarrollo del proyecto, para asegurar que se está cumpliendo con el nivel 2 del modelo de calidad CMM. En cada actividad que señala la guía, se hace el mapeo con los puntos adecuados del CMM para asegurar que se están cubriendo. Podemos ver el detalle de estos puntos del CMM-2 en la sección 2.3 de este trabajo.
Métricas	Se recomiendan las métricas genéricas que pueden aplicarse en la actividad actual. Este punto fue propuesto en la sección 2.4.3 y se hace referencia directa con la figura 2.19, donde se resumen las métricas genéricas. Podemos revisar la bibliografía para ahondar en técnicas más particulares.
Validaciones	Se recomiendan aquí, las pruebas, validaciones y criterios genéricos de aceptación del nivel de calidad de la actividad actual. Este punto fue presentado en la sección 2.4.4 del capítulo anterior, y se hace referencia directa a las figuras 2.22 y 2.23, así como indirectamente a la figura 2.22. También en la bibliografía podemos encontrar información más detallada de alguna técnica en especial.
Entradas, Salidas	Se resumen los elementos que pueden fungir como entregables de la fase. Estos conceptos se presentaron en la sección 2.1.1, y se hace referencia a la propuesta genérica que el PMBOK nos ofrece, para especificar las entradas y salidas de los procesos que intervienen en la actividad actual, vistos en la sección 2.2.2
Métodos, técnicas e Instrumentos	Aquí se engloban otras habilidades de las que se puede hacer uso para complementar, auxiliar y apoyar la actividad actual. El PMBOK también propone en forma general, una serie de elementos de apoyo para sus procesos, presentados en la sección 2.2.2.

CAPÍTULO 4. Fase de Inicio.

Este es el primer capítulo que conforma la guía para la administración de proyectos de software. Abarca la fase de Inicio del Ciclo de Vida que se presentó en la sección 3.1.1. Como ya se indicó en la unidad 3, la guía se presenta usando marcos (ver la sección 3.2.1), patrones (ver la sección 3.2.2) y una tabla resumen (ver la sección 3.2.3). Cada patrón hace referencia a un rol (ver la sección 3.1.2), los cuales están relacionados directamente con las actividades del Administrador Principal, y, en forma gráfica, estos patrones se encuentran detallados mediante un diagrama de actividades (ver el Anexo A).

4.1. Marco para la Fase de Inicio.

Procesos:	De Iniciación y de Planeación.	
Objetivos:	<ul style="list-style-type: none"> • Reconoce formalmente la existencia de un proyecto nuevo • Establece el alcance, compromisos y las limitaciones del proyecto • Establece un modelo básico del negocio y recomendaciones • Estima los recursos, tamaño, costos y el calendario del proyecto • Estima los riesgos potenciales • Genera al menos una arquitectura candidata inicial • Genera el Plan de Desarrollo del Proyecto 	
Compuesta por	<ul style="list-style-type: none"> • El 100 % de la Estrategia • El 30 % de los Requerimientos • El 25 % de las Pruebas • El 25 % de la Documentación 	
Tiempo utilizado	<ul style="list-style-type: none"> • El 10 % del total 	
Factores críticos de Éxito	<ul style="list-style-type: none"> • Participación activa de los ejecutivos claves y líderes de opinión • Corrección temprana de opiniones, ideas y del modelo del negocio • Reuniones exitosas de retroalimentación • Adaptabilidad a cambios imprevistos de circunstancias y objetivos. 	
Roles	<ul style="list-style-type: none"> • Administrador Principal • Administrador del Proyecto de Software (Administrador del Proyecto). • Administrador de Configuraciones • Administrador de Calidad • Líderes de Proyecto 	
Entregables de la Fase		
Entradas	<ul style="list-style-type: none"> • Políticas de la organización. • Plan estratégico global • Información histórica • Directrices e ideas del usuario. 	<ul style="list-style-type: none"> • Descripción general del producto. • Organigramas • Minutas de reuniones. • Descripción de recursos.

	<ul style="list-style-type: none">• Presupuesto y tiempos.	
Salidas:	<ul style="list-style-type: none">• Plan de Desarrollo del Proyecto:<ul style="list-style-type: none">▪ Descripción oficial del producto.▪ Descripción de restricciones.• Descripción de supuestos.▪ Lista de actividades.• Calendario de trabajo.• Asignación de roles y responsabilidades• Equipo de trabajo.• Plan de Abastecimiento.▪ Requerimiento de recursos.• Estimación de costos.	<ul style="list-style-type: none">• Origen y síntomas de riesgos.<ul style="list-style-type: none">▪ Eventos potenciales de riesgo.▪ Riesgos a resolver y a ignorar• Plan de Calidad.• Plan de Configuración.• Cotizaciones.• Presentación visual del Plan de Desarrollo de Software.• Asignación del Administrador del Proyecto.• Plan de Contingencia
Subprocesos:	<ul style="list-style-type: none">• De Iniciación:<ul style="list-style-type: none">▪ 2.1. Iniciación.• De Planeación:<ul style="list-style-type: none">▪ 2.2. Planeación del Alcance.▪ 2.3. Definición del Alcance▪ 3.1. Definición de Actividades.▪ 4.1. Planeación de Recursos.▪ 3.2. Secuencia de Actividades.▪ 3.3. Estimación de Actividades.▪ 4.2. Estimación de Costos.▪ 3.4. Desarrollo del Calendario.▪ 4.3. Presupuestos de Costos.▪ 1.1. Creación del Plan del Proy.• De Ejecución:<ul style="list-style-type: none">▪ 7.2. Distribución de la información▪ 9.3. Cotización	<ul style="list-style-type: none">▪ 5.1. Planeación de la Calidad.▪ 6.1. Planeación de la Organización▪ 6.2. Selección del Personal.▪ 7.1. Planeación de la Comunicación▪ 8.1. Identificación de Riesgos.▪ 8.2. Cuantificar los Riesgos.▪ 8.3. Creación del Plan de Contención▪ 9.1. Planeación del Abastecimiento.▪ 9.2. Planeación de la Cotización.▪ 9.4. Selección del proveedor▪ 2.4. Verificación del alcance
Métricas:	<ul style="list-style-type: none">• Trabajo y Progreso.• Cambios y estabilidad.• Estropicios y modularidad.• MTBF y maduración.	<ul style="list-style-type: none">• Costo presupuesta y gastos.• Trabajo duplicado y adaptabilidad.• Equipo de trabajo y su dinámica.
Criterios de Evaluación:	<ul style="list-style-type: none">• Involucrados• Estimados• Seguimiento	<ul style="list-style-type: none">• Requerimientos• Arquitectura inicial
Verificaciones:	<ul style="list-style-type: none">• Estrategia con individuos externos.• Estrategia con Admón. Principal.• Estrategia con Admón. de Calidad.• Comunicación e Información.• Compromisos y motivación y capacitación	<ul style="list-style-type: none">• Estrategia con Admón. del Proyecto• Estrategia con Administración de la Configuración.• Estrategia con Líderes de Proyecto.• Planeación y estimación.• Modelo del Ciclo de Vida.
Procedimientos Métodos, Técnicas e Instrumentos	<ul style="list-style-type: none">• Reuniones• Reportes de Progreso• Experiencia en toma de decisiones.• Medidas de rendimiento.• Análisis del producto.• Análisis de costo / beneficio.	<ul style="list-style-type: none">• Criterios de selección de proyectos.• Planeación de proyectos.• Compresión de tiempos.• Seguros• Ciclos de Vida del Software• Descomposición.

- Técnicas de costeo
- Identificación de alternativas
- Análisis de riesgos
- Listas de verificación
- Valor monetario esperado
- Análisis, conocimiento y habilidades de los involucrados
- Creación de reportes
- Estimaciones análogas
- Técnicas para cotizar
- Tecnología de comunicaciones
- Prácticas de recursos humanos y reclutamiento
- Uso de ayudas visuales para presentaciones
- Software de administración de proyectos

Patrones aplicables

- La Fase Inicial (*The Initiate Phase*) [AMB98]
- Definición y Validación de Requerimientos Iniciales (*Define and Validate Initial Requirements*) [AMB98]
- Definición de Documentos Administrativos Iniciales (*Define the Initial Management Documents*) [AMB98]
- Justificación del Proyecto (*Justify the Project*) [AMB98]
- Definición de la infraestructura del proyecto (*Define the Project Infrastructure*) [AMB98]
- El Administrador Principal en la fase de Inicio
- El Administrador del Proyecto en la fase de Inicio
- El Administrador de Configuración en la fase de Inicio
- El Administrador de la Calidad en la fase de Inicio
- Los Líderes de Proyecto en la fase de Inicio

Antipatrones

- Muerte por Planeación (*Death by Planning*) [BRO98]
- Mazorca (*Corncock*) [BRO98]
- Violencia Intelectual (*Intellectual Violence*) [BRO98]
- Administración Irracional (*Irrational Management*) [BRO98]
- Humo y Espejos (*Smoke and Mirrors*) [BRO98]

Área clave del CMM

- Planeación del Proyecto de Software
- Administración de Requerimientos
- Aseguramiento de la Calidad del Software
- Administración de la Configuración del Software
- Administración de Subcontratistas
- Supervisión y seguimiento del Proyecto de Software

4.2. Patrones por rol para la Fase de Inicio.

EL ADMINISTRADOR PRINCIPAL EN LA FASE DE INICIO

Nombre del Patrón: **El Administrador Principal en la fase de Inicio**

Problema: Queremos determinar las actividades y responsabilidades del Administrador Principal en la fase de Inicio de un proyecto software.

Contexto: Dentro de una organización cuyo servicio es desarrollar productos nuevos de software o modificar los ya existentes, el nacimiento de un nuevo proyecto, que cumpla con dichos objetivos, pertenece generalmente a un proyecto mayor que responde a las necesidades particulares de la organización. Partiendo de esto, el Administrador Principal es la persona que reconoce la oportunidad de negocio relacionada con el desarrollo de un producto de software, encargándose de conseguir y concretar dichos proyectos y apoyado por los clientes, asegurar las condiciones necesarias para que dichos proyectos se lleven a cabo con éxito. El Administrador Principal sigue las políticas y lineamientos propios de la organización, que a su vez pudo haber ayudado a establecer, y mantiene la relación estratégica con el cliente y con los elementos que controlarán el desarrollo del proyecto de software. Además, este administrador principal, es el responsable de definir y delegar los compromisos referentes al producto de software, hechos con individuos y grupos externos a la organización

Relacionado con:

- La Fase Inicial (*The Initiate Phase*) [AMB98].
- Justificación del Proyecto (*Justify the Project*) [AMB96].
- El Administrador del Proyecto en la fase de Inicio

Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades¹ de la figura 4.1; y las actividades correspondientes están detalladas en la tabla 4.1.

ADMINISTRADOR DEL PROYECTO EN LA FASE DE INICIO

Nombre del Patrón: **El Administrador del Proyecto en la fase de Inicio.**

Problema: Queremos determinar las actividades y responsabilidades del Administrador del Proyecto de Software (Administrador del Proyecto) en la fase de Inicio del desarrollo del Proceso de Software.

Contexto: Una vez designado el Administrador del Proyecto de Software, este debe tomar el control general y absoluto de todo lo referente al Proyecto de Software. En la fase de Inicio, su actuación es primordial puesto que es aquí donde se inicia la definición de todos los lineamientos administrativos y técnicos; y se plantea un esquema inicial del ciclo de vida del Proceso de Software, indicándose las líneas base principales. Además, se espera también la estimación de tiempos, costos y recursos para cada elemento que intervenga en el Proceso de Software, construyéndose con esto, el Plan de Desarrollo del Software, responsabilidad primordial del Administrador del Proyecto.

Relacionado con:

- La Fase Inicial (*The Initiate Phase*) [AMB98].
- Definición y Validación de Requerimientos Iniciales (*Define and Validate Initial Requirements*) [AMB98].
- Definición de Documentos Administrativos Iniciales (*Define the Initial Management Documents*) [AMB98].
- Justificación del Proyecto (*Justify the Project*) [AMB98].

¹ Vea el anexo A.

- Definición de la infraestructura del proyecto (*Define the Project Infrastructure*. [AMB98])
- El Administrador Principal en la fase de Inicio
- El Líder del Proyecto en la fase de Inicio
- El Administrador de la Configuración del Software en la fase de Inicio
- El Administrador de la Calidad en la fase de Inicio

Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades² de la figura 4.2, y las actividades correspondientes están detalladas en la tabla 4.2

EL ADMINISTRADOR DE LA CONFIGURACIÓN DEL SOFTWARE EN LA FASE DE INICIO

Nombre del Patrón	El Administrador de la Configuración del Software en la fase de Inicio.
Problema	Queremos determinar las actividades y responsabilidades del Administrador de la Configuración en la fase de Inicio del desarrollo del Proceso de Software
Contexto	Al momento de iniciar un proyecto de software, ya debe existir o bien debe determinarse al responsable de la planeación y coordinación de las actividades referentes a la administración de la configuración del Proyecto de Software. En la fase de Inicio, su presencia es básica puesto que es cuando se definen todos los lineamientos administrativos y técnicos y cuando se plantea un esquema inicial del ciclo de vida, indicándose las líneas base principales. A partir de este planteamiento, debe construirse el Plan de la Configuración del Software, responsabilidad del Administrador de la Configuración. Además, dependiendo del tamaño del proyecto de software, puede proponer todo un equipo de trabajo para conformar al Grupo de la Administración de la Configuración del Software. Para el caso de esta guía, las actividades del Grupo de la Administración de la Configuración del Software serán realizadas por el propio Administrador de la Configuración.
Relacionado con	<ul style="list-style-type: none"> • La Fase Inicial (<i>The Initiate Phase</i>) [AMB98] • Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98] • Definición de Documentos Administrativos Iniciales (<i>Define the Initial Management Documents</i>) [AMB98] • Justificación del Proyecto (<i>Justify the Project</i>) [AMB98] • Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98] • El Administrador del Proyecto en la fase de Inicio • El Administrador de la Calidad en la fase de Inicio • El Líder del Proyecto en la fase de Inicio
Solución	La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades ³ de la figura 4.3; y las actividades correspondientes están detalladas en la tabla 4.3

EL ADMINISTRADOR DE LA CALIDAD DEL SOFTWARE EN LA FASE DE INICIO

Nombre del Patrón	El Administrador de la Calidad del Software en la fase de Inicio.
Problema	Queremos determinar las actividades y responsabilidades del Administrador de la Calidad en la fase de Inicio del desarrollo del Proceso de Software
Contexto	Al momento de iniciar un proyecto de software, ya debe existir o bien, debe determinarse al responsable de la planeación y coordinación de las actividades referentes al aseguramiento de la calidad del Proyecto de Software. En la fase de

² *idem*

³ *idem*

Inicio este rol también juega un papel importante por que es aquí donde se presenta la estrategia, el tipo y el calendario de revisiones que se llevarán a cabo para garantizar y consolidar que los esfuerzos dentro del proyecto encajan con los estándares y con las expectativas, que tanto el usuario como la organización tienen de dicho proyecto. Desde este punto de vista, la calidad y su aseguramiento son críticos en el éxito de un proyecto y por lo tanto, el planteamiento inicial, conjuntamente con los demás administradores involucrados, forma parte integral de la etapa de Inicio. Dependiendo del tamaño del proyecto de software (en tiempo y en funcionalidad), puede proponer todo un equipo de trabajo para conformar al Grupo de la Administración de la Calidad del Software. Para el caso de esta guía, las actividades del Grupo de la Administración de la Calidad del Software serán realizadas por el propio Administrador de la Calidad.

- Relacionado con
- La Fase Inicial (*The Initiate Phase*) [AMB98]
 - Definición de Documentos Administrativos Iniciales (*Define the Initial Management Documents*) [AMB98].
 - Justificación del Proyecto (*Justify the Project*) [AMB98].
 - El Administrador Principal del Proyecto en la fase de Inicio.
 - El Administrador del Proyecto en la fase de Inicio.
 - El Administrador de la Configuración en la fase de Inicio.
 - El Líder del Proyecto en la fase de Inicio.

Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades⁴ de la figura 4.4; y las actividades correspondientes están detalladas en la tabla 4.4.

EL LÍDER DEL PROYECTO EN LA FASE DE INICIO

Nombre del Patrón: **El Líder del Proyecto en la fase de Inicio.**

Problema: Queremos determinar las actividades y responsabilidades del Líder del Proyecto en la fase de Inicio del desarrollo del Proceso de Software.

Contexto: Al momento de comenzar un proyecto nuevo de software, se plantean los requerimientos iniciales en forma general, y de la misma manera, se desea conocer una o varias posibles soluciones. Es cuando se inicia la creación de un Plan de Desarrollo que debe estar basado totalmente en la naturaleza del problema, tanto en los aspectos de presupuestos y tiempos, como de los aspectos técnicos. Para esto, el Líder del Proyecto o los Líderes del Proyecto, deben trabajar conjuntamente con el Administrador del Proyecto y apoyarlo en la creación de dichas soluciones. Está inmerso en la parte técnica del problema, proveyendo estimados de tiempos y recursos, así de los posibles riesgos que desde la estrategia ya pueden preverse. El Líder trabaja entonces, en actividades que lo relacionan con el GIS, el cliente y el grupo de Administradores del Proyecto.

- Relacionado con
- La Fase Inicial (*The Initiate Phase*) [AMB98].
 - Definición de Documentos Administrativos Iniciales (*Define the Initial Management Documents*) [AMB98].
 - Justificación del Proyecto (*Justify the Project*) [AMB98].
 - El Administrador Principal del Proyecto en la fase de Inicio.
 - El Administrador del Proyecto en la fase de Inicio.
 - El Administrador de la Configuración en la fase de Inicio
 - El Administrador de la Calidad en la fase de Inicio.

Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades⁵ de la figura 4.5; y las actividades correspondientes están detalladas en la tabla 4.5.

⁴ Idem

⁵ Idem

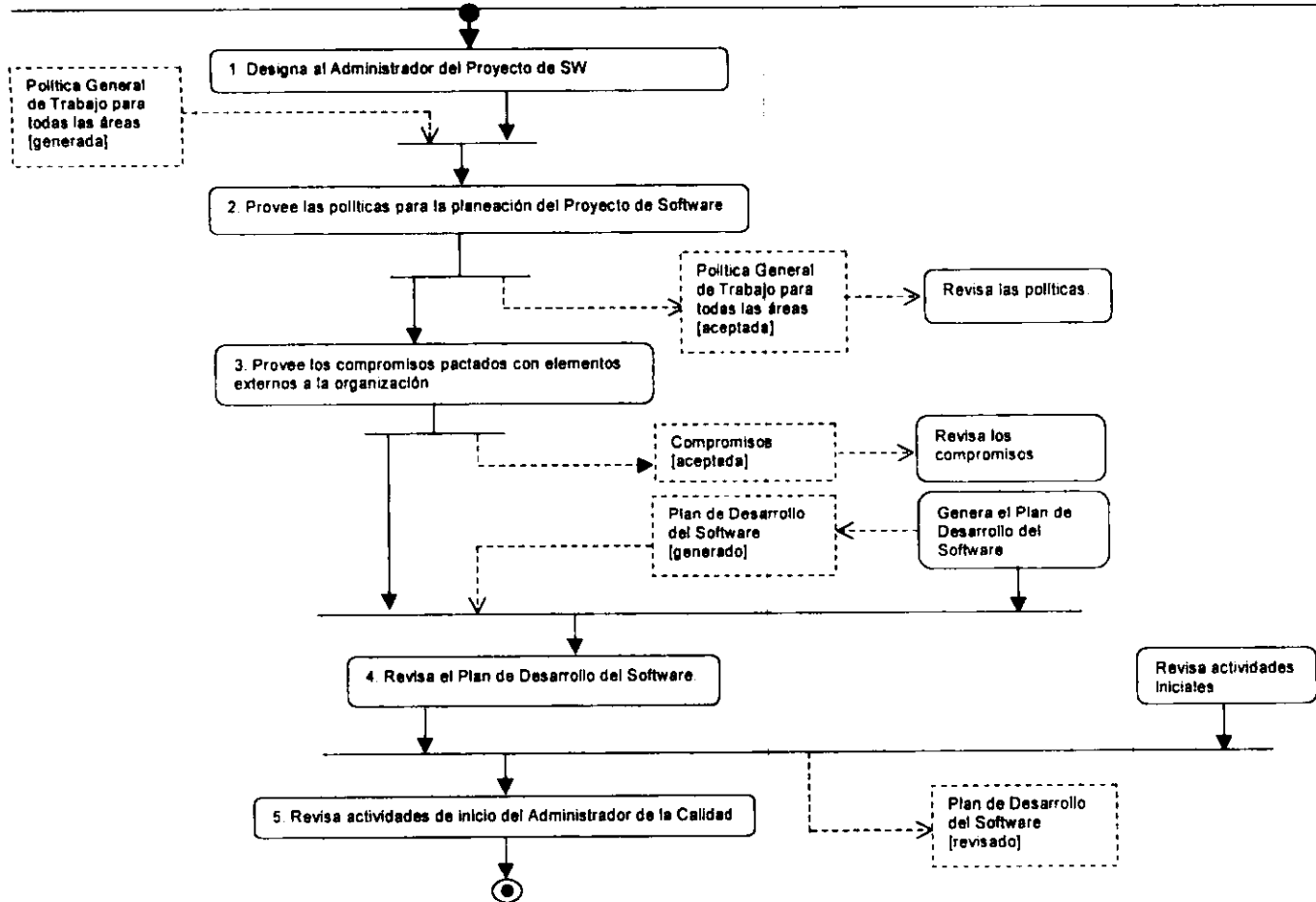


Figura 4.1

Actividades del Administrador Principal en la Fase de Inicio

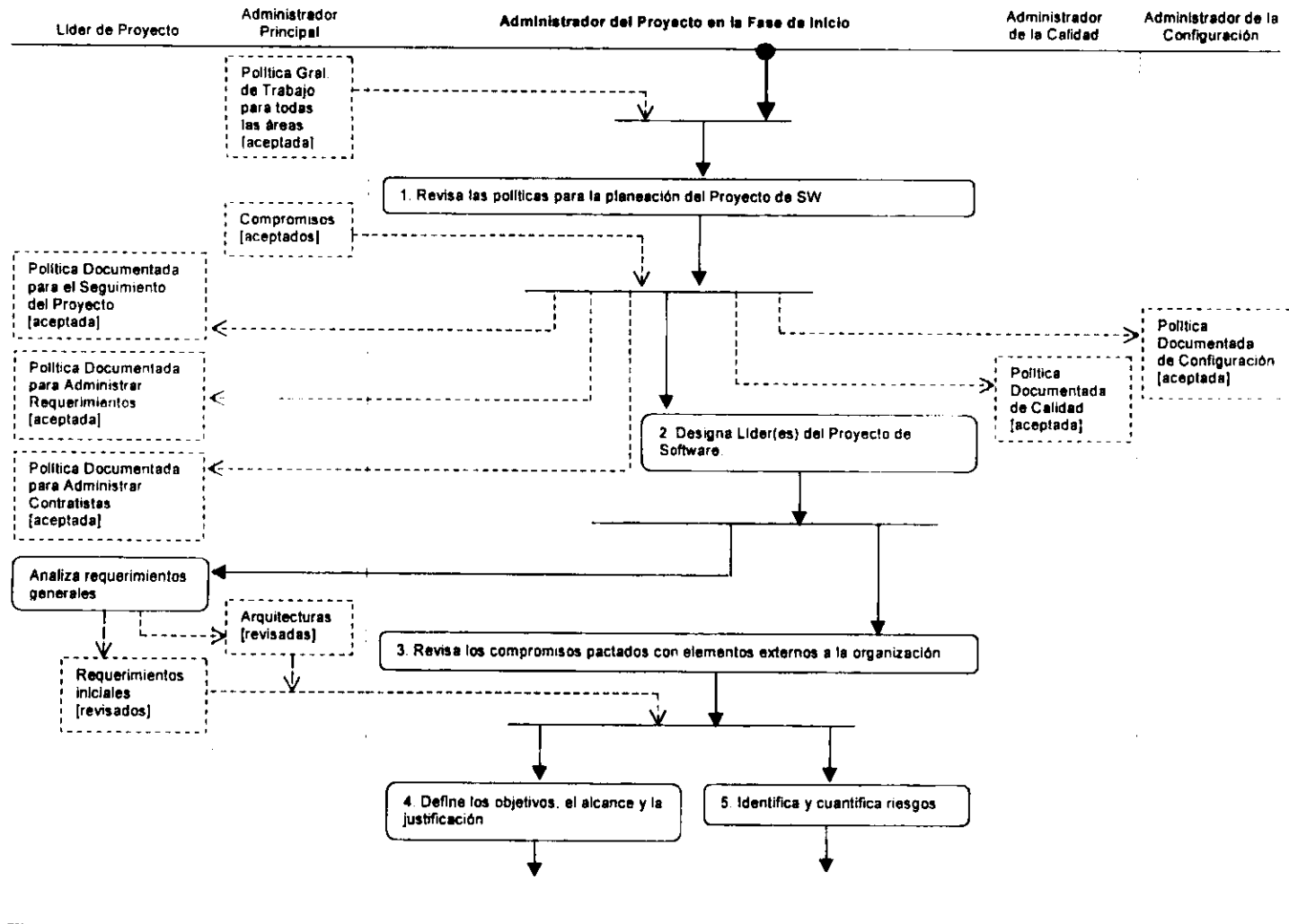
Actividad	Procesos del PMBOK	Métricas ⁶	Entradas	Métodos, técnicas e Instrumentos
	Actividades de CMM-3 ⁷	Validaciones ⁸	Salidas	
1 Designa al Administrador del Proyecto de SW.	<ul style="list-style-type: none"> 2.1 Iniciación. 			<ul style="list-style-type: none"> Experiencia en toma de decisiones. Manejo de personal
	<i>Planeación:</i> <ul style="list-style-type: none"> C1. Designa Administrador del Proyecto. 		<ul style="list-style-type: none"> Asignación del Administrador del Proyecto. 	
2 Provee las políticas para la planeación del Proyecto de Software.	<ul style="list-style-type: none"> 2.1 Iniciación. 		<ul style="list-style-type: none"> Políticas de la organización. 	<ul style="list-style-type: none"> Reuniones.
	<i>Planeación:</i> <ul style="list-style-type: none"> C.2. Seguir una política escrita de la organización para planear el Proyecto de SW. 	<ul style="list-style-type: none"> Estrategia, con la organización. 	<ul style="list-style-type: none"> Políticas de la organización aceptadas. 	
3 Provee los compromisos pactados con elementos externos a la organización.	<ul style="list-style-type: none"> 2.1 Iniciación. 		<ul style="list-style-type: none"> Descripción general del producto. Plan estratégico global. Información histórica. 	<ul style="list-style-type: none"> Reuniones. Criterios de selección de proyectos.
	<i>Planeación:</i> <ul style="list-style-type: none"> A4. Revisa los compromisos con individuos y grupos externos. 	<ul style="list-style-type: none"> Estrategia, con individuos y grupos externos. 	<ul style="list-style-type: none"> Compromisos aceptados. 	
4 Revisa el Plan de Desarrollo del Software.		<ul style="list-style-type: none"> Involucrados 	<ul style="list-style-type: none"> Plan de Desarrollo de Software generado. 	<ul style="list-style-type: none"> Reuniones. Experiencia en toma de decisiones. Reportes de progreso.
	<i>Planeación:</i> <ul style="list-style-type: none"> V1. Revisa periódicamente las actividades de planeación. V2. Revisa periódicamente las actividades de planeación con el Administrador del Proyecto. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. 	<ul style="list-style-type: none"> Plan de Desarrollo de Software revisado. 	
5 Revisa actividades de inicio del Administrador de la Calidad.		<ul style="list-style-type: none"> Trabajo y progreso. Seguimiento 		<ul style="list-style-type: none"> Reuniones. Experiencia en toma de decisiones Reportes de progreso.
	<i>Calidad:</i> <ul style="list-style-type: none"> V1. Revisa periódicamente actividades de calidad. 	<ul style="list-style-type: none"> Estrategia, con el Administrador de Calidad. 		

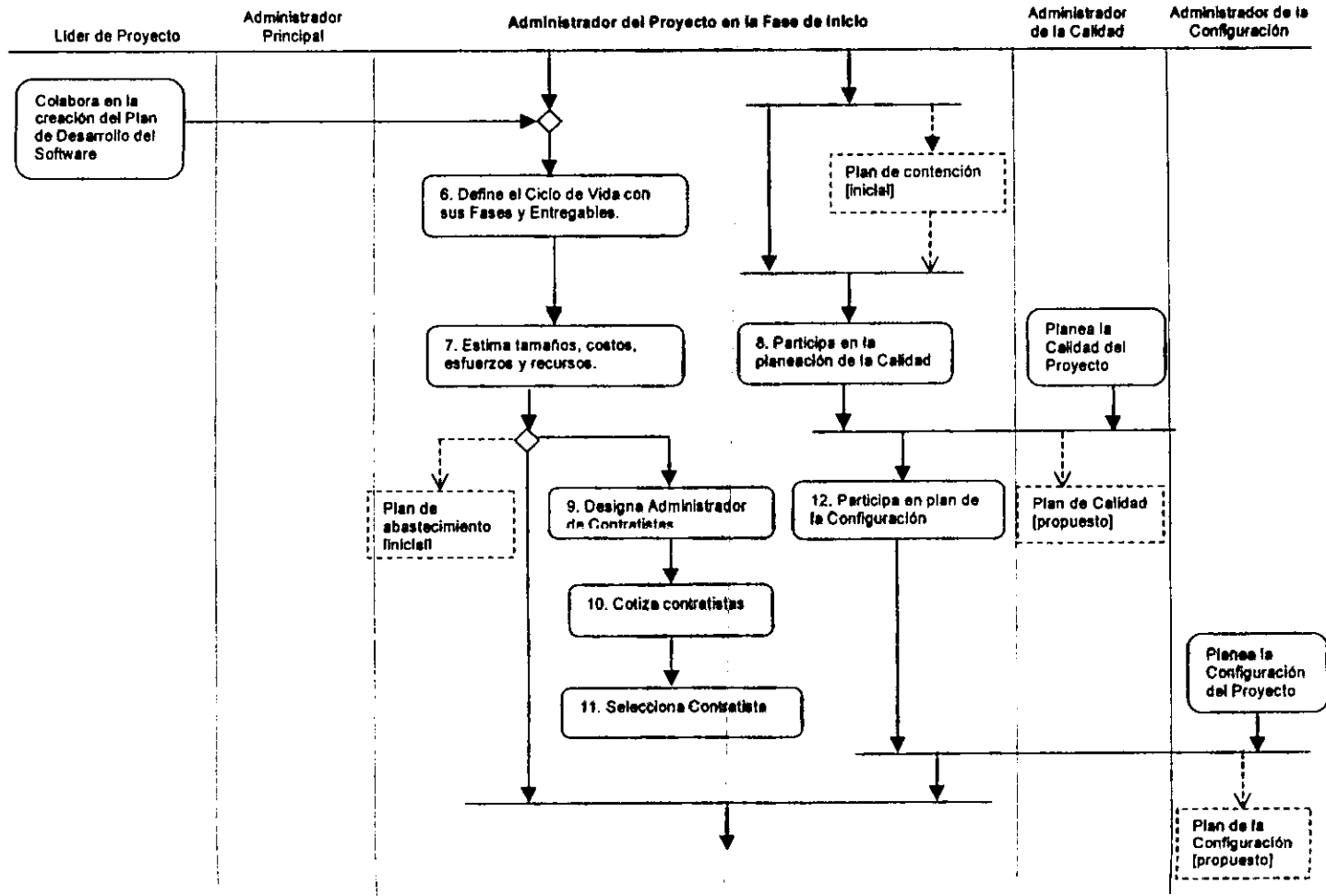
Tabla 4.1

⁶ Ver figura 2.20 y figura 2.24.

⁷ C (compromisos). A (actividades). V (Verificaciones).

⁸ Ver figura 2.23.





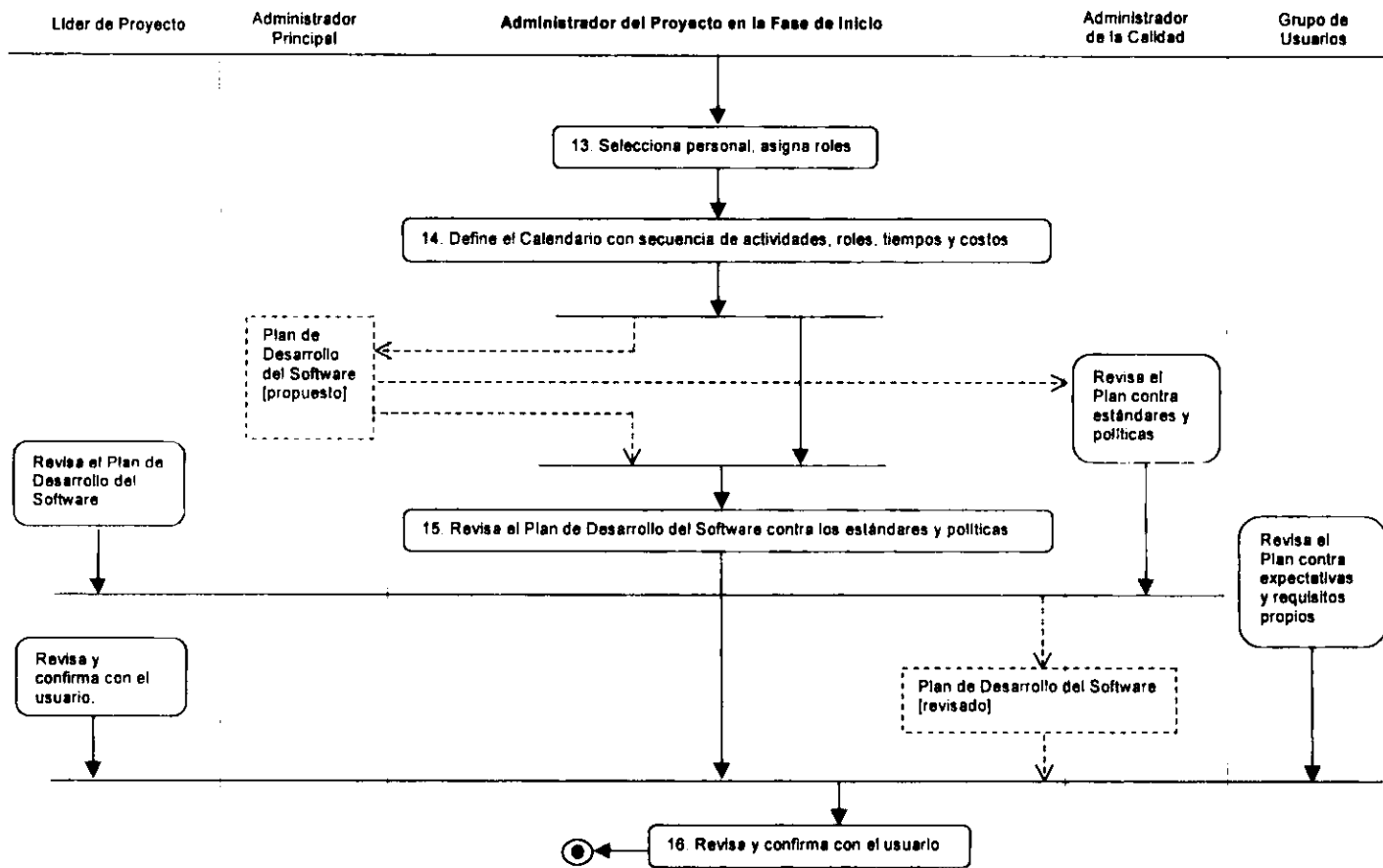


Figura 4.2

Actividades del Administrador del Proyecto en la Fase de Inicio

	Actividad	Procesos del PMBOK	Métricas ⁹	Entradas	Métodos, técnicas e Instrumentos
		Actividades de CMM-2 ¹⁰	Validaciones ¹¹	Salidas	
1	Revisa las políticas para la planeación del Proyecto de SW	<ul style="list-style-type: none"> 2.1 Iniciación. 		<ul style="list-style-type: none"> Política General de Trabajo aceptada para todas las áreas de la organización. 	<ul style="list-style-type: none"> Experiencia en toma de decisiones. Reuniones y entrevistas
	<i>Planeación:</i> <ul style="list-style-type: none"> C2. Seguir una política escrita de la organización para planear el Proyecto de SW. V1. Revisa periódicamente las actividades de planeación junto con el Administrador Principal. 	<ul style="list-style-type: none"> Estrategia, con Administrador Principal. Estrategia, con Administrador de Calidad. Estrategia, con Administrador de la Configuración. Estrategia, con los Líder(es) del Proyecto. 	<ul style="list-style-type: none"> Política documentada y aceptada para la Administración de Calidad. Política documentada y aceptada para la Administración de la Configuración. Política documentada y aceptada para el Seguimiento del Proyecto. Política documentada y aceptada para Administrar Requerimientos. Política documentada y aceptada para Administrar Contratistas. 		
2	Designa Líder(es) del Proyecto de Software.	<ul style="list-style-type: none"> 2.1 Iniciación. 			<ul style="list-style-type: none"> Experiencia en toma de decisiones. Manejo de personal.
	<i>Seguimiento:</i> <ul style="list-style-type: none"> C1. Designa un Administrador del Proyecto de Software. 		<ul style="list-style-type: none"> Asignación del Líder del Proyecto. 		
3	Revisa los compromisos pactados con elementos externos a la organización.	<ul style="list-style-type: none"> 2.1 Iniciación. 	<ul style="list-style-type: none"> Involucrados 	<ul style="list-style-type: none"> Minutas de reuniones. Compromisos aceptados. Conocimiento de los entrevistados Documentos de los entrevistados 	<ul style="list-style-type: none"> Experiencia en toma de decisiones. Reuniones y entrevistas Modelado y consolidación
	<i>Planeación:</i> <ul style="list-style-type: none"> A4. Revisa los compromisos con individuos y grupos externos. 	<ul style="list-style-type: none"> Estrategia, con individuos y grupos externos. 	<ul style="list-style-type: none"> Lista de los entrevistados. Notas consolidadas de entrevistas Modelo y definición inicial del negocio. Ejemplos útiles del negocio 		

⁹ Ver figura 2.20 y figura 2.24.

¹⁰ C (compromisos), A (actividades), V (Verificaciones).

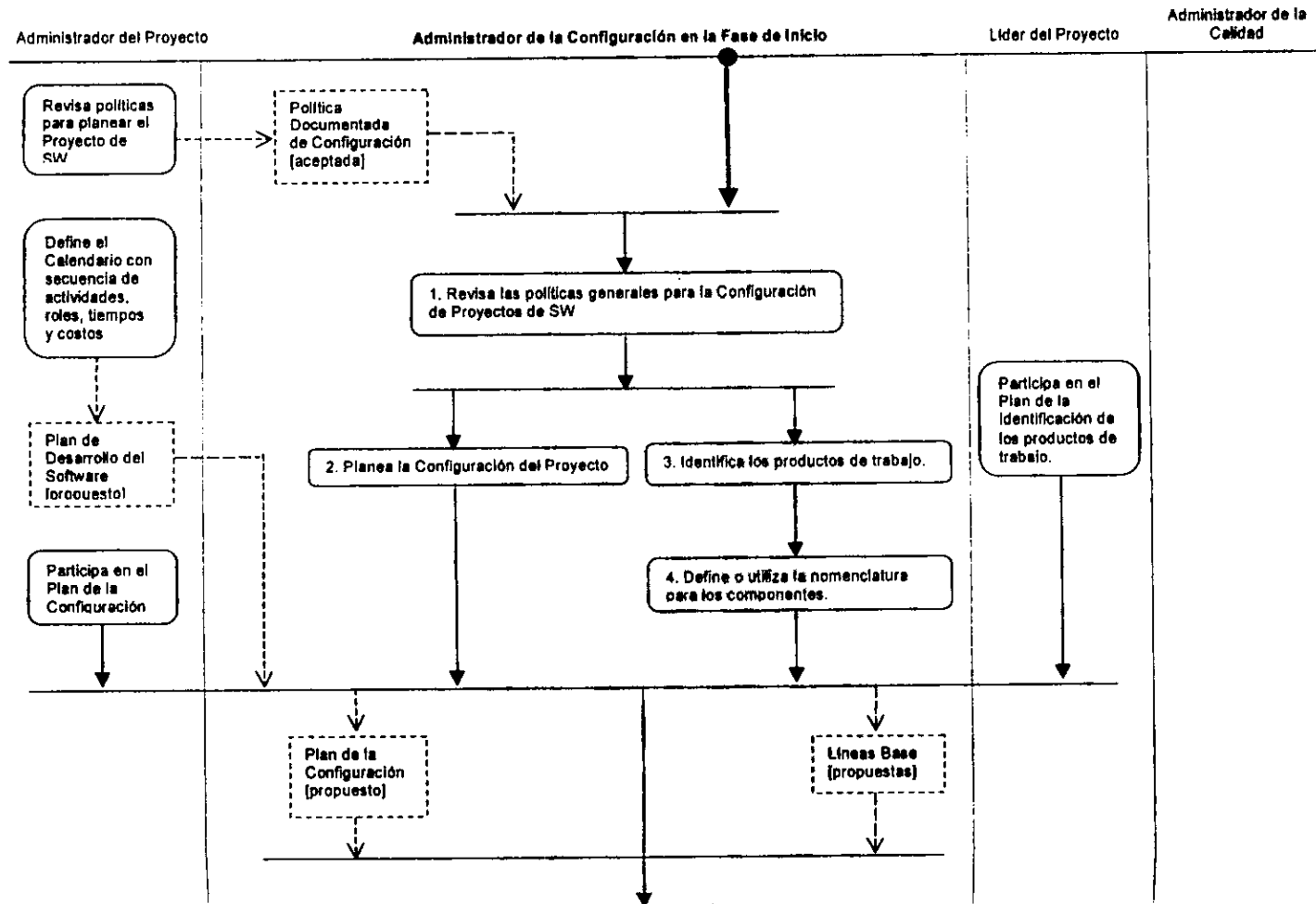
¹¹ Ver figura 2.23.

4	Define los objetivos el alcance y la justificación.	<ul style="list-style-type: none"> 2.2 Planeación del Alcance. 2.3 Definición del Alcance. 	<ul style="list-style-type: none"> Estimados. Involucrados. 	<ul style="list-style-type: none"> Descripción general del producto Directrices e ideas del usuario. Minutas de reuniones. Plan estratégico global. Arquitecturas y requerimientos revisados 	<ul style="list-style-type: none"> Reuniones. Experiencia en toma de decisiones. Medidas de rendimiento Análisis del producto. Análisis de costo / beneficio. Identificación de alternativas. Estudio de factibilidad
		<i>Planeación</i> <ul style="list-style-type: none"> A2. La planeación debe iniciar en las primeras etapas 	<ul style="list-style-type: none"> Planeación y estimación 	<ul style="list-style-type: none"> Compromisos aceptados. Objetivos, alcance y justificación. Descripción del producto y restricciones Descripción de supuestos Términos de referencia estratégicos Límites del proyecto con otros sistemas Reportes optativos de factibilidad 	
5	Identifica y cuantifica Riesgos	<ul style="list-style-type: none"> 8.1 Identificación de Riesgos. 8.2 Cuantificación de Riesgos. 8.3 Genera Plan de Contención. 	<ul style="list-style-type: none"> Estropicios y modularidad Trabajo duplicado y adaptabilidad. Estimados 	<ul style="list-style-type: none"> Minutas de reuniones. Arquitecturas revisadas Requerimientos iniciales revisados 	<ul style="list-style-type: none"> Reuniones. Análisis de riesgos. Listas de verificación. Valor monetario esperado Experiencia en toma de decisiones. Seguros. Control de proyectos
		<i>Planeación</i> <ul style="list-style-type: none"> A13. Identificar medir y documentar riesgos del SW 	<ul style="list-style-type: none"> Comunicación e Información 	<ul style="list-style-type: none"> Descripción de restricciones y supuestos Origen y síntomas de riesgos. Eventos potenciales de riesgo Riesgos a resolver y a ignorar. Plan de contención. 	
6	Define el Ciclo de Vida con sus Fases y Entregables	<ul style="list-style-type: none"> 3.1 Definición de Actividades. 3.3 Estimación de Actividades. 	<ul style="list-style-type: none"> Trabajo y progreso. Requerimientos 	<ul style="list-style-type: none"> Minutas de reuniones. Directrices e Ideas del usuario principal. 	<ul style="list-style-type: none"> Reuniones. Ciclos de Vida del Software. Descomposición. Planeación de proyectos Calendarización
		<i>Planeación</i> <ul style="list-style-type: none"> A5. Identifica el Ciclo de Vida de SW A8. Identifica los productos del SW 	<ul style="list-style-type: none"> Modelo del Ciclo de Vida 	<ul style="list-style-type: none"> Fases, entregables y actividades. Descripción funcional y de Información Glosario Inicial de términos 	
7	Estima tamaños, costos, esfuerzos y recursos.	<ul style="list-style-type: none"> 4.1 Planeación de Recursos. 4.2 Estimación de Costos. 9.1 Planeación del Abastecimiento. 	<ul style="list-style-type: none"> Costo presupuestado y gastos. Estimados 	<ul style="list-style-type: none"> Minutas de reuniones. Presupuestos, organigramas y tiempos. Descripción de recursos Descripción funcional y de información Glosario Inicial de términos Datos históricos y estimaciones de proyectos similares 	<ul style="list-style-type: none"> Reuniones. Técnicas de costeo. Experiencia en toma de decisiones. Identificación de alternativas. Estimaciones análogas.
		<i>Planeación</i> <ul style="list-style-type: none"> A9. Estima tamaños de productos A10 Estima costos y esfuerzos A11. Estima recursos. 	<ul style="list-style-type: none"> Planeación y estimación Comunicación e Información. Compromisos, motivación y capacitación 	<ul style="list-style-type: none"> Plan de Abastecimiento. Requerimiento de recursos Estimación de costos. 	

8	Participa en la planeación de la Calidad	<ul style="list-style-type: none"> 5.1. Planeación de la Calidad. 		<ul style="list-style-type: none"> Plan de Calidad inicial. 	<ul style="list-style-type: none"> Reuniones.
			<ul style="list-style-type: none"> Comunicación e Información. 	<ul style="list-style-type: none"> Plan de Calidad aceptado. 	
9	Designa Administrador de Contratistas	<ul style="list-style-type: none"> 9.2 Planeación de la cotización. 			<ul style="list-style-type: none"> Experiencia en toma de decisiones. Manejo de personal
		<p>Subcontratistas:</p> <ul style="list-style-type: none"> C2. Designa un Administrador de Subcontratistas. 		<ul style="list-style-type: none"> Designación del Administrador de Contratistas. 	
10	Cotiza contratistas	<ul style="list-style-type: none"> 9.2 Planeación de la Cotización. 9.3 Cotización. 		<ul style="list-style-type: none"> Minutas de reuniones. Perfiles y currícula de contratistas. 	<ul style="list-style-type: none"> Reuniones. Técnicas para cotizar. Experiencia en toma de decisiones.
			<ul style="list-style-type: none"> Comunicación e Información. 	<ul style="list-style-type: none"> Cotización. 	
11	Selecciona Contratista.	<ul style="list-style-type: none"> 9.4 Selección del Proveedor. 		<ul style="list-style-type: none"> Minutas de reuniones. Perfiles y currícula de contratistas. 	<ul style="list-style-type: none"> Reuniones. Técnicas para cotizar. Experiencia en toma de decisiones.
		<p>Subcontratistas:</p> <ul style="list-style-type: none"> A2. Selecciona al contratista según la evaluación y procedimientos. 	<ul style="list-style-type: none"> Comunicación e Información. 	<ul style="list-style-type: none"> Selección del Contratista. 	
12	Participa en plan de la Configuración	<ul style="list-style-type: none"> 7.1. Planeación de la Comunicación. 		<ul style="list-style-type: none"> Requerimientos de comunicación. Minutas de reuniones. Plan de Configuración inicial. 	<ul style="list-style-type: none"> Reuniones. Tecnología de comunicaciones.
			<ul style="list-style-type: none"> Comunicación e Información. 	<ul style="list-style-type: none"> Plan de Configuración aceptado. 	
13	Selecciona personal, asigna roles	<ul style="list-style-type: none"> 6.1 Planeación de la organización. 6.2 Selección del personal. 	<ul style="list-style-type: none"> Equipo de trabajo y su dinámica. 	<ul style="list-style-type: none"> Minutas de reuniones. Roles y organigramas. 	<ul style="list-style-type: none"> Prácticas de recurso humano y reclutamiento. Análisis, conocimiento y habilidades de los involucrados.
			<ul style="list-style-type: none"> Compromisos, motivación y capacitación. 	<ul style="list-style-type: none"> Asignación de roles y responsabilidades. Diagrama de la organización. Equipo de trabajo. 	
14	Define el Calendario con secuencia de actividades, roles, tiempos y costos	<ul style="list-style-type: none"> 3.2. Secuencia de Actividades. 3.4. Desarrollo del Calendario. 4.3. Presupuestos de Costos. 1.1. Creación del Plan de Desarrollo del SW. 	<ul style="list-style-type: none"> Trabajo y progreso. Estimados Requerimientos Involucrados 	<ul style="list-style-type: none"> Minutas de reuniones. Compromisos y obligaciones. Presupuestos y tiempos. Organigramas. 	<ul style="list-style-type: none"> Reuniones. Planeación de proyectos. Comprensión de tiempos. Uso de ayudas visuales para presentaciones. Creación de reportes. Software de administración de proyectos.
		<p>Planeación:</p> <ul style="list-style-type: none"> A6. Crea Plan de Desarrollo de SW A7. Documenta el PDS. A12. Crea el calendario del PDS. 	<ul style="list-style-type: none"> Planeación y estimación 	<ul style="list-style-type: none"> Presentación visual del Plan. Calendario de trabajo. Plan de Desarrollo del Software. 	

15	Revisa y presenta el Plan de Desarrollo del Software contra los estándares y políticas	<ul style="list-style-type: none"> 1.1. Creación del Plan de Desarrollo del Proyecto. 	<ul style="list-style-type: none"> MTBF y maduración. Seguimiento Requerimientos 	<ul style="list-style-type: none"> Plan de Desarrollo del Software Propuesto. Estándares y políticas de la organización. Minutas de reuniones. Compromisos y obligaciones 	<ul style="list-style-type: none"> Reuniones. Técnicas de evaluación y revisión Procesos y cuestionarios para el aseguramiento de la calidad Ayudas visuales Reportes
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> A15. Registra los datos de la planeación del Software V3 Calidad revisa, audita y reporta las actividades y productos de la planeación del Proyecto de Software <p><i>Subcontratistas:</i></p> <ul style="list-style-type: none"> V3 Calidad revisa, audita y reporta las actividades y productos de la Administración de Subcontratistas 	<ul style="list-style-type: none"> Estrategia Planeación y Estimación Comunicación e información 	<ul style="list-style-type: none"> Plan de Desarrollo del Software Revisado 	
16	Revisa y confirma con el usuario.	<ul style="list-style-type: none"> 2.4 Verificación del alcance. 	<ul style="list-style-type: none"> MTBF y maduración. Involucrados Seguimiento Requerimientos 	<ul style="list-style-type: none"> Plan de Desarrollo del Software Revisado 	<ul style="list-style-type: none"> Reuniones. Técnicas de entrevistas en grupos. Técnicas de presentación. Ayudas visuales Presentaciones estructuradas Reportes
		<p><i>Seguimiento</i></p> <ul style="list-style-type: none"> A3 La Administración principal revisará los compromisos y los cambios del PDS acordados con grupos externos 	<ul style="list-style-type: none"> Estrategia Planeación y Estimación Comunicación e información 	<ul style="list-style-type: none"> Plan de Desarrollo del Software Aceptado por el usuario Cambios a los modelos del Plan de Desarrollo 	

Tabla 4.2.



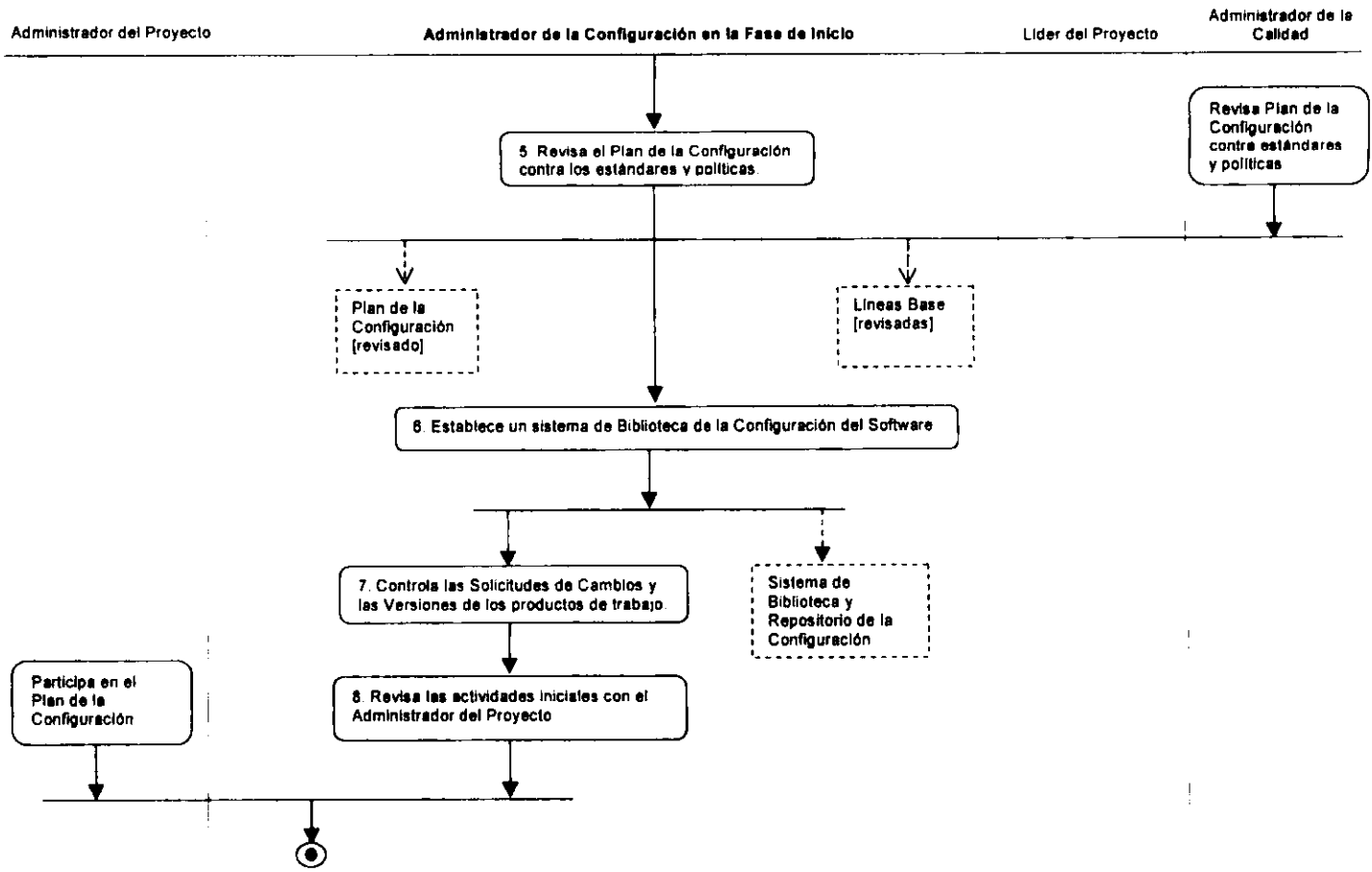


Figura 4.3

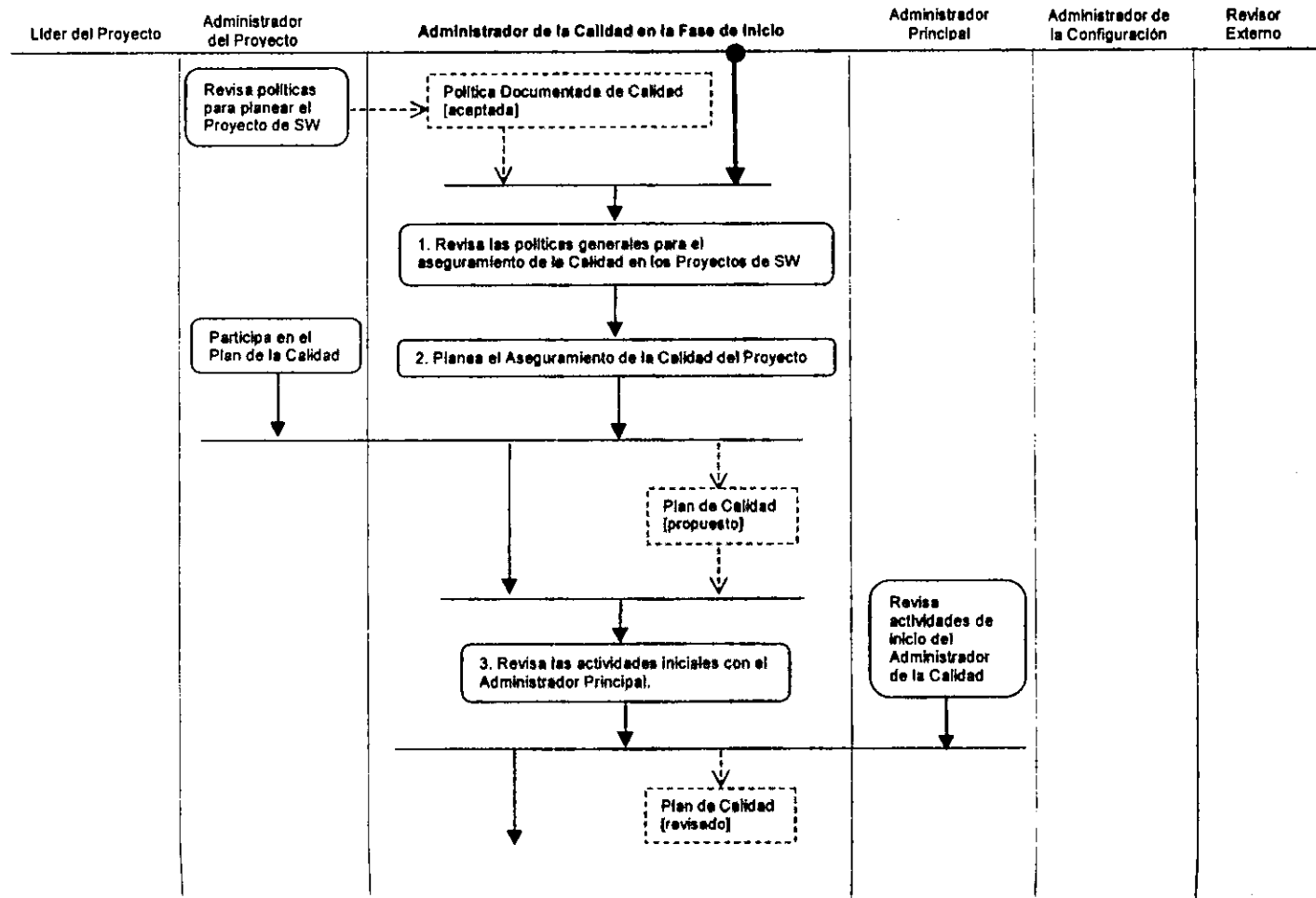
Actividades del Administrador de la Configuración en la Fase de Inicio

	Actividad	Procesos del PMBOK	Métricas ¹²	Entradas	Métodos, Métricas e Instrumentos
		Actividades de CMM-2 ¹³	Validaciones ¹⁴	Salidas	
1	Revisa las políticas generales para la Configuración de Proyectos de SW	<ul style="list-style-type: none"> 2.1 Iniciación. 		<ul style="list-style-type: none"> Política documentada y aceptada de la Configuración. 	<ul style="list-style-type: none"> Reuniones.
		<i>Configuración:</i> <ul style="list-style-type: none"> C1. Sigue una política escrita de la organización para la implantación de la Administración de la Configuración. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. 		
2	Planea la Configuración del Proyecto.	<ul style="list-style-type: none"> 7.1 Planeación de la Comunicación. 7.2 Distribución de la Información. 	<ul style="list-style-type: none"> Costo presupuestado y gastos. Cambios y estabilidad. 	<ul style="list-style-type: none"> Plan de Desarrollo del Software. Requerimientos de comunicación. Tecnología de comunicaciones. Descripción de restricciones. Descripción de supuestos. 	<ul style="list-style-type: none"> Experiencia en toma de decisiones. Reuniones. Experiencia en comunicación.
		<i>Configuración:</i> <ul style="list-style-type: none"> A1. Prepara un Plan de la Configuración del Software 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. Estrategia, con Líder(es) Planeación y Estimación. 	<ul style="list-style-type: none"> Plan propuesto de la Configuración del Software. Líneas Base propuestas. 	
3	Identifica los productos de trabajo.	<ul style="list-style-type: none"> 3.1 Definición de Actividades. 	<ul style="list-style-type: none"> Trabajo y progreso. Requerimientos 	<ul style="list-style-type: none"> Plan de Desarrollo del Software. Descripción de restricciones. Descripción de supuestos. 	<ul style="list-style-type: none"> Reuniones.
		<i>Configuración:</i> <ul style="list-style-type: none"> A4. Identifica productos de trabajo del SW. 	<ul style="list-style-type: none"> Modelo de Ciclo de Vida. 	<ul style="list-style-type: none"> Líneas Base propuestas. 	
4	Define o utiliza la nomenclatura para los componentes.	<ul style="list-style-type: none"> 3.1 Definición de Actividades. 	<ul style="list-style-type: none"> Trabajo y progreso. 	<ul style="list-style-type: none"> Plan de Desarrollo del Software. 	<ul style="list-style-type: none"> Administración de la configuración.
		<i>Configuración:</i> <ul style="list-style-type: none"> A4. Identifica productos de trabajo del SW. 	<ul style="list-style-type: none"> Comunicación e Información 	<ul style="list-style-type: none"> Nomenclatura de Líneas Bases propuestas. 	

¹² Ver figura 2.20 y figura 2.24.¹³ C (compromisos). A (actividades), V (Verificaciones).¹⁴ Ver figura 2.23.

5	Revisa el Plan de la Configuración contra los estándares y políticas	<ul style="list-style-type: none"> 7.1 Planeación de la Comunicación. 	<ul style="list-style-type: none"> MTBF y maduración. Seguimiento 	<ul style="list-style-type: none"> Plan propuesto de la Configuración. Lineas Base propuestas Estándares y políticas. Minutas de reuniones. Compromisos y obligaciones. 	<ul style="list-style-type: none"> Reuniones. Técnicas de evaluación y revisión Procesos y cuestionarios para el aseguramiento de la calidad
		<i>Configuración</i> <ul style="list-style-type: none"> V1 Revisa las actividades junto con el Administrador del Principal. V2 Revisa las actividades junto con el Administrador del Proyecto. V4 Calidad revisa, audita y reporta sus actividades y productos. 	<ul style="list-style-type: none"> Estrategia, con Administrador de la Calidad 	<ul style="list-style-type: none"> Plan revisado de la Configuración del Software Lineas Base revisadas 	
6	Establece un sistema de Biblioteca de la Configuración del Software	<ul style="list-style-type: none"> 7.2 Distribución de la información. 	<ul style="list-style-type: none"> Cambios y estabilidad 	<ul style="list-style-type: none"> Plan de Desarrollo del Software Información histórica. 	<ul style="list-style-type: none"> Experiencia en comunicación. Sistemas de obtención de información. Sistemas de distribución de información.
		<i>Configuración</i> <ul style="list-style-type: none"> A3 Establece un sistema de Biblioteca de la Configuración del Software. A7. Crea los productos de las Bibliotecas. 	<ul style="list-style-type: none"> Comunicación e Información. 	<ul style="list-style-type: none"> Sistema de Biblioteca y Repositorio de la Configuración 	
7	Controla las Solicitudes de Cambios y las Versiones de los productos de trabajo	<ul style="list-style-type: none"> 1.3 Control integral de cambios 7.3 Reporte del Seguimiento. 2.5 Control de cambios al alcance. 3.5 Control del Calendario 	<ul style="list-style-type: none"> Trabajo y progreso Cambios y estabilidad 	<ul style="list-style-type: none"> Plan de Desarrollo del Software Resultados del trabajo Solicitud de cambios. Sistemas de control de cambios 	<ul style="list-style-type: none"> Administración de la configuración. Revisión del seguimiento Reportes de progreso Medidas de rendimiento Planeación adicional
		<i>Configuración</i> <ul style="list-style-type: none"> A5. Supervisa los cambios y problemas reportados de elementos y unidades de la configuración. A7 Controla las versiones de productos en las Bibliotecas de las Lineas base A8. Registra el estado de los elementos y unidades de la configuración. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. Comunicación e Información 	<ul style="list-style-type: none"> Actualización del Plan de Desarrollo del Software Acciones correctivas, de actualización y modificación 	
8	Revisa las actividades Iniciales con el Administrador del Proyecto	<ul style="list-style-type: none"> 7.3 Reporte del Seguimiento 	<ul style="list-style-type: none"> Trabajo y progreso 	<ul style="list-style-type: none"> Resultados del trabajo 	<ul style="list-style-type: none"> Reuniones Reportes de progreso
		<i>Configuración</i> <ul style="list-style-type: none"> V1. Revisa periódicamente sus actividades con el Admón. Principal. V2. Revisa periódicamente sus actividades con Admón. del Proyecto A9. Genera y provee reportes del estado de la Configuración. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. 		

Tabla 4.3



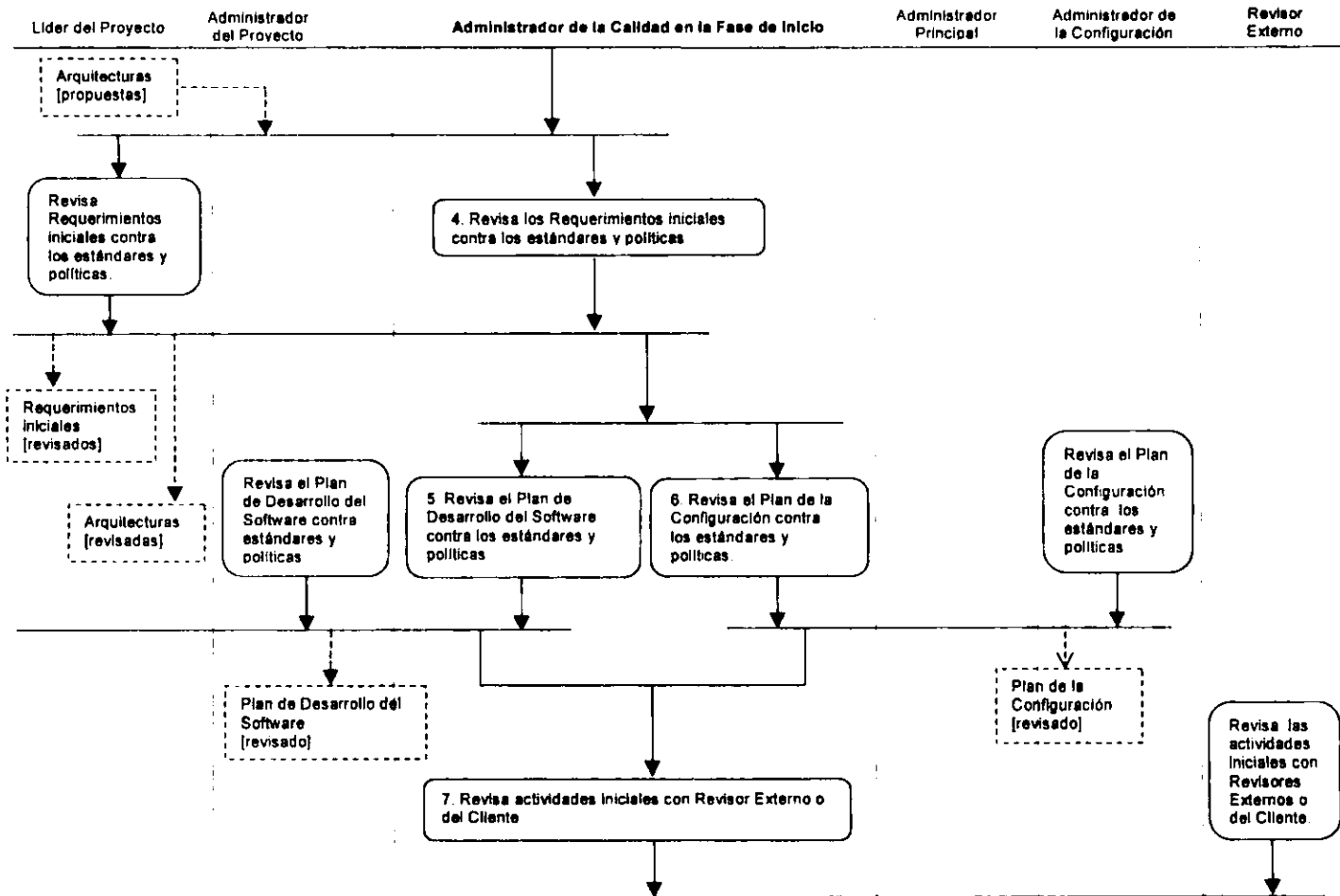


Figura 4.4

Actividades del Administrador de la Calidad en la Fase de Inicio

Actividad	Procesos del PMBOK	Méritos ¹⁵	Entradas	Méritos de Uso
	Actividades de Calidad	Verificaciones ¹⁶	Salidas	
1 Revisa las políticas generales para el aseguramiento de la Calidad en los Proyectos de SW	<ul style="list-style-type: none"> • 2.1 Iniciación. <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • C1. Sigue una política escrita de la organización para la implantación de la Administración de la Calidad. • A3. Participa en la preparación y revisión de estándares y procedimientos. 	<ul style="list-style-type: none"> • Estrategia, con el Administrador del Proyecto. 	<ul style="list-style-type: none"> • Política documentada y aceptada de la Calidad. 	<ul style="list-style-type: none"> • Reuniones.
2 Planea el Aseguramiento de la Calidad del Proyecto	<ul style="list-style-type: none"> • 5.1 Planeación de la Calidad. <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • A1. Prepara un Plan de la Administración de la Calidad del Software. 	<ul style="list-style-type: none"> • Estrategia, con el Administrador del Proyecto. • Planeación y Estimación. 	<ul style="list-style-type: none"> • Estándares y Políticas aceptadas. • Plan propuesto de la Calidad del Software. 	<ul style="list-style-type: none"> • Experiencia en toma de decisiones. • Reuniones. • Experiencia en mejora de procesos.
3 Revisa las actividades iniciales con el Administrador Principal.	<ul style="list-style-type: none"> • 5.1 Planeación de la Calidad. • 5.2. Aseguramiento de la Calidad. <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • V1. Revisa las actividades junto con el Administrador Principal. 	<ul style="list-style-type: none"> • Trabajo y progreso. • Seguimiento <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • Estrategia, con el Administrador Principal. • Planeación y estimación. 	<ul style="list-style-type: none"> • Plan de Calidad propuesto. • Estándares y Políticas aceptadas. <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • Plan de Calidad revisado. 	<ul style="list-style-type: none"> • Experiencia en toma de decisiones. • Reuniones.
4 Revisa los Requerimientos iniciales contra los estándares y políticas.	<ul style="list-style-type: none"> • 3.1 Definición de Actividades. • 5.2. Aseguramiento de la Calidad. <p><i>Requerimientos:</i></p> <ul style="list-style-type: none"> • V3. Calidad revisa, audita y reporta las actividades de la Administración de Requerimientos 	<ul style="list-style-type: none"> • Trabajo y progreso. • Seguimiento <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • Estrategia, con el Líder del Proyecto. 	<ul style="list-style-type: none"> • Requerimientos iniciales. • Arquitecturas propuestas. • Estándares y Políticas aceptadas. <p><i>Calidad:</i></p> <ul style="list-style-type: none"> • Requerimientos iniciales revisados. • Arquitecturas revisadas. 	<ul style="list-style-type: none"> • Técnicas de evaluación y revisión. • Procesos y cuestionarios para el aseguramiento de la calidad. • Reuniones.

¹⁵ Ver figura 2.20 y figura 2.24.

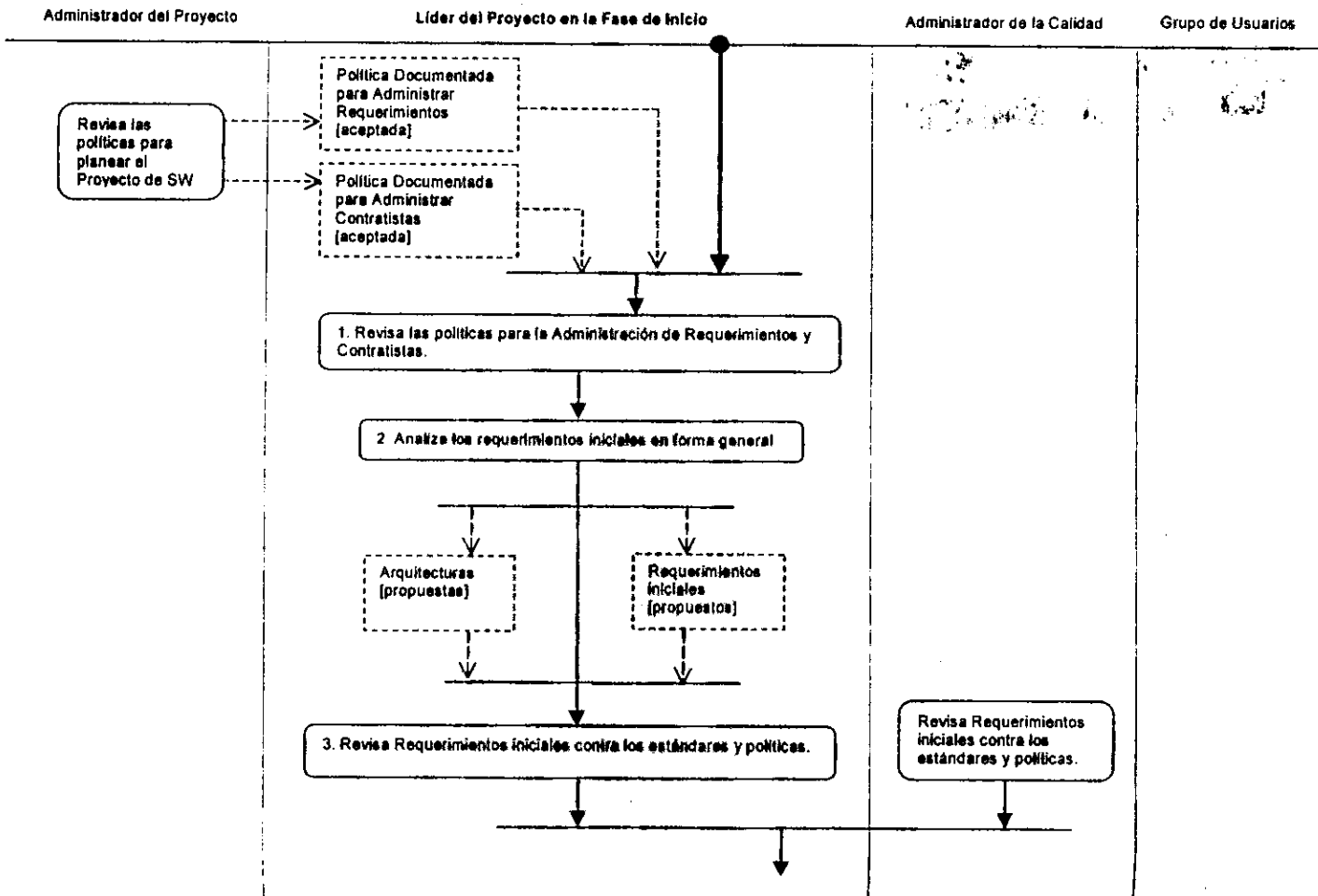
¹⁶ C (compromisos). A (actividades), V (Verificaciones).

¹⁷ Ver figura 2.23

5	Revisa el Plan de Desarrollo del Software contra los estándares y políticas	<ul style="list-style-type: none"> 1.1. Crea el Plan del Proyecto. 5.2. Aseguramiento de la Calidad 	<ul style="list-style-type: none"> Trabajo y progreso. Seguimiento 	<ul style="list-style-type: none"> Plan de Desarrollo del Software propuesto. Estándares y Políticas aceptadas 	<ul style="list-style-type: none"> Técnicas de evaluación y revisión. Procesos y cuestionarios para el aseguramiento de la calidad Reuniones
		<p><i>Calidad</i></p> <ul style="list-style-type: none"> A3 Participa en la preparación y revisión del Plan de Desarrollo del Software. V2 Revisa las actividades junto con el Administrador del Proyecto. <p><i>Planeación</i></p> <ul style="list-style-type: none"> V3 Calidad revisa, audita y reporta las actividades y productos de la Planeación <p><i>Subcontratistas</i></p> <ul style="list-style-type: none"> V3. Calidad revisa, audita y reporta las actividades y productos de la Administración de Subcontratistas. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto Planeación y estimación 	<ul style="list-style-type: none"> Plan de Desarrollo del Software revisado 	
6	Revisa el Plan de la Configuración contra los estándares y políticas	<ul style="list-style-type: none"> 7.1 Planeación de la Comunicación. 5.2. Aseguramiento de la Calidad. 	<ul style="list-style-type: none"> Trabajo y progreso. Seguimiento 	<ul style="list-style-type: none"> Plan propuesto de la Configuración del Software Líneas Base propuestas Estándares y Políticas aceptadas 	<ul style="list-style-type: none"> Reuniones. Técnicas de evaluación y revisión. Procesos y cuestionarios para el aseguramiento de la calidad
		<p><i>Configuración</i></p> <ul style="list-style-type: none"> V4. Calidad revisa, audita y reporta las actividades y productos de la Administración de la Configuración 	<ul style="list-style-type: none"> Estrategia, con el Administrador de la Configuración 	<ul style="list-style-type: none"> Plan revisado de la Configuración del Software. Líneas Base revisadas. 	
7	Revisa actividades iniciales con Revisor Externo o del Cliente.	<ul style="list-style-type: none"> 5.2. Aseguramiento de la Calidad. 	<ul style="list-style-type: none"> Trabajo y progreso. Seguimiento 	<ul style="list-style-type: none"> Plan de Calidad del Proyecto de Software sin aprobar. 	<ul style="list-style-type: none"> Reuniones. Técnicas de evaluación y revisión. Procesos y cuestionarios para el aseguramiento de la calidad
		<p><i>Calidad:</i></p> <ul style="list-style-type: none"> V3 Un grupo independiente de expertos revisa las actividades y productos de trabajo de la Administración de la Calidad. 	<ul style="list-style-type: none"> Estrategia, con el Líder del Proyecto 	<ul style="list-style-type: none"> Plan de Calidad del Proyecto de Software aprobado. 	

Tabla 4.4

SALON DE LA BIBLIOTECA
 1993 DE 1993



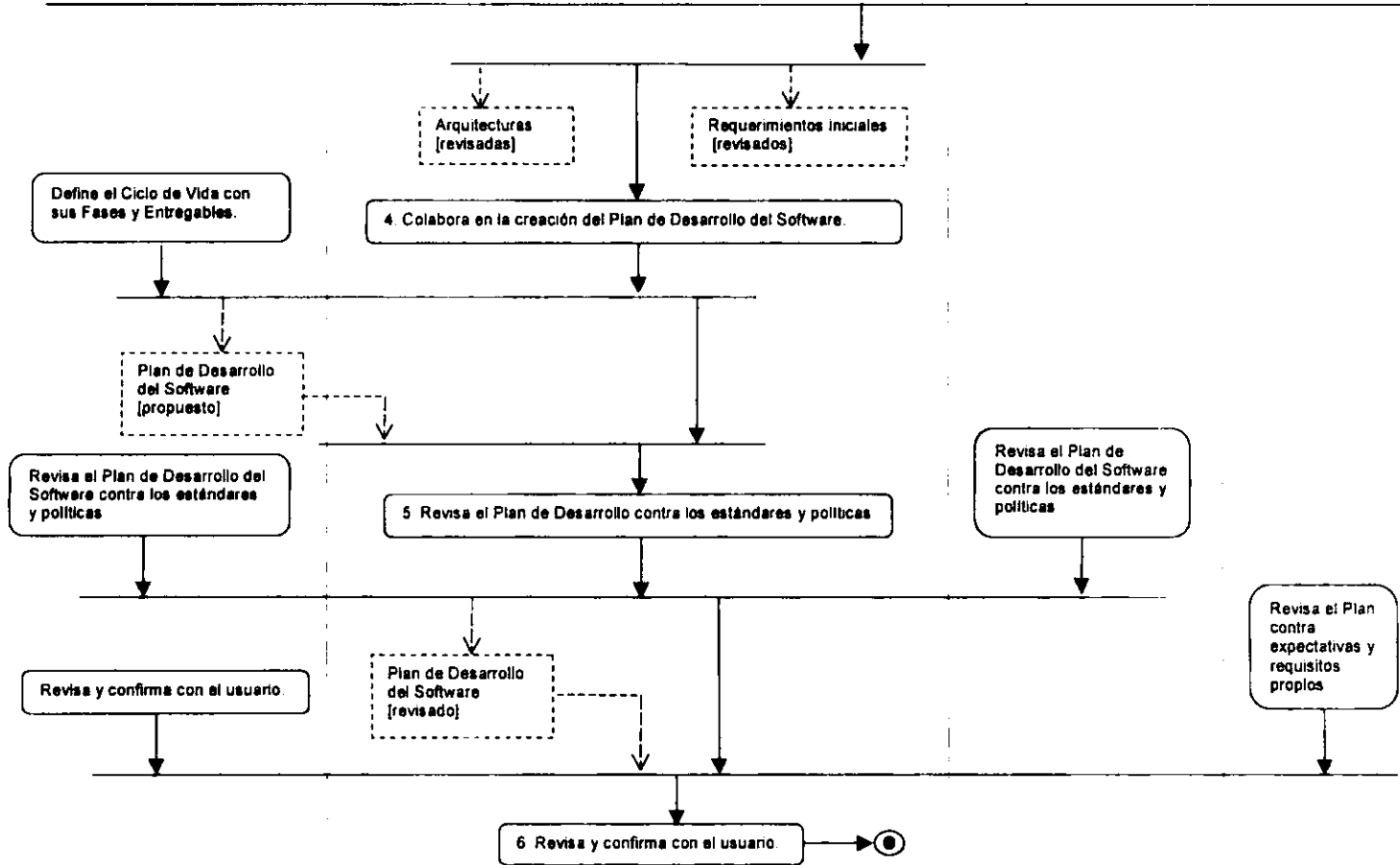


Figura 4.5

Actividades del Líder del Proyecto en la Fase de Inicio

Actividad	Procesos del PMBOK	Métricas ¹⁸	Entradas	Métodos, técnicas e Instrumentos
	Actividades de CMM-2 ¹⁹	Validaciones ²⁰	Salidas	
1 Revisa las políticas para la Administración de Requerimientos y Contratistas.	<ul style="list-style-type: none"> 2.1 Iniciación. 3.1 Definición de Actividades. 		<ul style="list-style-type: none"> Política Documentada y aceptada para Administrar Requerimientos. Política Documentada y aceptada para Administrar Contratistas. 	<ul style="list-style-type: none"> Reuniones.
	<p><i>Requerimientos:</i></p> <ul style="list-style-type: none"> C1. Sigue una política escrita de la organización para administrar los Requerimientos del Sistema. <p><i>Subcontratistas:</i></p> <ul style="list-style-type: none"> C1. Sigue una política escrita de la organización para la administración de Subcontratistas. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. 		
2 Analiza los requerimientos iniciales en forma general	<ul style="list-style-type: none"> 3.1 Definición de Actividades. 3.3 Estimación de Actividades. 	<ul style="list-style-type: none"> Trabajo y progreso. Requerimientos Arquitectura inicial Involucrados 	<ul style="list-style-type: none"> Minutas de reuniones y entrevistas. Compromisos aceptados. Descripción oficial del producto. Descripción de restricciones y supuestos. Directrices e ideas del usuario principal. Modelo y definición inicial del negocio. Ejemplos útiles del negocio Documentos de los entrevistados 	<ul style="list-style-type: none"> Reuniones y entrevistas Toma de declaraciones. CASE y Metodologías Ingeniería de Requerimientos. Identificación de alternativas. Ciclo de Vida del SW. Descomposición funcional. Modelado de la información.
	<p><i>Requerimientos:</i></p> <ul style="list-style-type: none"> A1. El GIS revisa los requerimientos antes de ser incorporados al Proyecto de SW. <p><i>Planeación:</i></p> <ul style="list-style-type: none"> A5. Identifica Ciclo de Vida del SW. A8. Identifica productos de SW. 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. Planeación y Estimación. 	<ul style="list-style-type: none"> Arquitecturas propuestas con prioridades, dependencias y supuestos. Requerimientos iniciales propuestos. Descripción funcional y de información Glosario inicial de términos Límites del proyecto con otros sistemas 	

¹⁸ Ver figura 2.20 y figura 2.24

¹⁹ C (compromisos), A (actividades), V (Verificaciones)

²⁰ Ver figura 2.23.

3	<p>Revisa los Requerimientos iniciales contra los estándares y políticas</p> <ul style="list-style-type: none"> • 3.1 Definición de Actividades • 5.2 Aseguramiento de la Calidad <p><i>Requerimientos</i></p> <ul style="list-style-type: none"> • V3 Calidad revisa, audita y reporta las actividades de Requerimientos 	<ul style="list-style-type: none"> • Trabajo y progreso • Seguimiento <p>• Estrategia con el Líder del Proyecto</p>	<ul style="list-style-type: none"> • Requerimientos iniciales. • Arquitecturas propuestas • Estándares y Políticas aceptadas <p>• Requerimientos iniciales revisados</p> <ul style="list-style-type: none"> • Arquitecturas revisadas 	<ul style="list-style-type: none"> • Técnicas de evaluación y revisión • Procesos y cuestionarios para el aseguramiento de la calidad • Reuniones
4	<p>Colabora en la creación del Plan de Desarrollo del Software</p> <ul style="list-style-type: none"> • 1.1 Crea el Plan del Proyecto • 2.2 Planeación del Alcance • 2.3 Definición del Alcance • 3.1 Definición de Actividades • 3.3 Estimación de Actividades • 4.1 Planeación de Recursos <p><i>Seguimiento</i></p> <ul style="list-style-type: none"> • A1 Documenta Plan de Desarrollo de SW • A2 Asigna responsabilidades sobre actividades y productos <p><i>Subcontratistas</i></p> <ul style="list-style-type: none"> • A1 Define y planea el trabajo de los subcontratistas 	<ul style="list-style-type: none"> • Trabajo y progreso • Costo presupuestado y gastos • Estimados • Involucrados • Requerimientos <p>• Estrategia, con el Administrador del Proyecto</p> <ul style="list-style-type: none"> • Planeación y estimación • Modelo del Ciclo de Vida 	<ul style="list-style-type: none"> • Minutas de reuniones • Arquitecturas revisadas • Requerimientos iniciales revisados • Presupuestos, tiempos y organigramas • Descripción de recursos • Compromisos aceptados <p>• Plan de Desarrollo del Software revisado</p>	<ul style="list-style-type: none"> • Reuniones • Experiencia en toma de decisiones • Análisis de costo / beneficio • Prácticas de recursos humanos y reclutamiento • Análisis conocimiento y habilidades de los involucrados • Estimaciones análogas
5	<p>Revisa el Plan de Desarrollo del Software contra los estándares y políticas</p> <ul style="list-style-type: none"> • 1.1 Creación del Plan de Desarrollo del Proyecto <p><i>Requerimientos</i></p> <ul style="list-style-type: none"> • V3 Calidad revisa, audita y reporta las actividades y productos de la administración de Requerimientos <p><i>Subcontratistas</i></p> <ul style="list-style-type: none"> • V3 Calidad revisa, audita y reporta las actividades y productos de la Administración de Subcontratistas 	<ul style="list-style-type: none"> • MTBF y maduración • Seguimiento <p>• Estrategia</p>	<ul style="list-style-type: none"> • Plan de Desarrollo del Software Propuesto <p>• Plan de Desarrollo del Software Revisado</p>	<ul style="list-style-type: none"> • Reuniones • Técnicas de evaluación y revisión • Procesos y cuestionarios para el aseguramiento de la calidad
6	<ul style="list-style-type: none"> • 2.4 Verificación del alcance 	<ul style="list-style-type: none"> • Involucrados • Seguimiento <p>• Estrategia</p> <ul style="list-style-type: none"> • Planeación y Estimación • Comunicación e Información 	<ul style="list-style-type: none"> • Plan de Desarrollo del Software Revisado <p>• PDS Aceptado por el usuario</p> <ul style="list-style-type: none"> • Cambios a los modelos del Plan de Desarrollo 	<ul style="list-style-type: none"> • Entrevistas en grupos • Técnicas de presentación • Ayudas visuales estructuradas • Reportes

Tabla 4.5

CAPÍTULO 5. Fases de Elaboración y Construcción.

Este es el segundo capítulo que conforma la guía para la administración de proyectos de software. Abarca dos fases del Inicio del Ciclo de Vida que se presentaron en la sección 3.1.1: la de Elaboración y la de Construcción.

Como ya se indicó en la unidad 3, la guía se presenta usando marcos (ver la sección 3.2.1), patrones (ver la sección 3.2.2) y una tabla resumen (ver la sección 3.2.3). Cada patrón hace referencia a un rol (ver la sección 3.1.2), los cuales están relacionados directamente con las actividades del Administrador Principal y, en forma gráfica, estos patrones se encuentran detallados mediante un diagrama de actividades (ver el Anexo A).

Para cada fase, se detalla su marco en forma independiente, pero los patrones de ambas fases, se fusionan en uno solo, dada las características similares en las actividades que cada rol realiza desde el punto de vista administrativo.

5.1. Marco para la Fase de Elaboración.

Procesos	De Planeación, de Control y de Ejecución
Objetivos	<ul style="list-style-type: none"> • Afina el Plan de Desarrollo del Proyecto • Detalla y especifica cada uno de los requerimientos iniciales • Analiza y diseña iterativamente los requerimientos iniciales • Propone y estructura las alternativas válidas de solución • Selecciona la mejor arquitectura de solución • Validar y autorizar con el usuario el camino a seguir • Crea la arquitectura base confiable para la fase de construcción • Demuestra que la arquitectura base resolverá el problema dentro del tiempo y costo presupuestado
Compuesta por	<ul style="list-style-type: none"> • El 70 % de los Requerimientos • El 100 % del Análisis • El 50 % del Diseño • El 25 % de las Pruebas • El 25 % de la Documentación
Tiempo utilizado	<ul style="list-style-type: none"> • El 30 % del total
Factores críticos de Éxito	<ul style="list-style-type: none"> • Participación activa y comprometida de los usuarios • Verificación de la existencia y de la calidad de los entregables de la fase anterior • Identificación de todos los elementos importantes y supuestos clave • Control total del trabajo en detalle para seguir los calendarios planeados • Reuniones exitosas de retroalimentación • Adaptabilidad a cambios imprevistos de circunstancias y objetivos

Roles:

- Administrador Principal.
- Administrador del Proyecto de Software (Administrador del Proyecto).
- Administrador de Configuraciones.
- Administrador de Calidad.
- Líderes de Proyecto.
- Grupos de Ingeniería de Software.

Entregables de la Fase:

Entradas:

- Plan de Desarrollo del Proyecto:
 - Descripción oficial del producto.
 - Descripción de restricciones.
 - Descripción de supuestos.
 - Lista de actividades.
 - Calendario de trabajo.
 - Asignación de roles y responsabilidades
 - Equipo de trabajo.
 - Plan de Abastecimiento.
 - Requerimiento de recursos.
 - Estimación de costos.
- Origen y síntomas de riesgos.
 - Eventos potenciales de riesgo.
 - Riesgos a resolver y a ignorar
- Plan de Calidad.
- Plan de Configuración.
- Cotizaciones.
- Plan de Contingencia

Salidas:

- Modelo de información aceptado
- Arquitectura del sistema.
- Tamaños y volúmenes de información
- Estándares de programación
- Plan de Construcción aceptado
- Modelo de funcionalidad aceptado
- Esquemas lógicos y físicos.
- Especificación de componentes
- Manuales de usuario preliminares
- Plan de Transición aceptado

Subprocesos:

- De Planeación:
 - 8.1. Identificar riesgos
 - 8.2. Cuantificar riesgos
- De Control:
 - 7.3. Reporte del seguimiento.
 - 1.3. Control integral de cambios
 - 2.5. Control de cambios al alcance
 - 3.5. Control del calendario
 - 4.4. Control de costos
 - 5.3. Control de calidad
 - 8.4. Plan de contingencia
- De Ejecución:
 - 1.2. Ejecución del plan del proyecto
 - 7.2. Distribución de la información
 - 9.3. Cotización
 - 6.3. Desarrollo del equipo de trabajo
 - 5.2. Aseguramiento de la calidad
 - 9.5. Administración de contratos

Métricas:

- Trabajo y Progreso.
- Estimación y planeación.
- Cambios y estabilidad.
- Estropicios y modularidad.
- Costo presupuesta y gastos.
- Trabajo duplicado y adaptabilidad.
- Equipo de trabajo y su dinámica.

Criterios de Evaluación:

- Involucrados
- Estabilidad de visibilidad
- Riesgos
- Defectos
- Requerimientos
- Estabilidad de solución
- Seguimiento

Verificaciones	<ul style="list-style-type: none"> • Estrategia con individuos externos. • Estrategia con Admón. Principal • Estrategia con Admón. de Calidad • Comunicación e Información • Compromisos y motivación y capacitación • Requerimientos 	<ul style="list-style-type: none"> • Estrategia con Admón. del Proyecto • Estrategia con Administración de la Configuración. • Estrategia con Líderes de Proyecto • Planeación y estimación. • Modelo del Ciclo de Vida. • Análisis y Diseño
Procedimientos Métodos, Técnicas e Instrumentos	<ul style="list-style-type: none"> • Reuniones. • Reportes de Progreso • Experiencia en toma de decisiones. • Medidas de rendimiento. • Análisis del producto • Análisis de costo / beneficio • Técnicas de costeo • Identificación de alternativas • Análisis de riesgos • Listas de verificación • Valor monetario esperado. • Análisis, conocimiento y habilidades de los involucrados • Creación de reportes 	<ul style="list-style-type: none"> • Criterios de selección de proyectos. • Planeación de proyectos • Compresión de tiempos. • Seguros • Ciclos de Vida del Software • Descomposición. • Estimaciones análogas. • Técnicas para cotizar • Tecnología de comunicaciones • Prácticas de recursos humanos y reclutamiento • Uso de ayudas visuales para presentaciones • Software de administración de proyectos
Patrones aplicables:	<ul style="list-style-type: none"> • La Fase Inicial (<i>The Initiate Phase</i>) [AMB98] • Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98] • Justificación del Proyecto (<i>Justify the Project</i>) [AMB98] • Definición de Documentos Administrativos Iniciales (<i>Define the Initial Management Documents</i>) [AMB98]. • Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98]. • El Modelo (<i>Model</i>) [AMB98] • El Administrador Principal en la fase de Elaboración y Construcción • El Administrador del Proyecto en la fase de Elaboración y Construcción • El Administrador de Configuración en la fase de Elaboración y Construcción • El Administrador de la Calidad en la fase de Elaboración y Construcción • Los Líderes de Proyecto en la fase de Elaboración y Construcción 	
Antipatrones	<ul style="list-style-type: none"> • Festival de Opiniones (<i>Blowhard Jamboree</i>) [BRO98]. • Parálisis del Análisis (<i>Analysis Paralysis</i>) [BRO98]. • Mazorca (<i>Corncock</i>) [BRO98] • Violencia Intelectual (<i>Intellectual Violence</i>) [BRO98]. • Administración Irracional (<i>Irrational Management</i>) [BRO98]. • Administración incorrecta del Proyecto (<i>Project Mismanagement</i>) [BRO98] • Arrojarlo en la pared (<i>Throw it over the wall</i>) [BRO98] • El odio (<i>The Feud</i>) [BRO98] 	
Área clave del CMM:	<ul style="list-style-type: none"> • Planeación del Proyecto de Software. • Administración de Requerimientos. • Aseguramiento de la Calidad del Software • Administración de la Configuración del Software. • Administración de Subcontratistas. • Supervisión y seguimiento del Proyecto de Software 	

5.2. Marco para la Fase de Construcción.

Procesos:	De Control y de Ejecución.	
Objetivos:	<ul style="list-style-type: none"> • Construye el sistema en términos de módulos o componentes. • Detalla los elementos propios de la construcción que resuelva cada uno de los requerimientos iniciales. • Madura y detalla el plan de la integración del sistema. • Madura y ejecuta el Plan de pruebas de unidad y pruebas de integración. • Implantar la plataforma de hardware que soluciona el problema. • Demuestra que la arquitectura diseñada resuelve el problema dentro del tiempo y costo presupuestado. 	
Compuesta por:	<ul style="list-style-type: none"> • El 50 % del • El 70 % de la • El 25 % de las • El 25 % de la 	Diseño Implementación Pruebas. Documentación.
Tiempo utilizado:	<ul style="list-style-type: none"> • El 40 % del total. 	
Factores críticos de Éxito	<ul style="list-style-type: none"> • Previsión y solución de problemas que generen bajo desempeño (infraestructura de redes, control de procesos, seguridad y espacios, etc.). • Puesta a punto de programas y bases de datos. • Planes de pruebas que incluyan pruebas de límite, excepción, de rendimiento, de máximo esfuerzo, etc. • Verificación de la existencia y de la calidad de los entregables de la fase anterior. • Control total del trabajo en detalle para seguir los calendarios planeados. • Reuniones exitosas de retroalimentación. • Adaptabilidad a cambios imprevistos de circunstancias y objetivos. 	
Roles:	<ul style="list-style-type: none"> • Administrador Principal. • Administrador del Proyecto de Software (Administrador del Proyecto). • Administrador de Configuraciones. • Administrador de Calidad. • Líderes de Proyecto. • Grupos de Ingeniería de Software. • Ingenieros especializados (Administrador de Bases de Datos, de Redes, experto en tecnología Internet, etc.). 	
Entregables de la Fase:		
Entradas:	<ul style="list-style-type: none"> • Modelo de información aceptado • Arquitectura del sistema. • Tamaños y volúmenes de información • Estándares de programación • Plan de Construcción aceptado 	<ul style="list-style-type: none"> • Modelo de funcionalidad aceptado • Esquemas lógicos y físicos. • Especificación de componentes • Manuales de usuario preliminares • Plan de Transición aceptado
Salidas:	<ul style="list-style-type: none"> • Producto de Software construido • Manuales de Usuario y de Operación 	<ul style="list-style-type: none"> • Resultado de todas las pruebas. • Plan de Transición aceptado.

Subprocesos	<ul style="list-style-type: none"> • De Control: <ul style="list-style-type: none"> ▪ 7.3. Reporte del seguimiento. ▪ 1.3. Control integral de cambios ▪ 2.5. Control de cambios al alcance ▪ 3.5. Control del calendario • De Ejecución: <ul style="list-style-type: none"> ▪ 1.2. Ejecución del plan del proyecto ▪ 7.2. Distribución de la información ▪ 9.5. Administración de contratos 	<ul style="list-style-type: none"> ▪ 4.4. Control de costos ▪ 5.3. Control de calidad ▪ 8.4. Plan de contingencia ▪ 6.3. Desarrollo del equipo de trabajo ▪ 5.2. Aseguramiento de la calidad
Métricas	<ul style="list-style-type: none"> • Trabajo y Progreso • Estimación y planeación. • Cambios y estabilidad. • Estropicios y modularidad. 	<ul style="list-style-type: none"> • Costo presupuesta y gastos. • Trabajo duplicado y adaptabilidad • Equipo de trabajo y su dinámica • MTBF y maduración
Criterios de Evaluación:		
<ul style="list-style-type: none"> • Involucrados • Estabilidad de visibilidad • Riesgos • Defectos • Estabilidad de construcción 	<ul style="list-style-type: none"> • Requerimientos • Estabilidad de solución • Congruencia • Seguimiento 	
Verificaciones	<ul style="list-style-type: none"> • Estrategia con individuos externos. • Estrategia con Admón. Principal. • Estrategia con Admón. de Calidad. • Comunicación e información. • Compromisos y motivación y capacitación • Requerimientos 	<ul style="list-style-type: none"> • Estrategia con Admón. del Proyecto • Estrategia con Administración de la Configuración • Estrategia con Líderes de Proyecto • Planeación y estimación. • Modelo del Ciclo de Vida. • Análisis y Diseño
Procedimientos Métodos, Técnicas e Instrumentos	<ul style="list-style-type: none"> • Reuniones • Reportes de Progreso • Experiencia en toma de decisiones • Medidas de rendimiento • Análisis del producto • Análisis de costo / beneficio • Técnicas de costeo • Identificación de alternativas • Análisis de riesgos • Listas de verificación • Valor monetario esperado. • Análisis, conocimiento y habilidades de los involucrados • Creación de reportes 	<ul style="list-style-type: none"> • Criterios de selección de proyectos. • Planeación de proyectos • Compresión de tiempos • Seguros • Ciclos de Vida del Software • Descomposición • Estimaciones análogas • Técnicas para cotizar. • Tecnología de comunicaciones • Prácticas de recursos humanos y reclutamiento. • Uso de ayudas visuales para presentaciones. • Software de administración de proyectos.
Patrones aplicables	<ul style="list-style-type: none"> • La Fase Inicial (<i>The Initiate Phase</i>) [AMB98]. • Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98]. • Justificación del Proyecto (<i>Justify the Project</i>) [AMB98]. • Definición de Documentos Administrativos Iniciales (<i>Define the Initial Management Documents</i>) [AMB98]. • Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98]. 	

- El Modelo (*Model*) [AMB98].
- La fase de Construcción (*The Construct phase*) [AMB98].
- Programa (*Program*) [AMB98].
- Generalización (*Generalize*) [AMB98].
- El Administrador Principal en la fase de Elaboración y Construcción
- El Administrador del Proyecto en la fase de Elaboración y Construcción
- El Administrador de Configuración en la fase de Elaboración y Construcción
- El Administrador de la Calidad en la fase de Elaboración y Construcción
- Los Líderes de Proyecto en la fase de Elaboración y Construcción

Antipatrones

- Festival de Opiniones (*Blowhard Jamboree*) [BRO98].
- Parálisis del Análisis (*Analysis Paralysis*) [BRO98].
- Mazorca (*Corncock*) [BRO98].
- Violencia Intelectual (*Intellectual Violence*) [BRO98].
- Administración Irracional (*Irrational Management*) [BRO98].
- Administración incorrecta del Proyecto (*Project Mismanagement*) [BRO98].
- Arrojarlo en la pared (*Throw it over the wall*) [BRO98].
- El odio (*The Feud*) [BRO98].

Área clave del CMM:

- Planeación del Proyecto de Software.
- Administración de Requerimientos.
- Aseguramiento de la Calidad del Software.
- Administración de la Configuración del Software.
- Administración de Subcontratistas.
- Supervisión y seguimiento del Proyecto de Software.

5.3. Patrones por rol para la Fase de Elaboración y Construcción.

EL ADMINISTRADOR PRINCIPAL EN LA FASE DE ELABORACIÓN Y CONSTRUCCIÓN

Nombre del Patrón	El Administrador Principal en la fase de Elaboración y Construcción
Problema:	Queremos determinar las actividades y responsabilidades del Administrador Principal en la fase de Elaboración y Construcción de un proyecto software.
Relacionado con:	<ul style="list-style-type: none"> • La Fase Inicial (<i>The Initiate Phase</i>) [AMB98] • Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98] • Justificación del Proyecto (<i>Justify the Project</i>) [AMB98] • Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98]. • El Modelo (<i>Model</i>) [AMB98]. • El Administrador del Proyecto en la fase de Elaboración y Construcción • El Administrador de la Calidad en la fase de Elaboración y Construcción
Solución:	La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades ¹ de la figura 5.1; y las actividades correspondientes están detalladas en la tabla 5.1.

EL ADMINISTRADOR DEL PROYECTO EN LA FASE DE ELABORACIÓN Y CONSTRUCCIÓN

Nombre del Patrón	El Administrador del Proyecto en la fase de Elaboración y Construcción
Problema:	Queremos determinar las actividades y responsabilidades del Administrador del Proyecto de Software (Administrador del Proyecto) en la fase de Elaboración y Construcción del desarrollo del Proceso de Software.
Relacionado con:	<ul style="list-style-type: none"> • La Fase Inicial (<i>The Initiate Phase</i>) [AMB98] • Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98]. • Justificación del Proyecto (<i>Justify the Project</i>) [AMB98]. • Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98]. • El Modelo (<i>Model</i>) [AMB98] • El Administrador Principal en la fase de Elaboración y Construcción • El Líder del Proyecto en la fase de Elaboración y Construcción • El Administrador de la Configuración del Software en la fase de Elaboración y Construcción • El Administrador de la Calidad en la fase de Elaboración y Construcción
Solución:	La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades ² de la figura 5.2; y las actividades correspondientes están detalladas en la tabla 5.2.

EL ADMINISTRADOR DE LA CONFIGURACIÓN DEL SOFTWARE EN LA FASE DE ELABORACIÓN Y CONSTRUCCIÓN

Nombre del Patrón	El Administrador de la Configuración del Software en la fase de Elaboración y Construcción.
-------------------	--

¹ Vea el anexo A.

² *Idem*.

- Problema: Queremos determinar las actividades y responsabilidades del Administrador de la Configuración en la fase de Elaboración y Construcción del desarrollo del Proceso de Software.
- Relacionado con:
- Definición de la infraestructura del proyecto (*Define the Project Infrastructure*) [AMB98].
 - El Administrador del Proyecto en la fase de Elaboración y Construcción
 - El Administrador de la Calidad en la fase de Elaboración y Construcción
 - Los Líderes de Proyecto en la fase de Elaboración y Construcción
- Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades³ de la figura 5.3; y las actividades correspondientes están detalladas en la tabla 5.3.

EL ADMINISTRADOR DE LA CALIDAD EN LA FASE DE ELABORACIÓN Y CONSTRUCCIÓN

- Nombre del Patrón: **El Administrador de la Calidad en la fase de Elaboración y Construcción.**
- Problema: Queremos determinar las actividades y responsabilidades del Administrador de la Calidad en la fase de Elaboración y Construcción del desarrollo del Proceso de Software.
- Relacionado con:
- Definición de Documentos Administrativos Iniciales (*Define the Initial Management Documents*) [AMB98].
 - El Administrador del Proyecto en la fase de Elaboración y Construcción
 - El Administrador de la Configuración en la fase de Elaboración y Construcción
 - Los Líderes de Proyecto en la fase de Elaboración y Construcción
- Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades⁴ de la figura 5.4; y las actividades correspondientes están detalladas en la tabla 5.4.

EL LÍDER DEL PROYECTO EN LA FASE DE ELABORACIÓN Y CONSTRUCCIÓN

- Nombre del Patrón: **El Líder del Proyecto en la fase de Elaboración y Construcción**
- Problema: Queremos determinar las actividades y responsabilidades del Líder del Proyecto en la fase de Elaboración y Construcción del desarrollo del Proceso de Software.
- Relacionado con:
- La Fase Inicial (*The Initiate Phase*) [AMB98].
 - Definición y Validación de Requerimientos Iniciales (*Define and Validate Initial Requirements*) [AMB98].
 - Justificación del Proyecto (*Justify the Project*) [AMB98]
 - Definición de la infraestructura del proyecto (*Define the Project Infrastructure*) [AMB98]
 - El Modelo (*Model*) [AMB98].
 - El Administrador Principal en la fase de Elaboración y Construcción
 - El Administrador del Proyecto en la fase de Elaboración y Construcción
 - El Administrador de la Configuración del Software en la fase de Elaboración y Construcción
 - El Administrador de la Calidad en la fase de Elaboración y Construcción
- Solución: La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades⁵ de la figura 5.5; y las actividades correspondientes están detalladas en la tabla 5.5.

³ *idem*

⁴ *idem*.

⁵ *idem*.

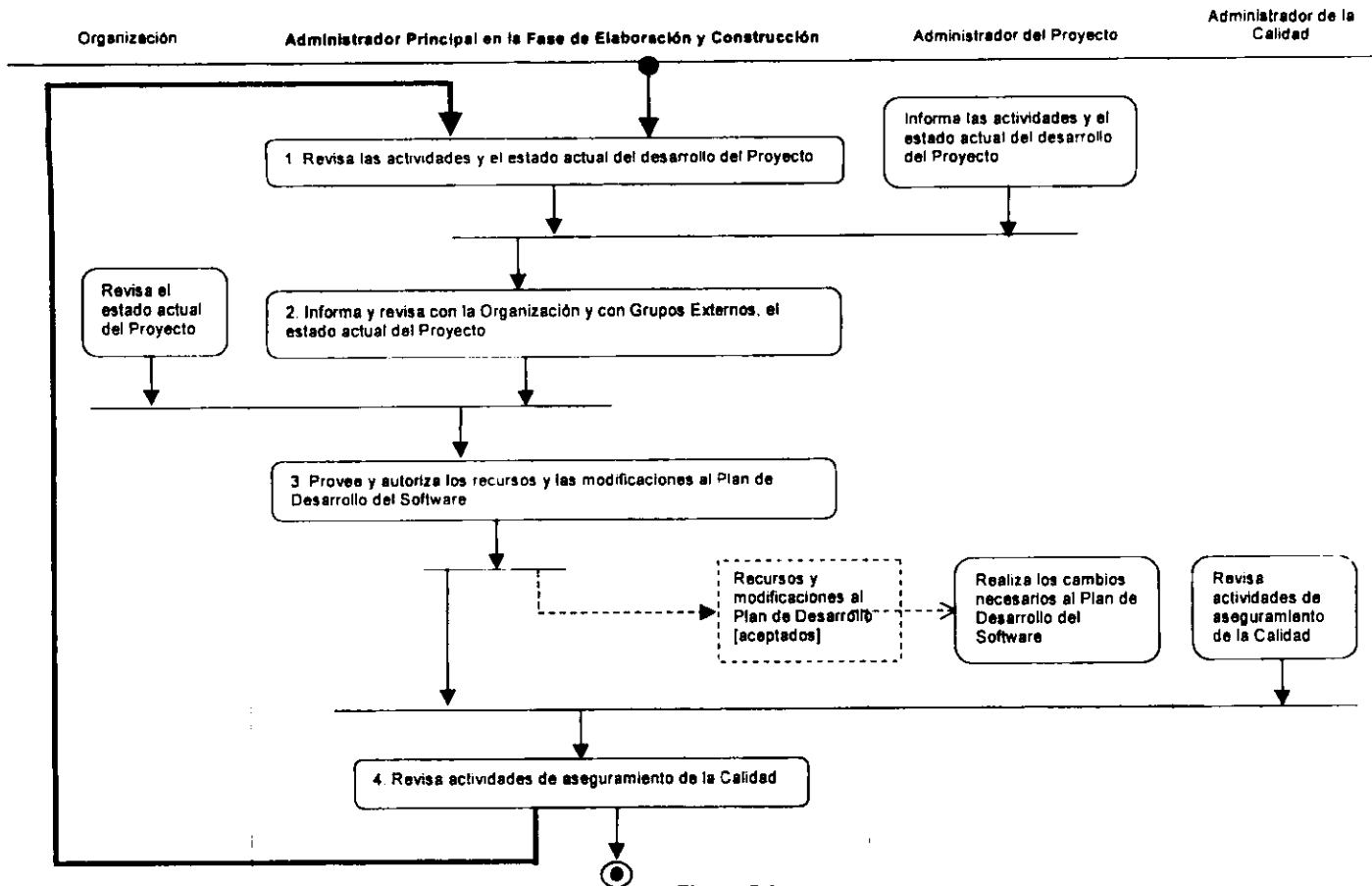


Figura 5.1

Actividades del Administrador Principal en la Fase de Elaboración y Construcción

Actividad	Procesos del PMBOK	Métricas ⁶	Entradas	Métodos, Métricas e Instrumentos
	Actividades de CMM-2 ⁷	Validaciones ⁸	Salidas	
1 Revisa las actividades y el estado actual del desarrollo del Proyecto.	<ul style="list-style-type: none"> 7.3 Reporte del seguimiento <p><i>Planeación:</i></p> <ul style="list-style-type: none"> V1. Revisa periódicamente actividades de planeación V2. Revisa periódicamente actividades de planeación con el Administrador del Proyecto. <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> A2. Revisa el Plan de Desarrollo del Software 	<ul style="list-style-type: none"> Trabajo y progreso Seguimiento <p>Estrategia con el Administrador del Proyecto</p>	<ul style="list-style-type: none"> Reporte de avances 	<ul style="list-style-type: none"> Experiencia en toma de decisiones. Reportes de progreso
2 Informa y revisa con la Organización y con Grupos Externos, el estado actual del Proyecto	<ul style="list-style-type: none"> 7.3 Reporte del seguimiento <p><i>Planeación:</i></p> <ul style="list-style-type: none"> A4. La Administración principal revisa los compromisos con individuos y grupos externos <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> A3. La Administración principal revisa los compromisos del Proyecto y los cambios acordados con individuos y grupos externos. 	<ul style="list-style-type: none"> Involucrados <p>Estrategia, con la organización y grupos externos</p>	<ul style="list-style-type: none"> Reporte de avances 	<ul style="list-style-type: none"> Reuniones.
3 Provee y autoriza los recursos y las modificaciones al Plan de Desarrollo del Software	<ul style="list-style-type: none"> 1.3 Control integral de cambios. <p><i>Requerimientos:</i></p> <ul style="list-style-type: none"> Revisa e incorpora al proyecto, todos los cambios a los requerimientos. 	<ul style="list-style-type: none"> Cambios y estabilidad Estabilidad de visión <p>Estrategia</p> <p>Planeación y Estimación</p>	<ul style="list-style-type: none"> Reportes de avance Recursos y modificaciones al Plan de Desarrollo 	<ul style="list-style-type: none"> Reuniones. Experiencia en toma de decisiones.
4 Revisa actividades de aseguramiento de la Calidad	<ul style="list-style-type: none"> 5.3 Control de la calidad <p><i>Calidad:</i></p> <ul style="list-style-type: none"> V1. Revisa periódicamente actividades de calidad. 	<ul style="list-style-type: none"> Seguimiento <p>Estrategia, con el Administrador de Calidad.</p>	<ul style="list-style-type: none"> Plan y avances de calidad 	<ul style="list-style-type: none"> Reuniones. Experiencia en toma de decisiones. Reportes de progreso

Tabla 5.1

⁶ Ver figura 2.20 y figura 2.24.

⁷ C (compromisos), A (actividades), V (Verificaciones)

⁸ Ver figura 2.23.

1. Se integra con otros grupos de trabajo: auditores, equipo técnico, equipo de operación, administradores de datos, proveedores, y demás

Estándares de entregables del Análisis y Diseño [aceptados]

Estándares de entregables de la Construcción [aceptados]

Reporte de Progreso del desarrollo del Proyecto [revisado]

Plan de Análisis y Diseño detallado [revisado]

Plan de la Construcción detallado [revisado]

Plan de Desarrollo del SW aceptado por el usuario [aceptado]

2. Revisa el Plan de desarrollo de seguimiento al progreso real del proyecto contra lo planeado

2.1 Sigue y toma acciones correctivas a los tamaños y productos de SW

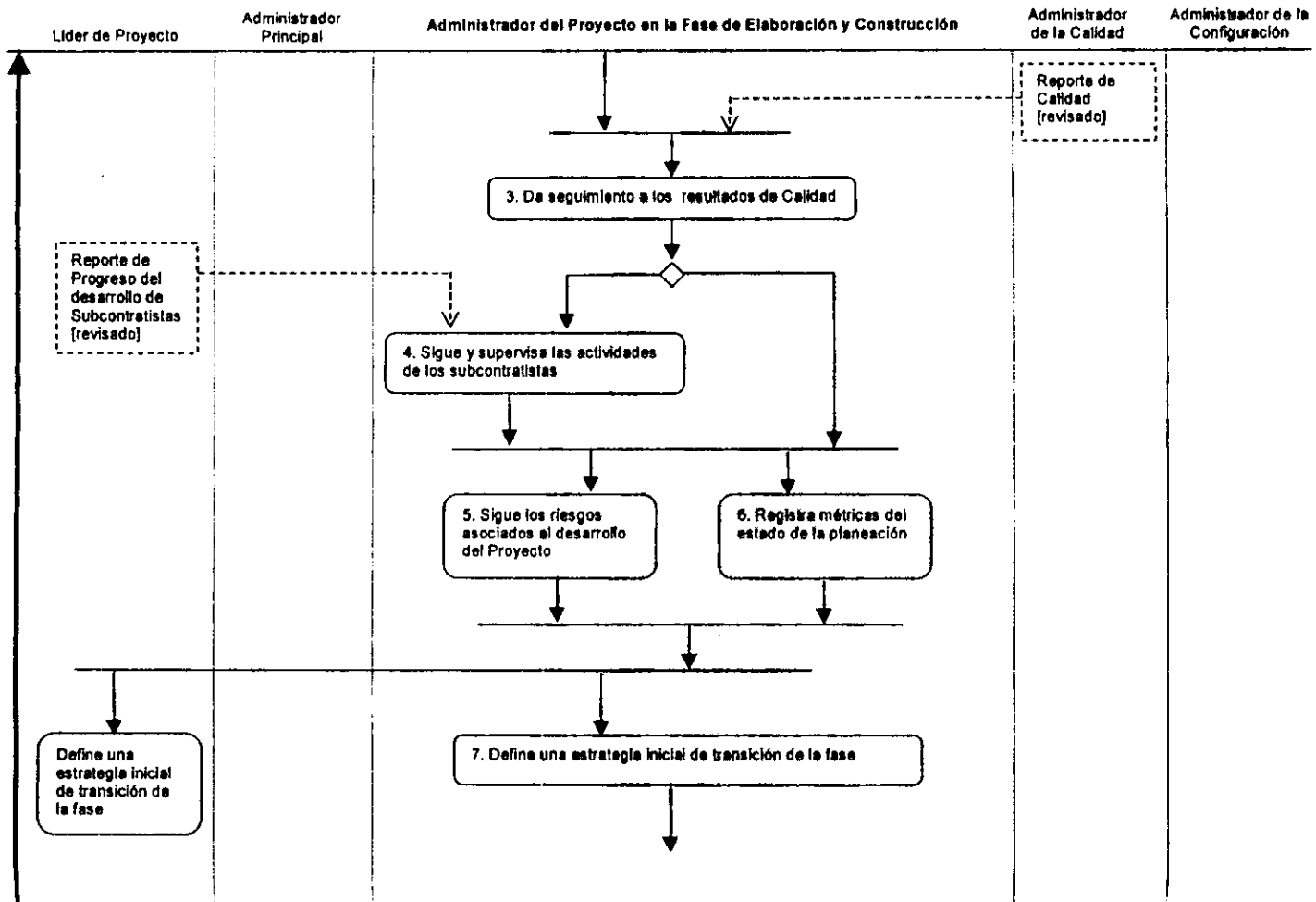
2.2 Sigue y toma acciones correctivas a los costos y esfuerzos

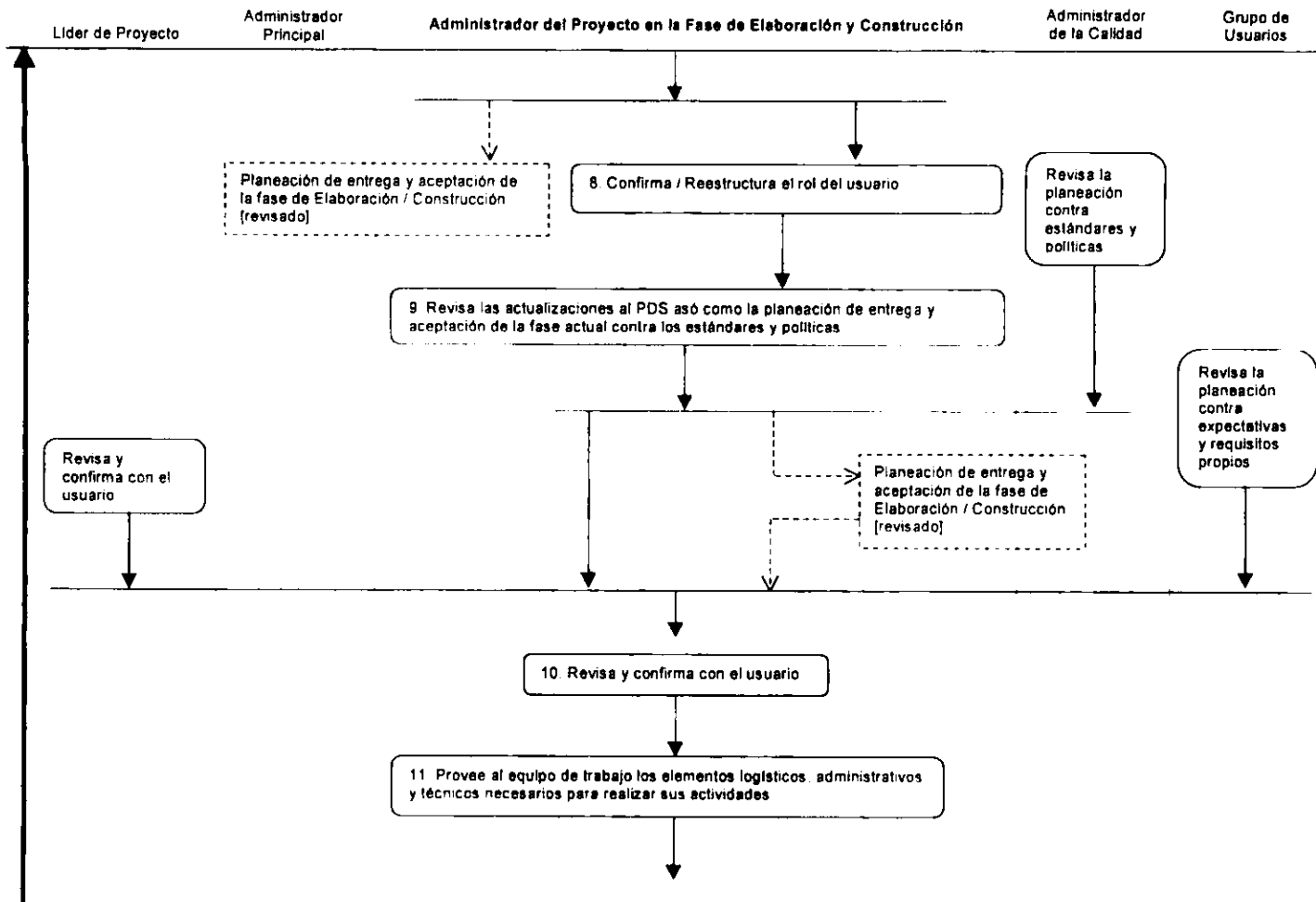
2.3 Sigue y toma acciones correctivas a los recursos de cómputo

2.4 Sigue y toma acciones correctivas al equipo de trabajo de ingeniería de SW

2.5 Sigue y toma acciones correctivas al calendario de actividades

2.6 Sigue y toma acciones correctivas de las actividades técnicas de Ingeniería de SW





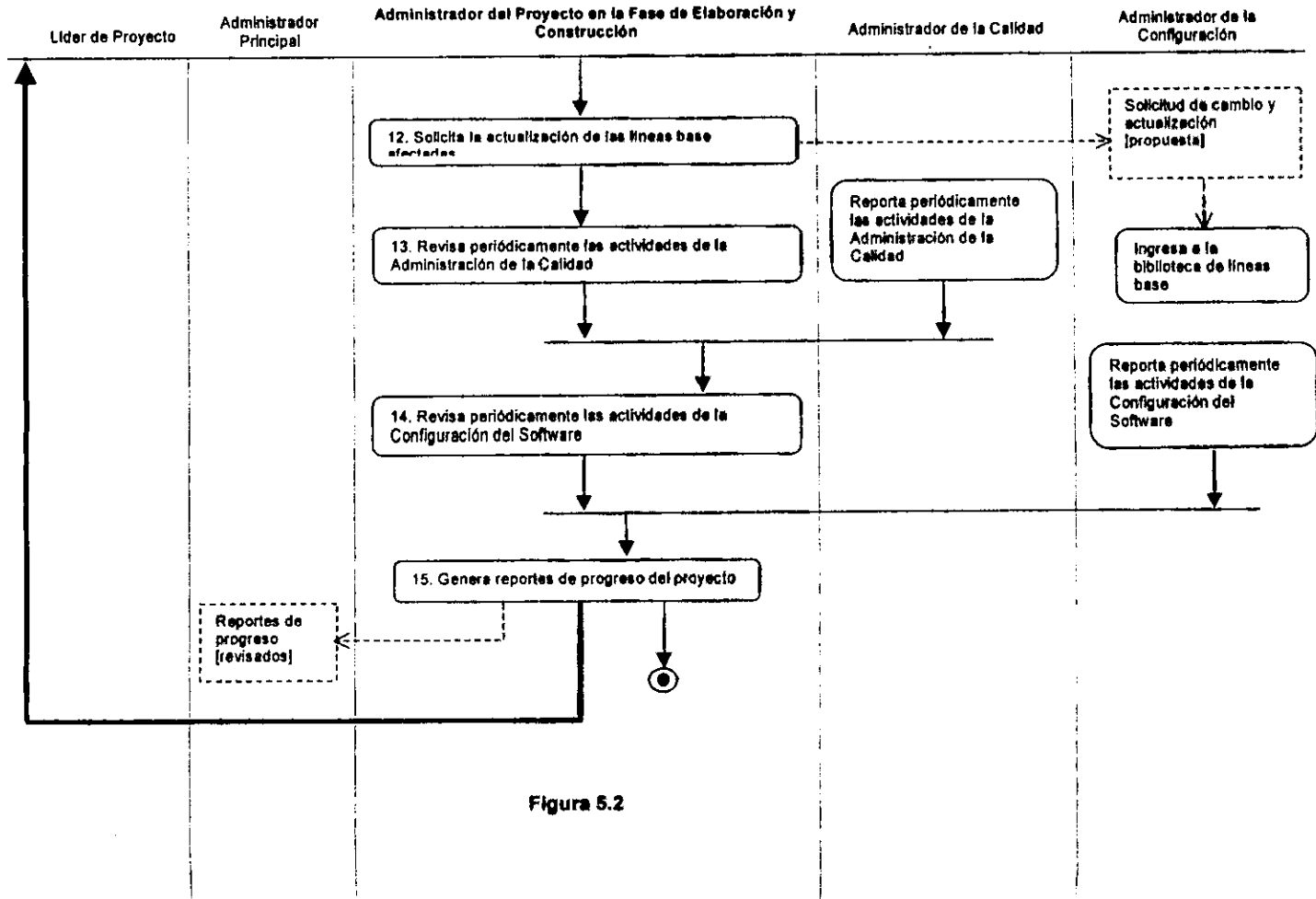


Figura 5.2

Actividades del Administrador del Proyecto en la Fase de Elaboración y Construcción

	Actividad	Procesos del PMBOK	Métricas ⁹	Entradas	Métodos, técnicas e Instrumentos
		Actividades de CMM-2 ¹⁰	Validaciones ¹¹	Salidas	
1	Se integra con otros grupos de trabajo auditores, equipo técnico, equipo de operación, administradores de datos, proveedores, y demás	<ul style="list-style-type: none"> • 6.2 Selección del personal • 6.3 Desarrollo del equipo de trabajo <p><i>Planeación</i></p> <ul style="list-style-type: none"> • A1 El GIS¹² forma parte del equipo de trabajo • A2 El GIS participa en la planeación • A3 El GIS participa con otros grupos en toda la vida del proyecto 	<ul style="list-style-type: none"> • Estrategia, con Administrador Principal, con Administrador de Calidad con Administrador de la Configuración y con los Lider(es) del Proyecto 		<ul style="list-style-type: none"> • Reuniones y entrevistas • Manejo de personal • Prácticas de recursos humanos • Análisis, conocimiento y habilidades de los involucrados
2	Revisa el Plan de desarrollo, da seguimiento al progreso real del proyecto contra lo planeado	<ul style="list-style-type: none"> • 1.2 Ejecución del Plan del Proyecto • 2.5 Control de cambios al alcance • 3.5 Control del calendario • 4.4 Control de costos • 8.3 Plan de contención • 8.4 Plan de contingencia <p><i>Seguimiento</i></p> <ul style="list-style-type: none"> • A1. Usar el Plan documentado para dar seguimiento del Proyecto • A2. Revisa el Plan del Proyecto • A5. Sigue los tamaños de productos de SW • A6. Sigue a los costos y esfuerzos • A7. Sigue los recursos de cómputo • A8. Sigue el calendario • A9. Sigue los aspectos técnicos • A13. Revisa los resultados contra lo planeado • V1. Revisa periódicamente las actividades con el Administrador del Proyecto 	<ul style="list-style-type: none"> • Trabajo y progreso • Costo presupuestado y gastos • Equipo de trabajo y su dinámica • Estimados • Seguimiento • Estabilidad de visibilidad • Estabilidad de solución <p>• Planeación y Estimación</p> <ul style="list-style-type: none"> • Compromisos, motivación y capacitación 	<ul style="list-style-type: none"> • Estándares aceptados de entregables del Análisis, Diseño y Construcción • Plan de Desarrollo del SW aceptado por el usuario • Reporte de Progreso del desarrollo del Proyecto • Planes revisados del Análisis Diseño y Construcción <p>• Solicitud de ajustes y cambios al Plan de Desarrollo del Software</p>	<ul style="list-style-type: none"> • Reportes de progreso • Control de proyectos • Experiencia en toma de decisiones • Medidas de rendimiento • Identificación de alternativas • Control de riesgos • Software de administración de proyectos

⁹ Ver figura 2.20 y figura 2.24

¹⁰ C (compromisos) A (actividades) V (Verificaciones)

¹¹ Ver figura 2.23

¹² Grupo de Ingeniería de Software

3	De seguimiento a los resultados de Calidad	<ul style="list-style-type: none"> • 5.2 Aseguramiento de la calidad • 5.3 Control de calidad <p><i>Planación:</i></p> <ul style="list-style-type: none"> • A4. Revisa los compromisos con individuos y grupos externos. 	<ul style="list-style-type: none"> • Estropicios y modularidad • Trabajo duplicado y adaptabilidad • MTBF y maduración • Defectos • Estabilidad de construcción 	<ul style="list-style-type: none"> • Reportes revisados de calidad 	<ul style="list-style-type: none"> • Experiencia en toma de decisiones. • Reuniones y entrevistas • Técnicas de evaluación y revisión • Listas de verificación
4	Segue y supervisa las actividades de los subcontratistas	<ul style="list-style-type: none"> • 9.5 Administración de contratos <p><i>Subcontratistas:</i></p> <ul style="list-style-type: none"> • A3. Usa el contrato con el contratista para su administración. • A4. Revisa el plan de desarrollo del subcontratista. • A5. Usa el plan del subcontratista para supervisar sus actividades. • A6. Sigue los cambios a los compromisos del contrato de los Subcontratistas • A7. El administrador del Proyecto revisa periódicamente la administración de subcontratistas • A13. Evalúa periódicamente el rendimiento del Subcontratista. 	<ul style="list-style-type: none"> • Trabajo y progreso • Costo presupuestado y gastos • Equipo de trabajo y su dinámica • Cambios y estabilidad • Involucrados • Estabilidad de construcción 	<ul style="list-style-type: none"> • Reporte revisado de Progreso del desarrollo de Subcontratistas 	<ul style="list-style-type: none"> • Reuniones. • Experiencia en toma de decisiones. • Medidas de rendimiento. • Análisis, conocimiento y habilidades de los involucrados

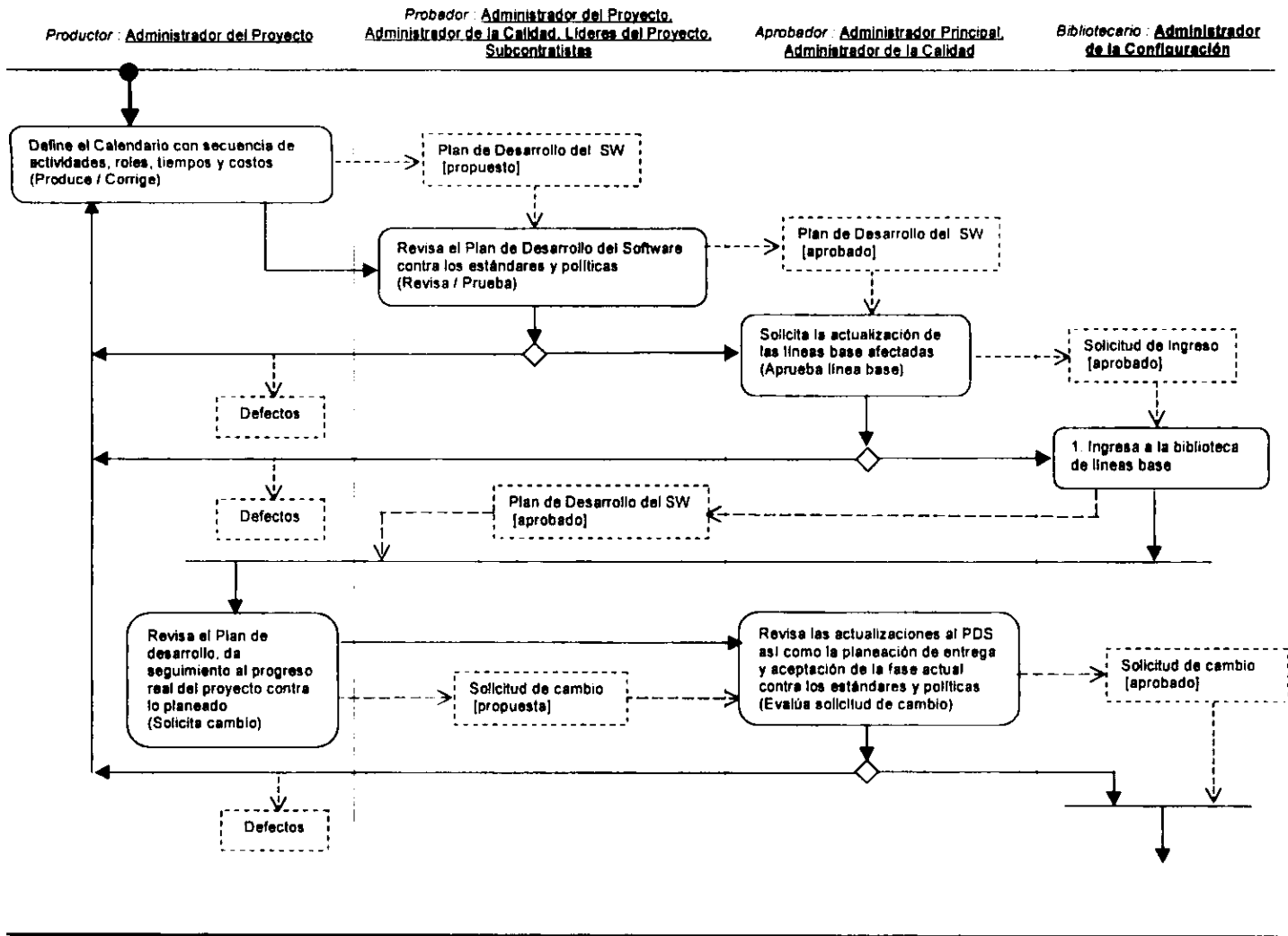
5	Sigue los riesgos asociados al desarrollo del Proyecto	<ul style="list-style-type: none"> • 8.1 Identificación de Riesgos. • 8.2 Cuantificación de Riesgos • 8.3 Genera Plan de Contención • 8.4 Plan de Contingencia 	<ul style="list-style-type: none"> • Cambios y estabilidad. • Estropicios y modularidad • Trabajo duplicado y adaptabilidad. • Congruencia • Defectos • Estabilidad de construcción 	<ul style="list-style-type: none"> • Reporte revisado de Progreso del desarrollo del proyecto • Plan de contingencia • Plan de contención 	<ul style="list-style-type: none"> • Reuniones. • Análisis de riesgos • Control de riesgos • Listas de verificación • Experiencia en toma de decisiones
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> • A13. Identificar medir y documentar riesgos del SW <p><i>Seguimiento</i></p> <ul style="list-style-type: none"> • A10. Sigue los riesgos asociados al proyecto 	<ul style="list-style-type: none"> • Comunicación e Información 	<ul style="list-style-type: none"> • Solicitud de ajustes y cambios al Plan de Desarrollo del Software 	
6	Registra métricas del estado de la planeación	<ul style="list-style-type: none"> • 2.4 Verificación del alcance 	<ul style="list-style-type: none"> • Trabajo y progreso • Estimados 	<ul style="list-style-type: none"> • Plan de Desarrollo del Software • Reporte revisado de Progreso del desarrollo del proyecto 	<ul style="list-style-type: none"> • Reuniones • Medidas de rendimiento
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> • A15. Registra datos de la planeación <p><i>Seguimiento</i></p> <ul style="list-style-type: none"> • A11. Registra métricas de datos y cambios a la planeación 	<ul style="list-style-type: none"> • Planeación y Estimación 		
7	Define una estrategia inicial de transición de la fase	<ul style="list-style-type: none"> • 2.4 Verificación del alcance 	<ul style="list-style-type: none"> • Trabajo y progreso • Estabilidad de construcción • Transición 		<ul style="list-style-type: none"> • Reuniones. • Control de proyectos • Experiencia en toma de decisiones.
			<ul style="list-style-type: none"> • Análisis y Diseño 	<ul style="list-style-type: none"> • Plan de Transición por revisar 	
8	Confirma / Reestructura el rol del usuario	<ul style="list-style-type: none"> • 6.3 Desarrollo del equipo de trabajo 	<ul style="list-style-type: none"> • Equipo de trabajo y su dinámica • Involucrados • Satisfacción 		<ul style="list-style-type: none"> • Reuniones y entrevistas • Análisis, conocimiento y habilidades de los involucrados • Manejo de personal
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> • A4. Revisa los compromisos con individuos y grupos externos 	<ul style="list-style-type: none"> • Compromisos, motivación y capacitación 	<ul style="list-style-type: none"> • Lista de actividades específicas para el usuario para las fases actuales y para la Transición 	

9	Revisa las actualizaciones al PDS así como la planeación de entrega y aceptación de la fase actual contra los estándares y políticas	<ul style="list-style-type: none"> 5.2 Aseguramiento de la Calidad 2.4 Verificación del alcance 	<ul style="list-style-type: none"> Trabajo y progreso Involucrados Transición Satisfacción 	<ul style="list-style-type: none"> Plan de Transición por revisar Estándares de calidad 	<ul style="list-style-type: none"> Reuniones. Revisión de estándares Listas de verificación
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> C2. Seguir una política escrita de la organización V3. Revisa periódicamente las actividades con el Administrador de la Calidad <p><i>Seguimiento</i></p> <ul style="list-style-type: none"> C2. Seguir una política escrita de la organización 	<ul style="list-style-type: none"> Comunicación e información. 	<ul style="list-style-type: none"> Planeación de entrega y aceptación de la fase de Elaboración / Construcción revisada 	
10	Revisa y confirma con el usuario	<ul style="list-style-type: none"> 7.2 Distribución de la Información 	<ul style="list-style-type: none"> Involucrados Satisfacción 	<ul style="list-style-type: none"> Planeación de entrega y aceptación de la fase de Elaboración / Construcción revisada Lista de actividades específicas para el usuario para las fases actuales y para la Transición 	<ul style="list-style-type: none"> Reuniones y entrevistas Análisis, conocimiento y habilidades de los involucrados. Uso de ayudas visuales para presentaciones Creación de reportes
		<p><i>Planeación:</i></p> <ul style="list-style-type: none"> A4. Revisa los compromisos con individuos y grupos externos 	<ul style="list-style-type: none"> Comunicación e información. 	<ul style="list-style-type: none"> Planeación de entrega y aceptación de la fase de Elaboración / Construcción aceptada por el usuario Lista de actividades específicas para el usuario para las fases actuales y para la Transición aceptadas por el usuario 	
11	Provee al equipo de trabajo los elementos logísticos, administrativos y técnicos necesarios para realizar sus actividades	<ul style="list-style-type: none"> 8.3 Plan de contención 8.4 Plan de contingencia 		<ul style="list-style-type: none"> Reporte revisado de Progreso del desarrollo del proyecto Plan de contingencia Plan de contención 	<ul style="list-style-type: none"> Reuniones. Experiencia en toma de decisiones Proposiciones análogas Conocimiento de elementos técnicos y logísticos
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> A14. Prepara instrumentos de apoyo y ayuda para el GIS 	<ul style="list-style-type: none"> Comunicación e información. 		

12	Solicita la actualización de las líneas base afectadas	<ul style="list-style-type: none"> • 1.3 Control integral de cambios • 2.5 Control de cambios al alcance • 3.5 Control del calendario • 4.4 Control de costos • 7.2 Distribución de la Información 	<ul style="list-style-type: none"> • Cambios y estabilidad. • Estropicios y modularidad. • Trabajo duplicado y adaptabilidad. • MTBF y maduración • Requerimientos • Análisis y Diseño • Unidad de Software • Estabilidad de construcción 		<ul style="list-style-type: none"> • Creación de reportes • Análisis de riesgos • Listas de verificación
		<p><i>Planeación:</i></p> <ul style="list-style-type: none"> • A9. Estima tamaños de productos • A10. Estima costos y esfuerzos • A11. Estima recursos <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> • A4. Comunicar al GIS la aprobación de los cambios <p><i>Configuración:</i></p> <ul style="list-style-type: none"> • A5. Supervisa los cambios y problemas reportados de elementos y unidades de la configuración • A6. Controla los cambios a las líneas base 		<ul style="list-style-type: none"> • Solicitud de ajustes y cambios al Plan de Desarrollo del Software • Solicitud de ajustes y cambios al Plan de Desarrollo de Subcontratistas 	
13	Revisa periódicamente las actividades de la Administración de la Calidad	<ul style="list-style-type: none"> • 5.2 Aseguramiento de la calidad • 5.3 Control de calidad 	<ul style="list-style-type: none"> • Trabajo y progreso 	<ul style="list-style-type: none"> • Plan de la Calidad. • Reportes de progreso de Calidad 	<ul style="list-style-type: none"> • Reuniones. • Revisión de estándares • Listas de verificación
		<p><i>Calidad:</i></p> <ul style="list-style-type: none"> • V1. Revisa periódicamente las actividades con el Administrador del Proyecto 	<ul style="list-style-type: none"> • Estrategia con Administrador de la Calidad • Compromisos, motivación y capacitación 	<ul style="list-style-type: none"> • Cambios, ajustes, sugerencias y aprobaciones 	
14	Revisa periódicamente las actividades de la Configuración del Software	<ul style="list-style-type: none"> • 5.2 Aseguramiento de la calidad • 5.3 Control de calidad 	<ul style="list-style-type: none"> • Trabajo y progreso 	<ul style="list-style-type: none"> • Plan de la Configuración • Reportes de progreso de la Configuración 	<ul style="list-style-type: none"> • Reuniones. • Revisión de estándares • Listas de verificación
		<p><i>Configuración:</i></p> <ul style="list-style-type: none"> • V1. Revisa periódicamente las actividades con el Administrador Principal • V2. Revisa periódicamente las actividades con el Administrador del Proyecto 	<ul style="list-style-type: none"> • Estrategia con Administrador de la Configuración • Compromisos, motivación y capacitación 	<ul style="list-style-type: none"> • Cambios, ajustes, sugerencias y aprobaciones 	

15	Genera reportes de progreso del proyecto	<ul style="list-style-type: none"> • 7.3 Reporte del seguimiento • 7.2 Distribución de la información • 5.2 Aseguramiento de la calidad 			<ul style="list-style-type: none"> • Uso de ayudas visuales para presentaciones • Creación de reportes • Software de administración de proyectos • Análisis, conocimiento y habilidades de los involucrados.
		<i>Planeación</i> <ul style="list-style-type: none"> • V1. Revisa periódicamente las actividades con el Administrador Principal • V2. Revisa periódicamente las actividades con el Administrador del Proyecto <i>Seguimiento:</i> <ul style="list-style-type: none"> • A3. Administración Principal revisa cambios y compromisos al Plan de Desarrollo 	<ul style="list-style-type: none"> • Comunicación e Información 	<ul style="list-style-type: none"> • Reportes de progreso 	

Tabla 5.2.



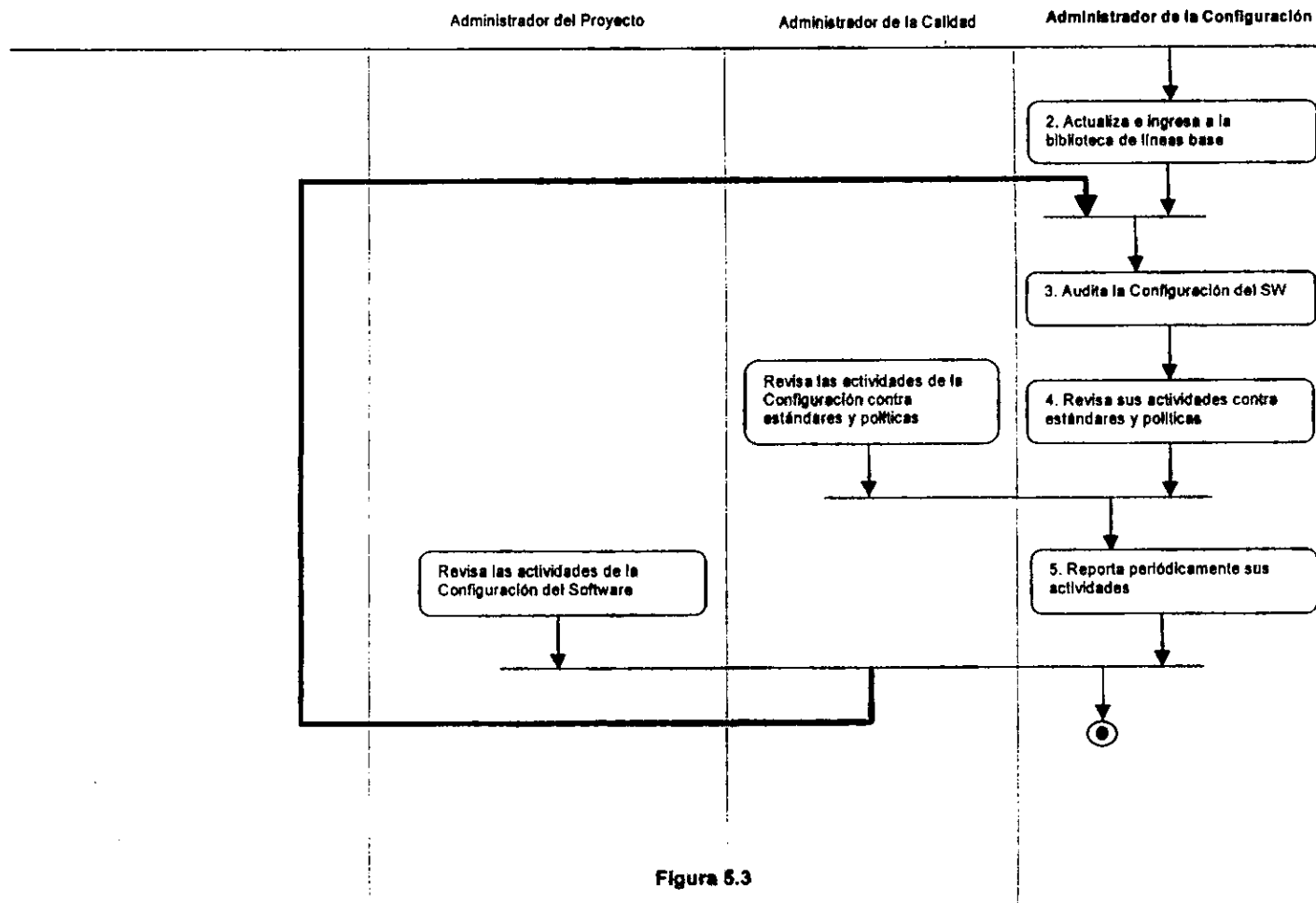


Figura 5.3

Actividades del Administrador de la Configuración en la Fase de Elaboración y Construcción

Actividad	Procesos del PMBOK	Métricas ¹³	Entradas	Métodos, técnicas e Instrumentos
	Actividades de CMM-2 ¹⁴	Validaciones ¹⁵	Salidas	
1 Ingresar a la biblioteca de líneas base	<ul style="list-style-type: none"> • 7.2 Distribución de la Información • 1.3 Control integral de cambios 		<ul style="list-style-type: none"> • Solicitud de ingreso 	<ul style="list-style-type: none"> • Formatos y estándares • Guías y Procedimientos
	<p><i>Configuración</i></p> <ul style="list-style-type: none"> • A2 Usa el Plan de la Configuración para realizar sus actividades • A5 Supervisa los cambios y problemas reportado de elementos y unidades de la configuración • A6 Controla los cambios a las líneas base • A7 Controla las versiones de los productos de las Bibliotecas de las líneas base 		<ul style="list-style-type: none"> • Biblioteca actualizada • Bitácora de trabajo 	
2 Actualiza e ingresa a la biblioteca de líneas base	<ul style="list-style-type: none"> • 7.2 Distribución de la información • 1.3 Control integral de cambios 		<ul style="list-style-type: none"> • Solicitud de cambio 	<ul style="list-style-type: none"> • Formatos y estándares • Guías y Procedimientos
	<p><i>Configuración</i></p> <ul style="list-style-type: none"> • A2 Usa el Plan de la Configuración para realizar sus actividades • A5 Supervisa los cambios y problemas reportado de elementos y unidades de la configuración • A6 Controla los cambios a las líneas base • A7 Controla las versiones de los productos de las Bibliotecas de las líneas base 		<ul style="list-style-type: none"> • Biblioteca actualizada • Bitácora de trabajo 	

¹³ Ver figura 2.20 y figura 2.24

¹⁴ C (compromisos), A (actividades), V (Verificaciones)

¹⁵ Ver figura 2.23

3	Audita la Configuración del SW	<ul style="list-style-type: none"> • 1.3 Control integral de cambios • 5.2 Aseguramiento de la calidad • 5.3 Control de la calidad 	<ul style="list-style-type: none"> • Trabajo y progreso • Cambios y estabilidad • Estropicios y modularidad • Trabajo duplicado y adaptabilidad • MTBF y maduración • Congruencia • Defectos 	<ul style="list-style-type: none"> • Plan de la Configuración del SW • Reportes de progreso. • Estado actual de las Bibliotecas • Estándares y políticas de la organización. 	<ul style="list-style-type: none"> • Experiencia en toma de decisiones. • Revisión de estándares • Listas de verificación • Procesos y cuestionario para el control de documentos
		<p><i>Configuración:</i></p> <ul style="list-style-type: none"> • A8. Registra el estado de los elementos y unidades de la configuración • A9. Genera reportes de sus actividades y del contenido de las líneas base • A10. Realiza auditorías a las líneas base • V3. Audita las líneas base para verificar su consistencia 	<ul style="list-style-type: none"> • Comunicación e información 	<ul style="list-style-type: none"> • Reporte del estado actual de las Bibliotecas • Sugerencias, cambios, adecuaciones a la Configuración 	
4	Revisa sus actividades contra estándares y políticas	<ul style="list-style-type: none"> • 1.3 Control integral de cambios • 5.2 Aseguramiento de la calidad • 5.3 Control de la calidad 	<ul style="list-style-type: none"> • Cambios y estabilidad. • Estropicios y modularidad • Trabajo duplicado y adaptabilidad • MTBF y maduración 	<ul style="list-style-type: none"> • Plan de la Configuración del SW • Reportes de progreso. • Estado actual de las Bibliotecas • Estándares y políticas de la organización. 	<ul style="list-style-type: none"> • Experiencia en toma de decisiones. • Revisión de estándares • Listas de verificación • Procesos y cuestionario para el control de documentos • Procesos y cuestionario para el aseguramiento de la calidad
		<p><i>Configuración:</i></p> <ul style="list-style-type: none"> • V4. Calidad revisa, audita y reporta las actividades y productos de trabajo de la Configuración 	<ul style="list-style-type: none"> • Estrategia, con el Administrador de la Calidad 	<ul style="list-style-type: none"> • Reporte de progreso auditado por Calidad 	
5	Reporta periódicamente sus actividades	<ul style="list-style-type: none"> • 7.2 Distribución de la información. • 7.3 Reporte del seguimiento 	<ul style="list-style-type: none"> • Trabajo y progreso 	<ul style="list-style-type: none"> • Plan de la Configuración del SW • Reportes de progreso. • Estado actual de las Bibliotecas • Estándares y políticas de la organización. 	<ul style="list-style-type: none"> • Experiencia en toma de decisiones. • Listas de verificación • Uso de ayudas visuales para presentaciones • Creación de reportes
		<p><i>Configuración:</i></p> <ul style="list-style-type: none"> • A9. Genera reportes de sus actividades y del contenido de las líneas base • V1. Revisa periódicamente sus actividades con el Administrador Principal • V2. Revisa periódicamente sus actividades con el Administrador del Proyecto 	<ul style="list-style-type: none"> • Estrategia, con el Administrador del Proyecto. 	<ul style="list-style-type: none"> • Reporte de progreso revisado por el Administrador del Proyecto 	

Tabla 5.3

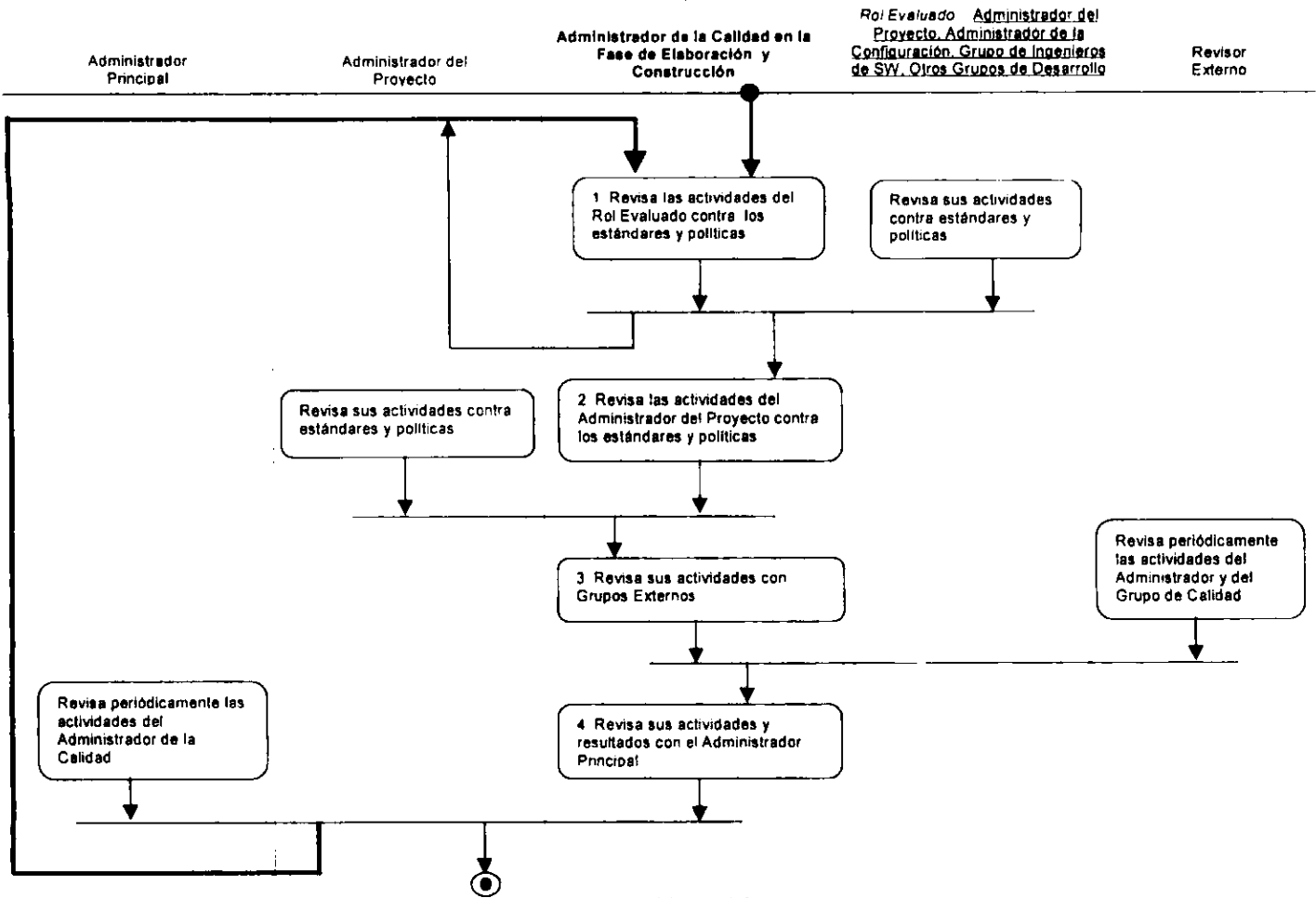


Figura 5.4

Actividades del Administrador de la Calidad en la Fase de Elaboración y Construcción

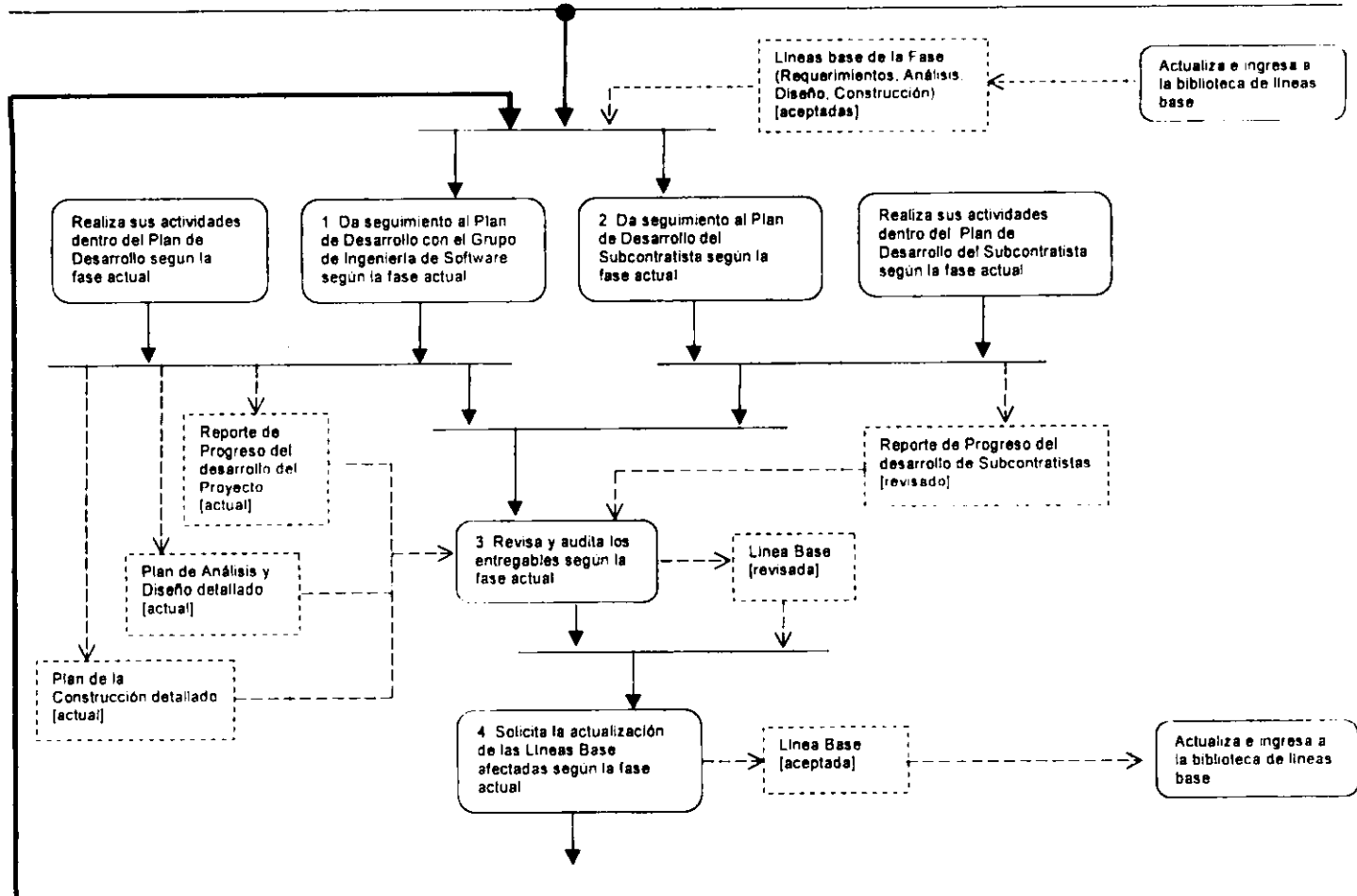
Actividad	Procesos del PMBOK	Métricas ¹⁶	Entradas	Métodos, técnicas e Instrumentos
	Actividades de CMM-2 ¹⁷	Validaciones ¹⁸	Salidas	
1 Revisa las actividades del Rol Evaluado contra los estándares y políticas	<ul style="list-style-type: none"> 5.2 Aseguramiento de la calidad 5.3 Control de la calidad 	<ul style="list-style-type: none"> Estropicios y modularidad Trabajo duplicado 	<ul style="list-style-type: none"> Descripción de los Procesos de cada Rol Reporte de progreso y actividades por Rol Estándares y Políticas aceptadas 	<ul style="list-style-type: none"> Reuniones y técnicas de evaluación Técnicas de aseguramiento de la Calidad.
	<p><i>Calidad:</i></p> <ul style="list-style-type: none"> A4. Revisa la realización de actividades de IS A5. Audita productos de trabajo de SW A6. Reporta sus actividades al GIS A7. Documenta y maneja las desviaciones en actividades y productos de trabajo 	<ul style="list-style-type: none"> Estrategia, con cada Rol 	<ul style="list-style-type: none"> Perfil, calificación, defectos y cambios 	
2 Revisa las actividades del Administrador del Proyecto contra los estándares y políticas	<ul style="list-style-type: none"> 5.2 Aseguramiento de la calidad 5.3 Control de la calidad 	<ul style="list-style-type: none"> Estropicios y modularidad Trabajo duplicado 	<ul style="list-style-type: none"> Plan de Calidad, Estándares y Políticas Resultados de la evaluación por Rol 	<ul style="list-style-type: none"> Reuniones y técnicas de evaluación Técnicas de aseguramiento de la Calidad.
	<p><i>Calidad:</i></p> <ul style="list-style-type: none"> V1. Revisa periódicamente sus actividades con el Administrador del Proyecto 	<ul style="list-style-type: none"> Estrategia, con el Administrador del Proyecto. 	<ul style="list-style-type: none"> Defectos, cambios y aprobación 	
3 Revisa sus actividades con Grupos Externos	<ul style="list-style-type: none"> 5.2 Aseguramiento de la calidad 5.3 Control de la calidad 	<ul style="list-style-type: none"> Estropicios y modularidad Trabajo duplicado 	<ul style="list-style-type: none"> Plan de Calidad, Estándares y Políticas Resultados de la evaluación por Rol 	<ul style="list-style-type: none"> Reuniones y técnicas de evaluación Técnicas de aseguramiento de la Calidad.
	<p><i>Calidad:</i></p> <ul style="list-style-type: none"> A8. Revisa sus auditorías periódicamente con el Grupo de Calidad del cliente 	<ul style="list-style-type: none"> Estrategia, con Grupos Externos y Cliente 	<ul style="list-style-type: none"> Perfil, calificación, defectos y cambios 	
4 Revisa sus actividades y resultados con el Administrador Principal	<ul style="list-style-type: none"> 5.2 Aseguramiento de la calidad 5.3 Control de la calidad 	<ul style="list-style-type: none"> Estropicios y modularidad Trabajo duplicado 	<ul style="list-style-type: none"> Plan de Calidad Resultados de la evaluación por Rol 	<ul style="list-style-type: none"> Reuniones y técnicas de evaluación Técnicas de aseguramiento de la Calidad.
	<p><i>Calidad:</i></p> <ul style="list-style-type: none"> A2. Sigue su propio Plan de Calidad V1. Revisa periódicamente sus actividades con el Administrador Principal 	<ul style="list-style-type: none"> Estrategia, con el Administrador Principal 	<ul style="list-style-type: none"> Defectos, cambios y aprobación 	

Tabla 5.4

¹⁶ Ver figura 2.20 y figura 2.24.

¹⁷ C (compromisos) A (actividades), V (Verificaciones)

¹⁸ Ver figura 2.23.



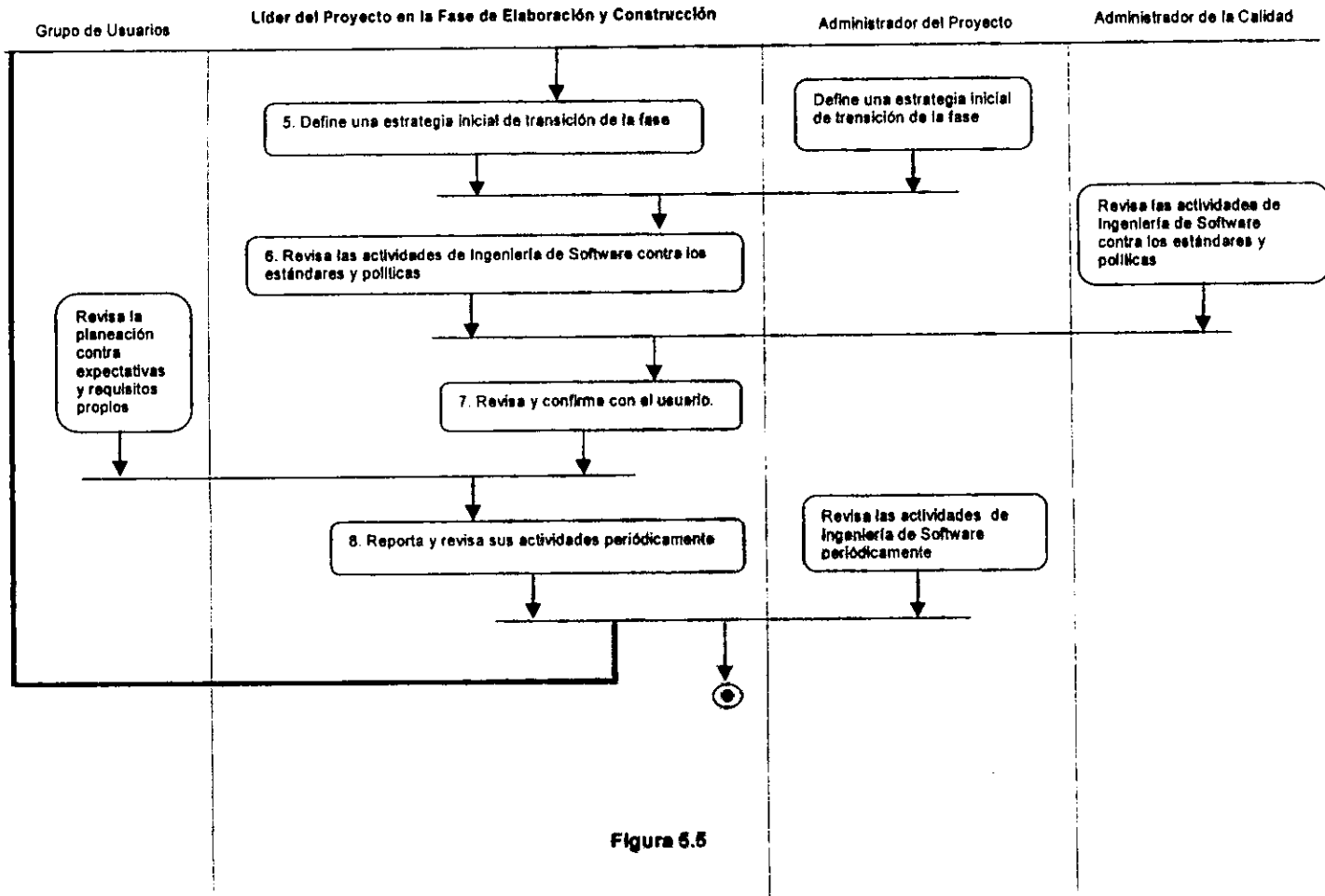


Figura 5.5

Actividades del Líder del Proyecto en la Fase de Elaboración y Construcción

Actividad	Procesos del PMBOK	Métricas ¹⁹	Entradas	Métodos, técnicas e Instrumentos
	Actividades de CMM-2 ²⁰	Validaciones ²¹	Salidas	
1 Da seguimiento al Plan de Desarrollo con el Grupo de Ingeniería de Software según la fase actual	<ul style="list-style-type: none"> • 1.2 Ejecución del Plan del Proyecto • 6.3 Desarrollo del Equipo de trabajo 	<ul style="list-style-type: none"> • Trabajo y progreso • Equipo de trabajo y su dinámica • Requerimientos • Arquitectura inicial • Estabilidad de solución • Estabilidad de construcción • Requerimientos • Análisis y Diseño • Unidad de Software 	<ul style="list-style-type: none"> • Líneas base aceptadas de la Fase 	<ul style="list-style-type: none"> • Reuniones y entrevistas • Toma de decisiones. • CASE y Metodologías • Ingeniería de Requerimientos • Identificación de alternativas • Ciclos de Vida del SW. • Descomposición funcional • Modelado de la información
	<p><i>Requerimientos</i></p> <ul style="list-style-type: none"> • A1 El GIS revisa los requerimientos antes de ser incorporados al Plan del Proyecto • A2 El GIS usa los requerimientos como base para planes, productos y actividades 		<ul style="list-style-type: none"> • Reporte actual del Progreso del desarrollo del Proyecto • Planes actuales del Análisis, Diseño y Construcción 	
2 Da seguimiento al Plan de Desarrollo del Subcontratista según la fase actual	<ul style="list-style-type: none"> • 1.2 Ejecución del Plan del Proyecto • 6.3 Desarrollo del Equipo de trabajo • 9.5 Administración de contratos 	<ul style="list-style-type: none"> • Trabajo y progreso • Equipo de trabajo y su dinámica • Requerimientos • Arquitectura inicial • Estabilidad de solución • Estabilidad de construcción • Requerimientos • Análisis y Diseño • Unidad de Software 	<ul style="list-style-type: none"> • Líneas base aceptadas de la Fase 	<ul style="list-style-type: none"> • Reuniones y entrevistas • Toma de decisiones
	<p><i>Subcontratistas</i></p> <ul style="list-style-type: none"> • A6. Realiza los cambios al contrato acordado con el subcontratista • A8. Revisa y retroalimenta con subcontratista • A9. Revisa que los resultados del subcontratista cumplan con el Plan • A12. Realiza pruebas de aceptación de los productos del subcontratista • A13. Evalúa el rendimiento del subcontratista 		<ul style="list-style-type: none"> • Reporte actual del Progreso del desarrollo de Subcontratistas 	

¹⁹ Ver figura 2.20 y figura 2.24

²⁰ C (compromisos) A (actividades), V (Verificaciones)

²¹ Ver figura 2.23

3	Revisa y audita los entregables según la fase actual	<ul style="list-style-type: none"> 5.2. Aseguramiento de la Calidad 5.3 Control de la Calidad 8.4 Plan de contingencia 	<ul style="list-style-type: none"> Trabajo y progreso. Cambios y estabilidad Estabilidad de viabilidad Congruencia Seguimiento Defectos 	<ul style="list-style-type: none"> Reporte actual del Progreso del desarrollo del Proyecto Planes actuales del Análisis, Diseño y Construcción Reporte actual del Progreso del desarrollo de Subcontratistas 	<ul style="list-style-type: none"> Técnicas de evaluación y revisión. Procesos y cuestionarios para el aseguramiento de la calidad. Reuniones.
		<p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> A12. El GIS realiza revisiones internas 		<ul style="list-style-type: none"> Línea Base revisada 	
4	Solicita la actualización de las Líneas Base afectadas según la fase actual	<ul style="list-style-type: none"> 7.2 Distribución de la información 1.3 Control integral de cambios 		<ul style="list-style-type: none"> Línea Base revisada 	<ul style="list-style-type: none"> Formatos y estándares Gulas y Procedimientos
		<p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> A1. Documenta Plan de Desarrollo de SW A2. Asigna responsabilidades sobre actividades y productos <p><i>Subcontratistas:</i></p> <ul style="list-style-type: none"> A11. Configuración revisa las actividades de la Administración de la Configuración del subcontratista 	<ul style="list-style-type: none"> Comunicación e información 	<ul style="list-style-type: none"> Línea Base aceptada 	
5	Define una estrategia inicial de transición de la fase	<ul style="list-style-type: none"> 2.4. Verificación del alcance. 	<ul style="list-style-type: none"> Involucrados Seguimiento 		<ul style="list-style-type: none"> Reuniones. Control de proyectos Experiencia en toma de decisiones.
			<ul style="list-style-type: none"> Planeación y Estimación Comunicación e información 	<ul style="list-style-type: none"> Plan de Transición 	

6	<p>Revisa las actividades de Ingeniería de Software contra los estándares y políticas</p>	<ul style="list-style-type: none"> • 5.2 Aseguramiento de la Calidad • 5.3 Control de la Calidad <p><i>Requerimientos</i></p> <ul style="list-style-type: none"> • V3 Calidad revisa, audita y reporta las actividades y productos de la administración de Requerimientos <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> • V3 Calidad revisa, audita y reporta las actividades del seguimiento <p><i>Subcontratistas</i></p> <ul style="list-style-type: none"> • A10. Calidad revisa las actividades de aseguramiento de calidad del subcontratista • V3 Calidad revisa, audita y reporta las actividades del subcontratista. 	<ul style="list-style-type: none"> • Trabajo y progreso • Involucrados • Satisfacción <p>• Comunicación e Información</p>	<ul style="list-style-type: none"> • Reporte actual del Progreso del desarrollo del Proyecto • Planes actuales del Análisis, Diseño y Construcción • Reporte actual del Progreso del desarrollo de Subcontratistas • Estándares y políticas <p>• Reporte revisado del Progreso del desarrollo del Proyecto</p> <p>• Reporte revisado del Progreso del desarrollo de Subcontratistas</p> <p>• Planes revisados del Análisis Diseño y Construcción</p>	<ul style="list-style-type: none"> • Reuniones • Técnicas de evaluación y revisión • Procesos y cuestionarios para el aseguramiento de la calidad
7	<p>Revisa y confirma con el usuario</p>	<ul style="list-style-type: none"> • 5.2 Aseguramiento de la Calidad • 5.3 Control de la Calidad 	<ul style="list-style-type: none"> • Trabajo y progreso • Involucrados • Satisfacción <p>• Comunicación e Información</p>	<ul style="list-style-type: none"> • Planes revisados del Análisis Diseño y Construcción <p>• Planes aceptados del Análisis Diseño y Construcción</p>	<ul style="list-style-type: none"> • Entrevistas en grupos • Técnicas de presentación • Ayudas visuales estructuradas • Reportes

8	Reporta y revisa sus actividades periódicamente	<ul style="list-style-type: none"> • 5.2 Aseguramiento de la Calidad • 5.3 Control de la Calidad 	<ul style="list-style-type: none"> • Trabajo y progreso • Involucrados • Satisfacción 	<ul style="list-style-type: none"> • Planes aceptados del Análisis, Diseño y Construcción • Reporte revisado del Progreso del desarrollo del Proyecto • Reporte revisado Progreso del desarrollo de Subcontratistas 	<ul style="list-style-type: none"> • Reuniones. • Técnicas de evaluación y revisión. • Procesos y cuestionarios para el aseguramiento de la calidad.
		<p><i>Requerimientos:</i></p> <ul style="list-style-type: none"> • V1. Revisa las actividades y productos de la administración de Requerimientos con el Administrador Principal • V2. Revisa las actividades y productos de la administración de Requerimientos con el Administrador del Proyecto <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> • V1. Revisa las actividades del seguimiento con el Administrador Principal • V2. Revisa las actividades del seguimiento con el Administrador del Proyecto <p><i>Subcontratistas:</i></p> <ul style="list-style-type: none"> • V1. Revisa las actividades del subcontratista con el Administrador Principal • V2. Revisa las actividades del subcontratista con el Administrador del Proyecto 	<ul style="list-style-type: none"> • Comunicación e Información. 		

Tabla 5.5

CAPÍTULO 6. Fase de Transición.

Este es el tercer capítulo que conforma la guía para la administración de proyectos de software. Abarca la última fase del Ciclo de Vida que se presentó en la sección 3.1.1: la de Transición.

Como ya se indicó en la unidad 3, la guía se presenta usando marcos (ver la sección 3.2.1), patrones (ver la sección 3.2.2) y una tabla resumen (ver la sección 3.2.3). Cada patrón hace referencia a un rol (ver la sección 3.1.2), los cuales están relacionados directamente con las actividades del Administrador Principal y, en forma gráfica, estos patrones se encuentran detallados mediante un diagrama de actividades (ver el Anexo A).

El caso de esta fase es similar a las dos anteriores (Elaboración y Construcción), diferenciándose en las actividades que conforman la terminación de un proyecto (el Mantenimiento puede verse como un segundo proyecto pactado con el usuario o bien, como parte del uso y producción ya bajo la responsabilidad del cliente). Se ve aquí que la mayoría de actividades realizadas en las fases anteriores se repiten en esta, por lo que su descripción sólo hace referencia al lugar donde fueron mencionadas por primera vez, y se añaden aquellas actividades propias de la finalización y entrega del producto construido.

Esto mismo sucede con los roles utilizados en esta fase. Tanto los roles del administrador de la Configuración, el de la Calidad y el del Líder de Proyecto, conservan el mismo comportamiento (desde el punto de vista administrativo) que los descritos en las fases de Elaboración y Construcción, por lo que en este capítulo sólo se hace referencia a los roles del administrador Principal y al del Proyecto.

6.1. Marco para la Fase de Transición.

Procesos	De Control. de Ejecución y de Finalización	
Objetivos	<ul style="list-style-type: none"> • Liberar el producto realizado y aprobado a los usuarios • Implantar el sistema en su ambiente final de trabajo • Verificar el resultado de la construcción y de la implantación. • Entregar toda la documentación necesaria para usar y mantener el sistema • Capacitar y entrenar a todos los usuarios que estarán en contacto con el sistema • Realizar las pruebas pertinentes que avalen la calidad del producto • Migrar o preparar la información necesaria para la operación del sistema • Iniciar el soporte y la asistencia al usuario • Realizar los ajustes últimos del sistema en las instalaciones del usuario. 	
Compuesta por:	<ul style="list-style-type: none"> • El 30 % de la • El 100 % de la • El 25 % de las • El 25 % de la 	<ul style="list-style-type: none"> Implementación. Implantación Pruebas. Documentación.
Tiempo utilizado:	<ul style="list-style-type: none"> • El 20 % del total 	

Factores críticos de Éxito	<ul style="list-style-type: none"> • Participación activa y comprometida de los usuarios. • Verificación de la existencia y de la calidad de los entregables de la fase anterior. • Control total del trabajo en detalle para seguir los calendarios planeados. • Reuniones exitosas de retroalimentación. • Adaptabilidad a cambios imprevistos de circunstancias y objetivos. • Crear un plan de capacitación apropiado y efectivo para todo los tipos de usuarios • Asegurar que el usuario realice y acepte todas las pruebas del sistema. • Planear una instalación coordinada junto con el personal del usuario. • Asegurar el calendario de trabajo con los usuarios y desarrolladores clave. • Asegurar que la integración del sistema coexista adecuadamente con otros sistemas relacionados.
----------------------------	--

Roles:	<ul style="list-style-type: none"> • Administrador Principal. • Administrador del Proyecto de Software (Administrador del Proyecto). • Administrador de Configuraciones. • Administrador de Calidad. • Líderes de Proyecto. • Grupos de Ingeniería de Software.
--------	---

Entregables de la Fase:

Entradas:	<ul style="list-style-type: none"> • Producto de Software construido • Manuales de Usuario y de Operación • Plan de Desarrollo del Proyecto • Plan de Configuración. 	<ul style="list-style-type: none"> • Resultado de todas las pruebas. • Plan de Transición aceptado. • Plan de Calidad. • Plan de Contingencia
Salidas:	<ul style="list-style-type: none"> • Sistema instalado y totalmente funcional • Plan de capacitación • Datos migrados o convertidos • Documentación completa del sistema • Adecuaciones para el mantenimiento 	<ul style="list-style-type: none"> • Plan de entrega del producto • Material de capacitación • Bitácora de fallas menores • Reporte de la Instalación • Plan de Transición aceptado

Subprocesos:	<ul style="list-style-type: none"> • De Control: <ul style="list-style-type: none"> • 7.3.Reporte del seguimiento. • 1.3.Control integral de cambios • 2.6.Control de cambios al alcance • 3.6.Control del calendario • De Ejecución: <ul style="list-style-type: none"> • 1.2.Ejecución del plan del proyecto • 7.2.Distribución de la información • 9.6.Administración de contratos • De Finalización: <ul style="list-style-type: none"> • 9.6.Terminación de contratos 	<ul style="list-style-type: none"> • 4.4.Control de costos • 6.3.Control de calidad • 8.4.Plan de contingencia • 6.3.Desarrollo del equipo de trabajo • 6.2.Aseguramiento de la calidad • 7.4.Control de la terminación administrativa
--------------	--	--

Métricas:	<ul style="list-style-type: none"> • Trabajo y Progreso. • Estimación y planeación. • Cambios y estabilidad. • Estropicios y modularidad. 	<ul style="list-style-type: none"> • Costo presupuesta y gastos. • Trabajo duplicado y adaptabilidad. • Equipo de trabajo y su dinámica.
-----------	---	---

Criterios de Evaluación:	<ul style="list-style-type: none"> • Involucrados • Estabilidad de visibilidad • Riesgos • Defectos 	<ul style="list-style-type: none"> • Transición • Estabilidad de solución • Seguimiento • Satisfacción
--------------------------	---	--

Verificaciones	<ul style="list-style-type: none"> • Estrategia con individuos externos • Estrategia con Admón. Principal • Estrategia con Admón. de Calidad • Comunicación e Información • Compromisos, motivación y capacitación • Unidad de Software • Integración • Instalación 	<ul style="list-style-type: none"> • Estrategia con Admón. del Proyecto • Estrategia con Administración de la Configuración. • Estrategia con Líderes de Proyecto. • Planeación y estimación. • Modelo del Ciclo de Vida. • Análisis y Diseño • Sistema
Procedimientos Métodos, Técnicas e Instrumentos	<ul style="list-style-type: none"> • Reuniones • Reportes de Progreso • Experiencia en toma de decisiones • Listas de verificación • Valor monetario esperado. • Análisis, conocimiento y habilidades de los involucrados • Creación de reportes • SW de administración de proyectos 	<ul style="list-style-type: none"> • Análisis de riesgos. • Planeación de proyectos • Compresión de tiempos • Estimaciones análogas. • Prácticas de recursos humanos y reclutamiento • Uso de ayudas visuales para presentaciones
Patrones aplicables:	<ul style="list-style-type: none"> • Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98]. • Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98]. • Definición de Documentos Administrativos Iniciales (<i>Define the Initial Management Documents</i>) [AMB98]. • El Modelo (<i>Model</i>) [AMB98]. • La fase de Construcción (<i>The Construct phase</i>) [AMB98]. • Programa (<i>Program</i>) [AMB98]. • Generalización (<i>Generalize</i>) [AMB98]. • Prueba en lo pequeño (<i>Test in the small</i>) [AMB98]. • El Administrador Principal en la fase de Transición • El Administrador del Proyecto en la fase de Transición • El Administrador de Configuración en la fase de Elaboración y Construcción • El Administrador de la Calidad en la fase de Elaboración y Construcción • Los Líderes de Proyecto en la fase de Elaboración y Construcción 	
Antipatrones:	<ul style="list-style-type: none"> • Miedo al Éxito (<i>Fear of Success</i>) [BRO98]. • Mazorca (<i>Corncock</i>) [BRO98]. • Violencia Intelectual (<i>Intellectual Violence</i>) [BRO98]. • Administración Irracional (<i>Irrational Management</i>) [BRO98]. • Administración incorrecta del Proyecto (<i>Project Mismanagement</i>) [BRO98]. • Arrojarlo en la pared (<i>Throw it over the wall</i>) [BRO98]. • Taladro de fuego (<i>Fire Drill</i>) [BRO98]. • El odio (<i>The Feud</i>) [BRO98]. 	
Área clave del CMM	<ul style="list-style-type: none"> • Planeación del Proyecto de Software • Administración de Requerimientos. • Aseguramiento de la Calidad del Software • Administración de la Configuración del Software. • Administración de Subcontratistas • Supervisión y seguimiento del Proyecto de Software. 	

6.2. Patrones por rol para la Fase de Transición.

EL ADMINISTRADOR PRINCIPAL EN LA FASE DE TRANSICIÓN

Nombre del Patrón:	El Administrador Principal en la fase de Transición
Problema	Queremos determinar las actividades y responsabilidades del Administrador Principal en la fase de Transición de un proyecto software.
Relacionado con:	<ul style="list-style-type: none">• Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98].• El Administrador del Proyecto en la fase de Transición• El Administrador de la Calidad en la fase de Elaboración y Construcción
Solución.	La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades ¹ de la figura 6.1; y las actividades correspondientes están detalladas en la tabla 6.1.

ADMINISTRADOR DEL PROYECTO EN LA FASE DE TRANSICIÓN

Nombre del Patrón:	El Administrador del Proyecto en la fase de Transición
Problema:	Queremos determinar las actividades y responsabilidades del Administrador del Proyecto de Software (Administrador del Proyecto) en la fase de Transición del desarrollo del Proceso de Software.
Relacionado con:	<ul style="list-style-type: none">• Definición y Validación de Requerimientos Iniciales (<i>Define and Validate Initial Requirements</i>) [AMB98].• Definición de la infraestructura del proyecto (<i>Define the Project Infrastructure</i>) [AMB98].• El Modelo (<i>Model</i>) [AMB98].• La fase de Construcción (<i>The Construct phase</i>) [AMB98].• Programa (<i>Program</i>) [AMB98].• Generalización (<i>Generalize</i>) [AMB98].• Prueba en lo pequeño (<i>Test in the small</i>) [AMB98].• El Administrador Principal en la fase de Transición• El Administrador de la Configuración del Software en la fase de Elaboración y Construcción• El Administrador de la Calidad en la fase de Elaboración y Construcción• El Líder del Proyecto en la fase de Elaboración y Construcción
Solución.	La solución propuesta se encuentra esquematizada mediante el Diagrama de Actividades ² de la figura 6.2; y las actividades correspondientes están detalladas en la tabla 6.2.

¹ Vea el anexo A.

² *Idem.*

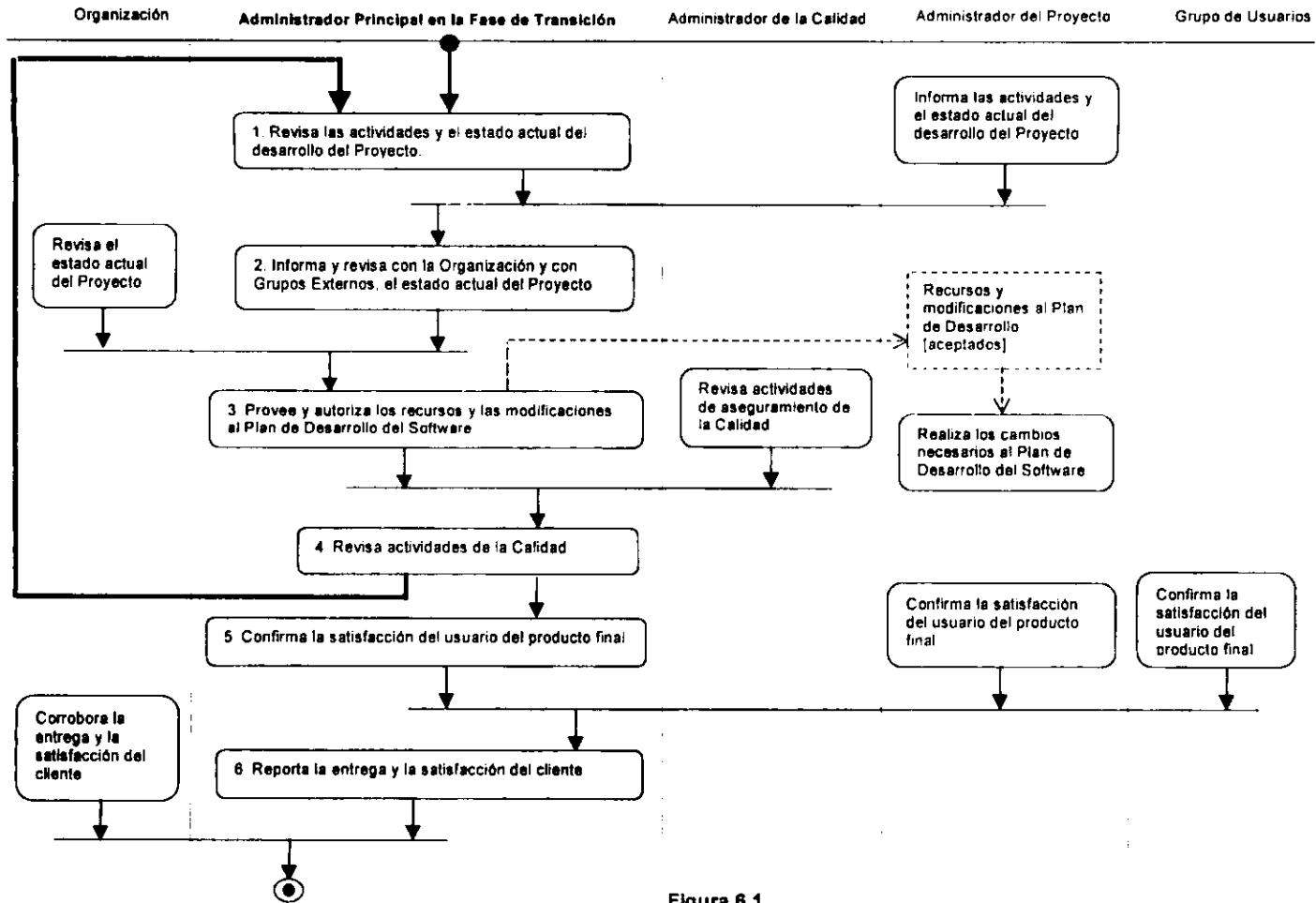


Figura 6.1

Actividades del Administrador Principal en la Fase de Transición

Actividad ³	Procesos del PMBOK	Métricas ⁴	Entradas	Métodos, técnicas e instrumentos
	Actividades de CMM-2 ⁵	Validaciones ⁶	Salidas	
5 Confirma la satisfacción del usuario del producto final	<ul style="list-style-type: none"> • 5.3 Control de Calidad • 7.4 Control de la terminación administrativa 	<ul style="list-style-type: none"> • Involucrados • Satisfacción 	<ul style="list-style-type: none"> • Documento de aceptación del producto. 	<ul style="list-style-type: none"> • Reuniones y entrevistas
	<p><i>Planeación:</i></p> <ul style="list-style-type: none"> • A4. La Administración principal revisa los compromisos con individuos y grupos externos <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> • A3. La Administración principal revisa los compromisos del Proyecto y los cambios acordados con individuos y grupos externos. 	<ul style="list-style-type: none"> • Estrategia, con grupos externos 		
6 Reporta la entrega y la satisfacción del cliente	<ul style="list-style-type: none"> • 5.3 Control de Calidad • 7.4 Control de la terminación administrativa 	<ul style="list-style-type: none"> • Involucrados • Satisfacción 	<ul style="list-style-type: none"> • Documento de aceptación del producto. 	<ul style="list-style-type: none"> • Reuniones y entrevistas
	<p><i>Planeación:</i></p> <ul style="list-style-type: none"> • A4 La Administración principal revisa los compromisos con individuos y grupos externos <p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> • A3. La Administración principal revisa los compromisos del Proyecto y los cambios acordados con individuos y grupos externos. 	<ul style="list-style-type: none"> • Estrategia, con la organización 	<ul style="list-style-type: none"> • Recursos y modificaciones al Plan de Desarrollo 	

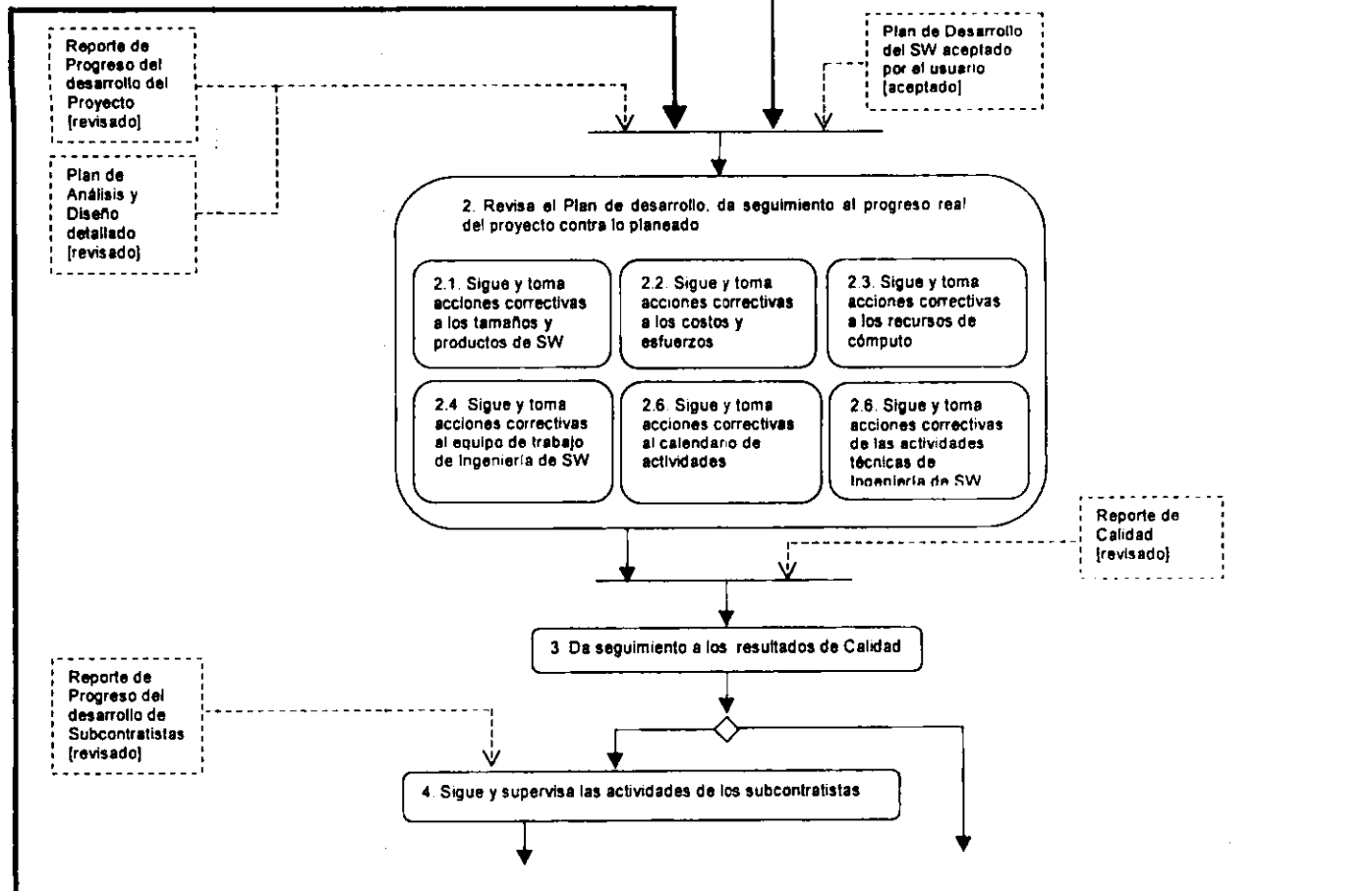
Tabla 6.1

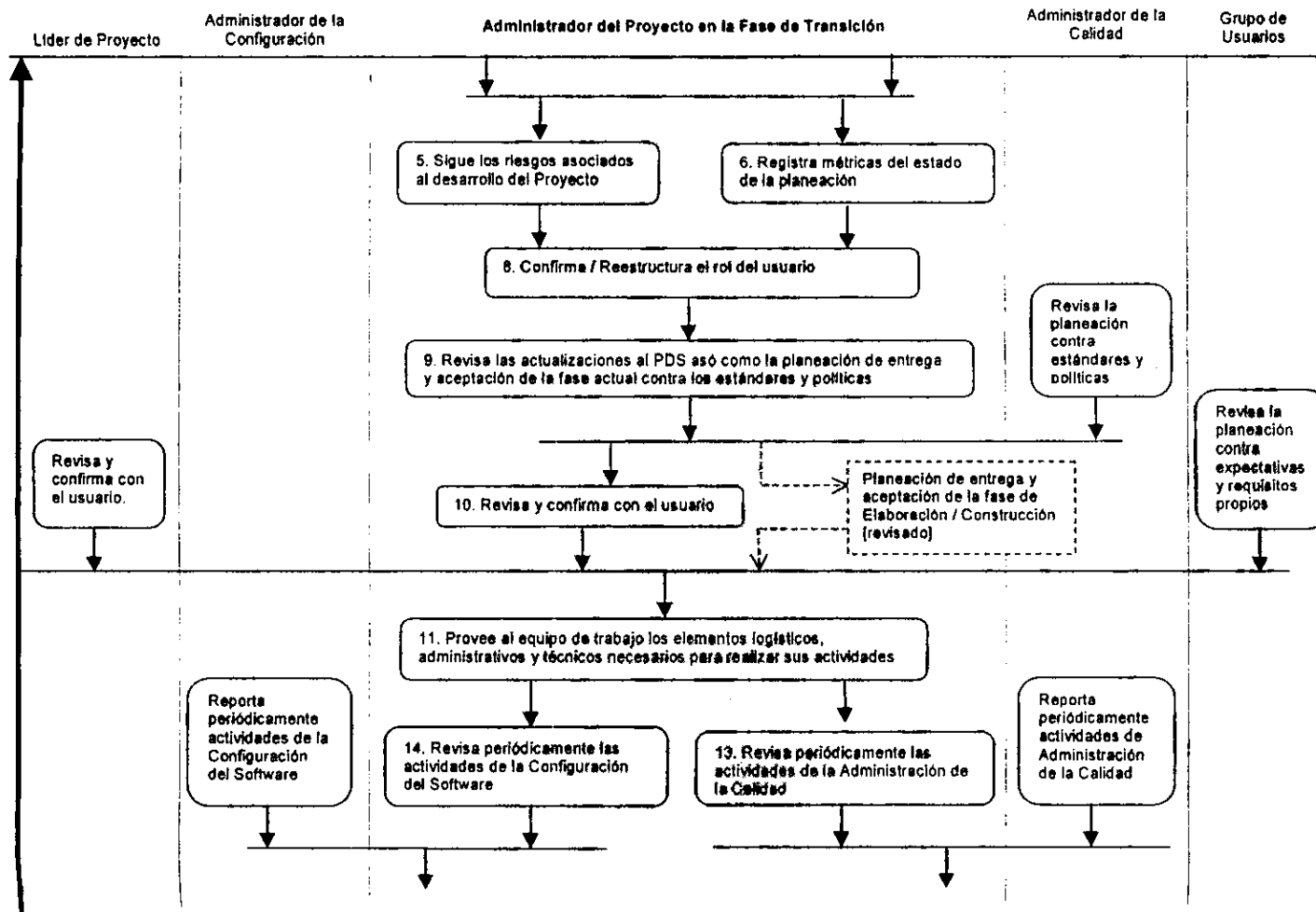
³ Las actividades 1 a 5 están detalladas en las "Actividades del Administrador Principal en la Fase de Elaboración y Construcción" (Tabla 5.1)

⁴ Ver figura 2.20 y figura 2.24.

⁵ C (compromisos), A (actividades), V (Verificaciones)

⁶ Ver figura 2.23.





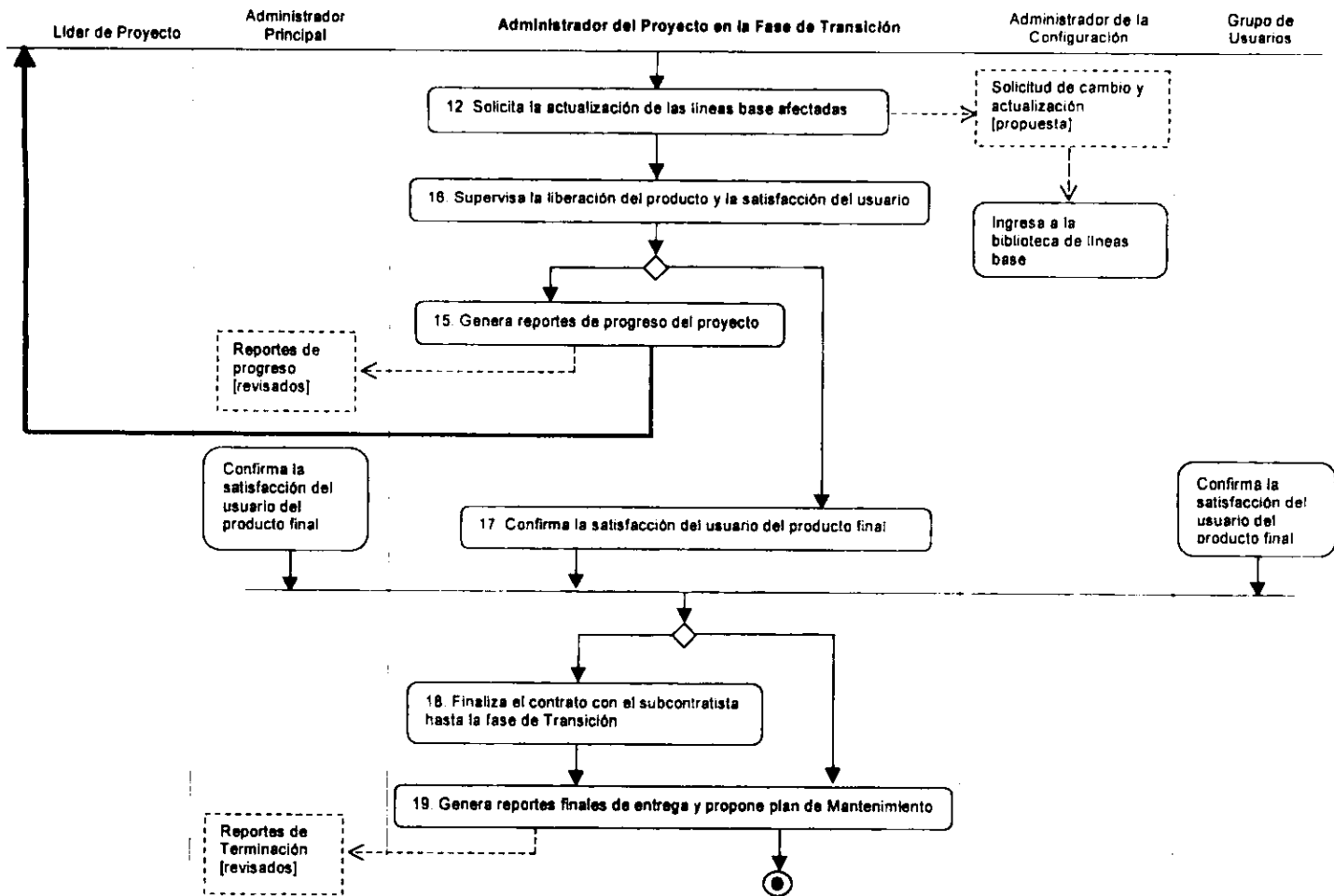


Figura 6.2

Actividades del Administrador del Proyecto en la Fase de Transición

	Actividad ⁷	Procesos del PMBOK	Métricas ⁸	Entradas	Métodos, técnicas e Instrumentos
		Actividades de CMM-2 ⁹	Validaciones ¹⁰	Salidas	
16	Supervisa la liberación del producto y la satisfacción del usuario	<ul style="list-style-type: none"> 5.3 Control de Calidad 7.4 Control de la terminación administrativa 	<ul style="list-style-type: none"> Costo presupuestado y gastos Involucrados Satisfacción 	<ul style="list-style-type: none"> 	<ul style="list-style-type: none"> Reuniones y entrevistas Análisis, conocimiento y habilidades de los involucrados.
		<p><i>Seguimiento:</i></p> <ul style="list-style-type: none"> A13. Revisa formalmente que los resultados cumplan con lo fijado en el Proyecto 	<ul style="list-style-type: none"> Estrategia, con Administrador Principal Estrategia, con grupos externos Integración Sistema Instalación 	<ul style="list-style-type: none"> Sistema Instalado Capacitación al usuario Documentación completa Bitácora de fallas menores Reporte de la instalación Datos migrados o convertidos Adecuaciones para el mantenimiento 	
17	Confirma la satisfacción del usuario del producto final	<ul style="list-style-type: none"> 5.3 Control de Calidad 7.4 Control de la terminación administrativa 	<ul style="list-style-type: none"> Costo presupuestado y gastos Involucrados Satisfacción 	<ul style="list-style-type: none"> Sistema Instalado Capacitación al usuario Documentación completa Bitácora de fallas menores Reporte de la instalación Datos migrados o convertidos Adecuaciones para el mantenimiento 	<ul style="list-style-type: none"> Reuniones y entrevistas Análisis, conocimiento y habilidades de los involucrados.
		<p><i>Planeación:</i></p> <ul style="list-style-type: none"> A4. Revisa los compromisos con individuos y grupos externos. 	<ul style="list-style-type: none"> Estrategia, con Administrador Principal Estrategia, con grupos externos Instalación 	<ul style="list-style-type: none"> Aceptación formal del usuario 	

⁷ Las actividades de la 2 a la 6 y de la 8 a la 15 están detalladas en las "Actividades del Administrador del Proyecto en la Fase de Elaboración y Construcción" (Tabla 5.2)

⁸ Ver figura 2.20 y figura 2.24.

⁹ C (compromisos), A (actividades), V (Verificaciones).

¹⁰ Ver figura 2.23.

18	Finaliza el contrato con el subcontratista hasta la fase de Transición	<ul style="list-style-type: none"> 7.4 Control de la terminación administrativa 9.6 Terminación de contratos 	<ul style="list-style-type: none"> Trabajo y progreso Costo presupuestado y gastos Equipo de trabajo y su dinámica Estabilidad de construcción 	<ul style="list-style-type: none"> Reporte revisado de Progreso del desarrollo de Subcontratistas Contrato con el subcontratista 	<ul style="list-style-type: none"> Reuniones Experiencia en toma de decisiones Medidas de rendimiento Administración de contratos
		<p><i>Subcontratistas</i></p> <ul style="list-style-type: none"> A3 Usa el contrato con el contratista para su administración. A4. Revisa el plan de desarrollo del subcontratista. A6 Usa el plan del subcontratista para supervisar sus actividades A7. El administrador del Proyecto revisa periódicamente la administración de subcontratistas 	<ul style="list-style-type: none"> Integración Sistema Instalación 	<ul style="list-style-type: none"> Solicitud de ajustes y cambios al Plan de Desarrollo de Subcontratistas Terminación formal del contrato con el subcontratista hasta la fase de Transición 	
19	Genera reportes finales de entrega y propone plan de Mantenimiento	<ul style="list-style-type: none"> 7.3 Reporte del seguimiento 7.2 Distribución de la información 6.2 Aseguramiento de la calidad 			<ul style="list-style-type: none"> Uso de ayudas visuales para presentaciones Creación de reportes Software de administración de proyectos Análisis, conocimiento y habilidades de los involucrados
		<p><i>Planeación</i></p> <ul style="list-style-type: none"> V1. Revisa periódicamente las actividades con el Administrador Principal <p><i>Seguimiento</i></p> <ul style="list-style-type: none"> A3. Administración Principal revisa cambios y compromisos al Plan de Desarrollo 	<ul style="list-style-type: none"> Comunicación e información 	<ul style="list-style-type: none"> Reportes de Terminación 	

Tabla 6.2.

CONCLUSIONES

Tomando en cuenta el objetivo que dio origen a este trabajo es decir, el crear una Guía de apoyo para la Administración de Proyectos, usando las mejores prácticas propuestas por el Cuerpo de Conocimiento de la Administración de Proyectos, (*Project Management Body of Knowledge PMBOK*) del Instituto de Administración de Proyectos (*Project Management Institute PMI*) y cumpliendo con todas las áreas claves del nivel 2 del Modelo CMM (*Capability Maturity Model for Software*) del Instituto de Ingeniería de Software (*Software Engineering Institute, SEI*) de la Universidad de Carnegie Mellon, podemos concluir lo siguiente:

- Que esta tesis representa la síntesis y abstracción de varias líneas de trabajo teóricas y prácticas de la Ingeniería de Software. Aquí se revisaron conceptos y propuestas de las áreas principales que atañen a la administración de proyectos de software como el ciclo de vida de un proyecto, fases, productos entregables, métricas, evaluaciones y pruebas, y también se revisó el modelo de calidad CMM y la propuesta del Instituto de Administración de Proyectos de sus mejores prácticas.
- Que se espera que el uso de marcos y patrones en la descripción de los procesos de esta guía, resulte un medio claro y eficaz para lograr un grado adecuado de entendimiento, sin perder la flexibilidad de aplicación de dichos procesos según la naturaleza de cada proyecto. Como se usaron también patrones genéricos, creemos que estos pueden agrupar y simplificar los procesos que presentan la misma naturaleza y evitar así, la sobrecarga de información ganando mayor abstracción y entendimiento del proceso en sí.
- Que el modelo de mejora de la capacidad para desarrollar productos de software usado en este trabajo, permite tener una visión más general y completa de áreas claves relegadas en la práctica, pero cuya falta puede provocar serios retrasos en la planeación. Dos áreas fundamentales son la Administración de la Configuración y la Administración de la Calidad.
- Que las métricas, las pruebas y las validaciones, tienen un fin común, y es el lograr depurar el Proceso de Desarrollo, de tal forma que se asegure la creación de productos de calidad, utilizando de la mejor manera todos los recursos disponibles, humanos y materiales. Las métricas y las validaciones son elementos fundamentales para la administración de proyectos, pues es este el único camino que se tiene para corroborar el cumplimiento calendanzado de los objetivos que dieron origen a un proyecto.
- Que las habilidades básicas de un buen administrador del proyecto, están en la capacidad de asimilar y aprender de proyectos anteriores, así como de contar con la suficiente disciplina para seguir y verificar que se siga, una metodología fundamentada en las áreas principales de la Ingeniería de Software.

TRABAJOS POSTERIORES

Como un trabajo posterior, estaría el llevar esta guía a las consultorías de cómputo, y evaluarla en forma práctica, en diferentes proyectos, ambientes de desarrollo y equipos de trabajo. La información que estas pruebas arrojen, serviría de retroalimentación para afinar y mejorar los aspectos que esta guía cubre.

El utilizar esta guía en un proyecto real, podría ayudar a demostrar (cosa que creo *a priori*), que el seguimiento de una metodología de administración y control de proyectos de software, aunado al conocimiento de las reglas del negocio, pueden hacer posible la realización de un producto de software que cumpla las expectativas de costo y tiempo.

También, tomando en cuenta el principio fundamental del nivel 2 del CMM, se podría verificar si efectivamente que el aprovechar la experiencia de proyectos anteriores, realizados por uno mismo o por terceros y que fueron documentados, permitirá conformar una base de conocimiento que depurada, puede convertirse en reglas que aseguren la mejora en la práctica de la administración de proyectos.

Una vez que se cuente con dicha base de conocimiento, esta debería ser probada para comprobar que su uso puede evitar problemas y riesgos no previstos desde un principio en un proyecto; y mejorar y completar su planeación, estimación y seguimiento, de manera formal y eficiente.

Otra actividad que se desprende de este trabajo, sería el realizar las guías propias para cada rol en particular, y al nivel de detalle que sea necesario. Esto sería particularmente útil principalmente, para los roles que integran el Grupo de Ingeniería de Software, así como para los Líderes del Proyecto. Aquí se presentaron las actividades de este último pero desde el enfoque del administrador de proyectos, por lo que sería interesante poder atacar a cada rol a detalle y en función de la fase en la que se encuentre.

Esta guía puede verse reforzada y complementada, si se propusieran formatos de toda la documentación básica que el administrador de proyectos crea y utiliza a lo largo de la vida de un proyecto. Estos formatos de documentos deberían contener todos los aspectos básicos de las negociaciones con el usuario, así como la información del seguimiento y supervisión del desarrollo del proyecto (avances, retrocesos, métricas, riesgos, modificaciones, defectos, etc.). Al respecto puede estudiarse la propuesta dada en [CAN98].

Finalmente, como un trabajo posterior quedaría, el proponer, crear y describir, ya en forma particular y detallada, los procesos de apoyo a la administración de proyectos, estos son: el Proceso de Pruebas, el Proceso de la Toma de Métricas, el Proceso de la Estimación y el de la Documentación.

BIBLIOGRAFÍA

Administración de Proyectos.

- FUG95 Fuggetta, A.: *Il Processo Software. Aspetti strategici e organizzativi*; il Cardo editore in Venezia, Italia 1995
- OKT98 Oktaba, Hanna; Ibaranguoiba Guadalupe *Software Process Modeled with Objects: Static View*. Computación y Sistemas Vol. 1 No. 4, pp. 228-238; México, 1998
- PRE97 Pressman, Roger S.: *Software engineering A practitioner's approach*; McGraw Hill; 5ª edición. EUA 1997
- PRE98 Pressman, Roger S.; *Ingeniería del Software. un enfoque práctico*; McGraw Hill; 5ª edición. España 1998.
- RON92

Cuerpo de Conocimiento de la Administración de Proyectos.

- PMI96 Project Management Institute. *A Guide to the Project Management Body of Knowledge*; EUA 1996.

La administración de proyectos en el nivel 2 del CMM.

- PAU95 Paulk, Mark C; Weber Charles V, et al; *The Capability Maturity Model: Guidelines for Improving the Software Process*. Addison – Wesley; EUA 1995.

Pautas para el proceso de desarrollo.

Paradigma Estructurado

- BAR90 Barker, Richard; *CASE Method: tasks and deliverables*; Addison-Wesley Publishing Company; EUA 1990
- GIL98 Gil, Blanca, et al; *Sistema Integral de Contabilidad de una casa postproductora de cine y video*. Tesis de Licenciatura. Facultad de Ingeniería, UNAM, México 1998.
- PRE97 Pressman, Roger S.; *Software engineering. A practitioner's approach*; McGraw Hill; 5ª edición; EUA 1997
- PRE98 Pressman, Roger S.; *Ingeniería del Software. un enfoque práctico*; McGraw Hill; 5ª edición. España 1998.

Paradigma Orientado a Objetos

- BER86 Berzins, V; Gray, M; Naumann, D; *Abstraction – Based Software*; Communications of the ACM vol. 29; EUA mayo de 1986
- BOO91 Booch Grady; *Object oriented design with applications*; The Benjamin Cummings Publishing Company; EUA 1991.

- BOO96 Booch Grady; *Análisis y diseño orientado a objetos con aplicaciones*; Addison - Wesley / Diaz De Santos; 2ª Edición; EUA 1996.
- CAN98 Cantor, Murray; *Object-oriented project management with UML*; Wiley Computer Publishing, EUA 1998.
- DAH72 Dahl, O; Dijkstra, F; Hoare C.A.R.; *Structured Programming*; Academic Press, London, 1972.
- GHE97 Ghezzi, Carlo; Jazayeri, Mehdi; *Programming language concepts*; John Wiley & Sons; 3ª Edición; EUA 1997.
- IGA76 Igalls, D.; *The Smalltalk-76 Programming System Design and Implementation*; Proceedings of the Fifth Annual ACM Symposium on Principles of Programming Languages; EUA 1976.
- JAC99 Jacobson, Ivar; Booch Grady; Rumbaugh, James; *The Unified Software Development Process*; Addison Wesley; EUA 1999.
- KAP91 Grupo Kapellmann; *Developer Essentials*; México 1991.
- SHA84 Shaw, M; *Abstraction Techniques in Modern Programming Languages*; IEEE Software vol. 1; EUA octubre de 1984.

Métricas

- CAN98 Cantor, Murray; *Object-oriented project management with UML*; Wiley Computer Publishing, EUA 1998.
- PRE97 Pressman, Roger S.; *Software engineering. A practitioner's approach*; McGraw Hill; 5ª edición; EUA 1997.
- ROY98 Royce, Walker; *Software project management: a unified framework*; The Addison - Wesley object technology series; EUA 1998.

Evaluación y Pruebas

- KOO99 Koomen, Tim; Pol, Martin; *Test Process Improvement*; Addison - Wesley; Inglaterra 1999.
- ROY98 Royce, Walker; *Software project management: a unified framework*; The Addison - Wesley object technology series; EUA 1998.
- SOF98 *Little Book of Testing. Volumen II Implementation Techniques*; Software Program Manager Network; EUA 1998.

Marcos y Patrones

- AMB98 Ambler, Scott W; *Process Patterns*; Cambridge University Press, SIGS Books, EUA 1998.
- AMB99 Ambler, Scott W; *The Process Patterns Resource Page*; <http://www.amblysoft.com/processPatternsPage.html>
- APP97 Appleton, Brad; *Patterns for Conducting Process Improvement*, PloP; EUA 1997.
- APP97 Appleton, Brad; *Patterns and Software: Essential Concepts and Terminology*, Object Magazine Online; Mayo Vol. 3 No. 5; EUA 1997.
- BRO98 Brown J., William, Malveau C., Raphael et al; *AntiPatterns Refactoring Software, Architectures, and Projects in Crisis*; Wiley Computer Publishing; EUA 1998.

COA95 Coad, Peter et al: *Object models: strategies, patterns, and applications*: Prentice Hall: EUA 1995

Roles utilizados en esta guía.

BOO99 Jacobson, Ivar, Booch Grady, Rumbaugh, James: *The Unified Modeling Language User Guide*. Addison Wesley. EUA 1999

PAU95 Paulk, Mark C, Weber Charles V, et al. *The Capability Maturity Model. Guidelines for Improving the Software Process*. Addison – Wesley. EUA 1995

Anexo A. Diagrama de Actividad.

RUM99 Rumbaugh, James, Jacobson Ivar, Booch Grady. *The Unified Modeling Language Reference Manual*. Addison – Wesley, EUA 1999

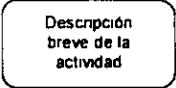



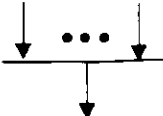
ANEXO

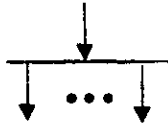
A. DIAGRAMAS DE ACTIVIDAD

Un diagrama de actividades es un medio para graficar el flujo de trabajo entre procesos. Como su nombre lo define, presenta la gráfica de actividades. Es un caso especial de un diagrama de estados (máquina de estados) donde la mayoría de los estados son estados de actividad o estados de acciones cuyas transiciones son ejecutadas cuando se completa la actividad. Este diagrama enfatiza los pasos secuenciales y concurrentes de un procedimiento. De aquí que el uso principal del diagrama de actividad, es el modelar los pasos de ejecución de un procedimiento [RUM99]

NOTACIÓN

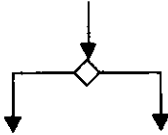
Los elementos que conforma un diagrama de actividades son

	Actividad	Es el elemento principal en un diagrama de actividades. Representa la acción o conjunto de acciones que se realizan sin interrupción. El nombre de la actividad describe brevemente el o los procesos que se ejecutan una vez que el flujo de acciones nos lleva a este elemento. Si no se especifica lo contrario, la actividad termina completamente su función antes de pasar a otra actividad.
	Transición o Flujo	Describe la secuencia de ejecución entre las actividades dentro de un diagrama. Representa el flujo de control entre actividades. Solamente un flujo puede entrar o salir de una actividad. Para tener diversas bifurcaciones tanto en la entrada como en la salida de una actividad, existen las uniones de control y los tenedores de control respectivamente.
	Estado inicial	Un estado inicial explícitamente muestra el comienzo de una máquina de estados. Sólo puede existir un estado inicial en cada diagrama de estados. En forma general, sólo una transición puede colocarse desde el estado inicial, sin embargo, pueden colocarse múltiples transiciones desde un estado inicial si al menos una de ellas está etiquetada con una condición.
	Estado final	Un estado final representa el estado terminal de un flujo. Especifica el final de la máquina de estado. Las transiciones sólo pueden entrar al estado final, una vez que el diagrama de actividades llega a un estado final, termina su existencia. Pueden existir más de un estado final en un diagrama de actividades.
	Unión de control	Elemento de sincronización que permite recibir más de un flujo de entrada para una actividad.



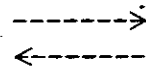
Tenedor de control

Elemento de sincronización que permite entregar más de un flujo de salida por una actividad.



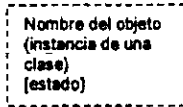
Decisión

Elemento de decisión que permite condicionar el flujo de procesos conforme a un criterio que puede o no cumplirse.



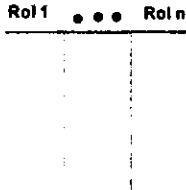
Valor de entrada o de salida

Representa un flujo de información o datos, que en sí es un objeto. Este flujo puede entrar o salir de una actividad, es decir utiliza el dato o lo crea o modifica respectivamente.



Objeto

Es el objeto que entra o sale de una actividad.



Carriles

Estos carriles especifican las áreas de acción para cada rol que interviene en el diagrama de actividades. Con estos elementos, se puede graficar la interrelación de las actividades en conjunto, denotando la sincronización, el paralelismo y la concurrencia, entre actividades que realizan diversos roles.