

# CAPÍTULO 8

## GESTIÓN DE LA INGENIERÍA DEL SOFTWARE

### ACRÓNIMOS

PMBOK	Guía al Proyecto de Gestión del Cuerpo de Conocimientos
SQA	Garantía de Calidad del Software

### INTRODUCCIÓN

La Gestión de la Ingeniería del Software puede definirse como la aplicación para actividades de gestión – planificación, coordinación, mediciones, monitoreo, control e informes – que asegure un desarrollo y mantenimiento del software sistemático, disciplinado y cuantificado (IEEE610.12-90).

El KA de Gestión de la Ingeniería del Software, por tanto, se encarga de la gestión y medición de la ingeniería del software. A pesar de que medir es un aspecto importante en todas las KAs, no es hasta aquí que se presenta el tema de programas de medición.

Aunque por una parte sea verdad afirmar que, en cierto sentido, debiera ser posible gestionar la ingeniería del software de la misma manera que cualquier otro proceso (complejo) existen aspectos específicos de los productos software y de los procesos del ciclo de vida del software que complican una gestión efectiva –sólo algunos de los cuales se apuntan a continuación:

- ♦ La percepción de los clientes es tal que con frecuencia existe una falta de aprecio de la complejidad inherente a la ingeniería del software, particularmente en relación al impacto que produce cambiar los requisitos.
- ♦ Es casi inevitable que los propios procesos de ingeniería del software generen la necesidad de nuevos o modificados requisitos del cliente.
- ♦ Como resultado, el software se construye con frecuencia mediante un proceso iterativo en vez de mediante una secuencia de tareas cerradas.
- ♦ La ingeniería del software incorpora necesariamente aspectos de creatividad y de disciplina – mantener un balance apropiado entre los dos es con frecuencia difícil.
- ♦ El grado de novedad y de complejidad del software son con frecuencia extremadamente altos.
- ♦ La tasa de cambio de la tecnología subyacente es muy rápida.

Con respecto a la ingeniería del software, las actividades de gestión tienen lugar en tres niveles:

gestión organizacional y de infraestructura, gestión de proyectos, y programa de planificación y control de mediciones. Estos dos últimos se cubren con más detalle en la descripción de este KA. A pesar de todo, esto no va en detrimento de la importancia de los temas de gestión organizacional.

Dado que la unión con las disciplinas señaladas – obviamente, la de gestión – es importante, se describirá con más detalle que las otras descripciones del KA. Los aspectos de gestión organizacional son importantes en términos de su impacto sobre la ingeniería del software –en gestión de políticas, por ejemplo: políticas organizativas y estándares que proporcionan el marco en el que se desenvuelve la ingeniería del software. Puede que se necesite que estas políticas se vean afectadas por los requisitos de un software de desarrollo y mantenimiento efectivo, y puede que se necesite establecer un número de políticas específicas de ingeniería del software para una gestión eficaz de la ingeniería del software a nivel organizacional. Por ejemplo, normalmente se necesitan políticas para establecer procesos específicos a nivel organizacional o procedimientos para tareas de ingeniería del software tales como el diseño, la implementación, la estimación, el seguimiento y los informes. Tales políticas son esenciales, por ejemplo, para una gestión de la ingeniería del software eficaz a largo plazo, ya que establecen una base consistente sobre la que analizar actuaciones anteriores e implementar mejoras.

Otro aspecto importante de la gestión es la gestión del personal: las políticas y procedimientos para contratar, entrenar y motivar al personal y actuar como mentor del desarrollo de una carrera son importantes no sólo a nivel de proyecto sino también para el éxito a largo plazo de una organización. Todo el personal de desarrollo del software puede haber sido entrenado del mismo modo o puede presentar retos para la gestión del personal (por ejemplo, mantener el capital en un contexto en el que la tecnología subyacente sufre cambios continuos y rápidos). Con frecuencia también se menciona la gestión de la comunicación como un aspecto pasado por alto pero importante de la actuación de los individuos en un campo en el que es necesario un entendimiento preciso de las necesidades del usuario y de los complejos, requisitos y diseños. Finalmente, es necesaria la gestión de la cartera de trabajo, que es la capacidad de tener una visión general, no sólo del conjunto del software en desarrollo sino también del software que ya se está utilizando en la organización. Más aún, la reutilización del software es un factor

clave en el mantenimiento y mejora de la productividad y competitividad. Una reutilización eficaz requiere una visión estratégica que refleje el poder único y los requisitos de esta técnica.

Los ingenieros del software deben entender los aspectos de gestión que se encuentran influidos de modo singular por el software, y además conocer sus aspectos más generales, incluso en los primeros cuatro años tras graduarse, como está marcado en la Guía.

La cultura y comportamiento organizacionales y la gestión comercial funcional en términos de consecución de metas, aportan la gestión en cadena, la publicidad, la ventas y la distribución, todas ellas influyen, aunque sea indirectamente, en el proceso de ingeniería del software de una organización.

La noción de gestión de proyectos tiene que ver con esta KA, como “la construcción de artefactos de software útiles”, se gestiona por lo general como (quizás programas de) proyectos individuales. A este respecto, encontramos un amplio respaldo en la Guía al Proyecto de Gestión del Cuerpo de Conocimientos (PMBOK) (PMI00), que en sí misma incluye las siguientes KAs de gestión de proyectos: gestión de integración del proyecto, gestión de objetivos del proyecto, gestión del tiempo del proyecto, gestión del coste del proyecto, gestión de la calidad del proyecto, gestión de los recursos humanos del proyecto y gestión de las comunicaciones del proyecto. Está claro que todos estos temas tienen una relación directa con el KA de Gestión de la Ingeniería del Software. Sería imposible e inadecuado intentar duplicar aquí el contenido de la Guía de la PMBOK. En su lugar, sugerimos que el lector que esté interesado en la gestión de proyectos más allá de lo que es específico a los proyectos de ingeniería del software consulte la propia PMBOK. La gestión de proyectos también se encuentra en el capítulo sobre las Disciplinas Señaladas de la Ingeniería del Software.

El KA de Gestión de la Ingeniería del Software consiste tanto en el proceso de gestión del proyecto de software en sus primeras cinco subáreas, como en la medición de la ingeniería del software en su última subárea. Aunque estos dos temas se suelen considerar como distintos, y de hecho poseen muchos aspectos únicos en sí mismos, su gran cercanía ha llevado a que se les trate de manera conjunta en esta KA. Desafortunadamente, se comparte la percepción común de que la industria del software entrega sus productos tarde, por encima de lo presupuestado, y de pobre calidad e incierta funcionalidad. Una gestión regulada por la medición – un principio presupuesto en cualquier disciplina de verdadera ingeniería – puede ayudar a darle la vuelta a esta percepción. En esencia, una gestión sin medición, cualitativa y cuantitativa, da la sensación de falta de rigor, y medir sin gestionar da la sensación de una falta de fines o de contexto. De igual manera, sin embargo, gestión y medición sin conocimientos de expertos es

igualmente ineficaz, por lo que debemos tener cuidado para evitar poner un énfasis excesivo en los aspectos cuantitativos de la Gestión de Ingeniería del Software (GIS). Una gestión eficaz requiere la combinación tanto de números como de experiencia. Aquí se adoptan las siguientes definiciones de trabajo:

- ♦ El *proceso de gestión* se refiere a las actividades que se emprenden para asegurarse de que los procesos de ingeniería del software se realizan de una manera consistente con las políticas, objetivos y estándares de la organización.
- ♦ La *medición* se refiere a la asignación de valores y etiquetas a los aspectos de la ingeniería del software (productos, procesos, y recursos según los define [Fen98]) y a los modelos que se derivan de ellos, se hayan desarrollado estos modelos utilizando técnicas estadísticas, conocimientos expertos u otras técnicas.

Las subáreas de gestión del proyecto de ingeniería del software hacen un uso extensivo del subárea de gestión de ingeniería del software.

No es de extrañar que esta KA esté relacionada de cerca con otras en la Guía del SWEBOK, y sería de particular utilidad leer las siguientes descripciones del KA junto con ésta.

- ♦ Los Requisitos del Software, en donde se describen algunas de las actividades que tendrán que realizarse durante la fase de definición de Iniciación y Alcance del proyecto.
- ♦ La Gestión de Configuración del Software, ya que trata de la identificación, control, consideraciones de estado, y auditoría de la configuración del software, junto con la gestión de entregas y repartos del software
- ♦ El Proceso de Ingeniería del Software, porque los procesos y los proyectos están estrechamente relacionados (esta KA también describe la medición de procesos y productos).
- ♦ La Calidad del Software, ya que la calidad es un objetivo constante de la gestión y es una meta con muchas actividades que tiene que gestionarse.

## DESCOMPOSICIÓN DE LOS TEMAS DE GESTIÓN DE LA INGENIERÍA DEL SOFTWARE

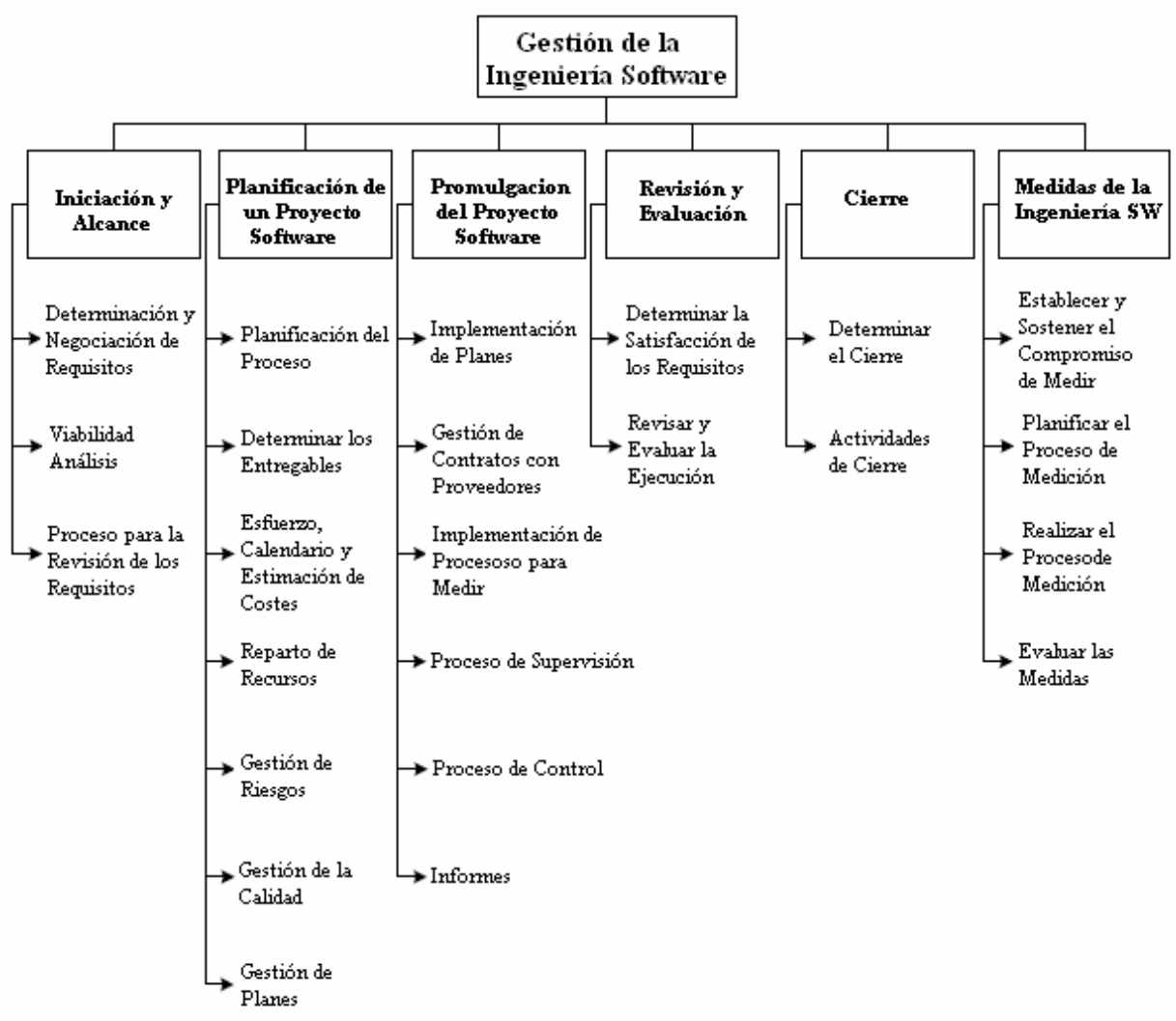
Hemos creado una descomposición basada tanto en temas como en ciclos de vida, ya que el KA de Gestión de Ingeniería del Software se ve aquí como un proceso organizacional que incorpora la noción de gestión de procesos y proyectos. Sin embargo, la base primaria de la descomposición de alto nivel es el proceso de gestionar un proyecto de ingeniería del software. Existen seis subáreas principales. Las primeras cinco subáreas siguen principalmente el

1 Proceso de Gestión IEEE/EIA 12207. Las seis  
2 subáreas son:

- 3 ♦ *Definición de iniciación y alcance*, que trata de
- 4 la decisión de iniciar un proyecto de ingeniería
- 5 del software.
- 6 ♦ *Planificación del proyecto de software*, que
- 7 afronta las actividades emprendidas para
- 8 prepararse para una ingeniería del software
- 9 exitosa desde una perspectiva de gestión.
- 10 ♦ *Promulgación del proyecto de software*, que trata
- 11 de las actividades de gestión de ingeniería del
- 12 software ampliamente aceptadas que tienen lugar
- 13 durante la ingeniería del software.

- 14 ♦ *Repaso y evaluación*, que buscan asegurarse de
- 15 que el software es satisfactorio.
- 16 ♦ *Cierre*, que afronta las actividades de post-
- 17 realización de un proyecto de ingeniería del
- 18 software.
- 19 ♦ *Medición de la ingeniería del software*, que trata
- 20 del desarrollo e implementación eficaz de los
- 21 programas de medición en las organizaciones de
- 22 ingeniería del software (IEEE12207.0-96)
- 23 La descomposición de los temas del KA de Gestión
- 24 de Ingeniería del Software se muestra en la Figura 1

**Figura 1** Descomposición de los temas del KA de Gestión de Ingeniería del Software



## 1. Iniciación y Alcance

El enfoque de este conjunto de actividades se centra en la determinación eficaz de los requisitos del software por medio de varios métodos de inducción y la valoración de la viabilidad del proyecto desde distintos puntos de vista. Una vez que se ha establecido la viabilidad, la tarea pendiente dentro de este proceso es la especificación de la validación de requisitos y del cambio de procedimientos (ver también el KA de Requisitos del Software).

### 1.1. Determinación y Negociación de los Requisitos

[Dor02: v2c4; Pfl01: c4; Pre04: c7; Som05: c5]

Los métodos de requisitos del software para la inducción de los requisitos (por ejemplo, observación), análisis (por ejemplo, modelado de datos, modelado de casos de uso), especificación y validación (por ejemplo, prototipado) deben seleccionarse y aplicarse, tomando en cuenta las distintas perspectivas del contratista. Esto lleva a la determinación del alcance del proyecto, de los objetivos, y de las restricciones. Ésta siempre es una actividad importante, ya que fija las fronteras visibles para el conjunto de tareas que se emprenden, y sucede así especialmente donde la tarea es de gran novedad. Puede encontrarse información adicional en el KA de los Requisitos del Software.

### 1.2. Viabilidad, Análisis (Técnico, Operacional, Financiero, Social/Político)

[Pre04: c6; Som05: c6]

Se debe asegurar a los ingenieros del software que hay disponibles capacidad y recursos adecuados en forma de personas, pericia, medios, infraestructura y apoyo (sea interna o externamente) para cerciorarse de que el proyecto pueda completarse con éxito de un modo oportuno y rentable (utilizando, por ejemplo, una matriz de requisitos y capacidades). A menudo esto requiere un cálculo del esfuerzo y del coste basado en los métodos adecuados (por ejemplo, técnicas de analogía reguladas por expertos).

### 1.3. Construir para verificar

Dado que los cambios son inevitables, es de vital importancia que desde el inicio se llegue a un acuerdo entre los contratistas acerca de los medios por los que se repasarán y revisarán el alcance y requisitos (por ejemplo, por medio de procedimientos pactados para la gestión de cambios). Esto claramente implica que el alcance y los requisitos no quedarán “grabados en piedra” sino que pueden y deben volverse a revisar en puntos predeterminados según se vaya desarrollando el proyecto (por ejemplo, en las revisiones del diseño, en las revisiones de gestión). Si se aceptan los cambios, deberá usarse entonces algún tipo de análisis de trazabilidad y de análisis de riesgos (ver el apartado 2.5 *Gestión de*

*Riesgos*) para determinar el impacto de esos cambios. También resultará útil un acercamiento que gestione los cambios cuando llegue el momento de repasar los resultados del proyecto, ya que el alcance y los requisitos tendrán que ser la base para evaluar el éxito. [Som05: el c6] Ver también la subárea de control de la configuración del software del KA de Gestión de la Configuración del Software.

## 2. Planificación de un Proyecto de Software

El proceso de planificación iterativa está regulado por el alcance y requisitos, y por el establecimiento de la viabilidad. A estas alturas, se evalúan los procesos del ciclo de vida del software y se selecciona el más apropiado (considerando la naturaleza del proyecto, su grado de novedad, su complejidad funcional y técnica, sus requisitos de calidad, etc).

Si la situación lo aconseja, se planea entonces el propio proyecto en la forma de una descomposición jerárquica de tareas, se especifican y caracterizan los entregables asociados de cada tarea en términos de calidad y de otros atributos en la línea de los requisitos declarados, y se emprende la descripción detallada del esfuerzo de realización, el calendario y la estimación de costes.

Más adelante se asignan los recursos a las tareas para optimizar la productividad del personal (a nivel de individuo, de equipo y organizacional), el uso de equipos y materiales, y la adhesión a los horarios. Se emprende una gestión de riesgos detallada y se discute el “perfil de riesgo” entre todos los contratistas de relieve, llegando a un acuerdo. Se determinan los procesos comprensivos de gestión de la calidad del software como parte del proceso en términos de procedimientos y responsabilidades para asegurar la calidad del software, la verificación y la validación, las revisiones y las auditorías (ver el KA de la Calidad del Software). Ya que es un proceso iterativo, resulta de vital importancia que se declaren y pacten con claridad los procesos y responsabilidades para la gestión, repaso y revisión del plan en ejecución.

### 2.1. Planificación de un Proceso

La selección de un modelo adecuado de ciclo de vida del software (por ejemplo, un prototipado evolutivo en espiral) y la adaptación y el despliegue de ciclos de vida del software, se emprenden a la luz del alcance particular y de los requisitos del proyecto. También se seleccionan métodos y herramientas pertinentes. [Dor02: el v1c6,v2c8; Pfl01: el c2; Pre04: el c2; Rei02: el c1,c3,c5; Som05: el c3; Tha97: el c3] A nivel de proyecto, se utilizan métodos y herramientas adecuados para descomponer el proyecto en tareas, con entradas asociadas, resultados y condiciones de finalización de obra (por ejemplo, una estructura para la descomposición del trabajo). [Dor02: el v2c7; Pfl01: el c3; Pre04: el c21;

1 Rei02: el c4,c5; Som05: el c4; Tha97: el c4,c6] Esto  
2 influye a su vez en las decisiones sobre el horario y  
3 estructura de la organización de alto nivel del  
4 proyecto.

## 5 2.2. *Determinar los Entregables*

6 Se especifica y caracteriza el producto o productos de  
7 cada tarea (por ejemplo, un diseño arquitectónico, un  
8 informe de inspección). [Pfl01: el c3; Pre04: el c24;  
9 Tha97: el c4] Se evalúan las oportunidades de  
10 reutilizar los componentes del software de desarrollos  
11 anteriores o de utilizar productos software del  
12 mercado. Se planifica la utilización de terceras  
13 personas y del software obtenido y se seleccionan los  
14 proveedores.

## 15 2.3. *Esfuerzo, Calendario y Cálculo del Coste*

16 Partiendo de la descomposición de tareas, entradas y  
17 resultados, se determina el rango de esfuerzo  
18 esperado que se requiere para cada tarea, utilizando  
19 un modelo de estimación calibrado basado en datos  
20 históricos sobre el esfuerzo empleado, cuando estén  
21 disponibles y sean pertinentes, u otros métodos como  
22 el juicio de un especialista. Se establecen las  
23 dependencias de las tareas y se identifican los cuellos  
24 de botella potenciales utilizando los métodos  
25 convenientes (por ejemplo, el análisis del camino  
26 crítico). Cuando sea posible se solucionan los cuellos  
27 de botella, y se elabora el esperado cuadro de tareas  
28 con los horarios de inicio, duraciones y horarios de  
29 finalización bien planificados (por ejemplo, el  
30 diagrama PERT). Los requisitos de recursos  
31 (personas, herramientas) se traducen en estimaciones  
32 de costo. [Dor02: el v2c7; Fen98: el c12; Pfl01: el c3;  
33 Pre04: el c23, el c24.; Rei02: el c5,c6; Som05: el  
34 c4,c23; Tha97: el c5] Ésta es una actividad muy  
35 iterativa que debe ser negociada y revisada hasta que  
36 se alcance un acuerdo general entre los contratistas  
37 afectados (principalmente de ingeniería y gestión).

## 38 2.4 *Reparto de Recursos*

39 [Pfl01: c3; Pre04: c24; Rei02: c8,c9; Som05:  
40 c4; Tha97: c6,c7]

41 Los equipos, medios y personas se asocian a las  
42 tareas programadas, incluyendo la asignación de  
43 responsabilidades de cara a su completa realización  
44 (usando, por ejemplo, un diagrama de Gantt). Esta  
45 actividad está regulada y limitada por la  
46 disponibilidad de los recursos y por su uso óptimo  
47 bajo estas circunstancias, así como por temas  
48 relacionados con el personal (por ejemplo,  
49 productividad de los individuos y equipos, dinámicas  
50 de equipo, estructuras organizativas y de equipo).

## 51 2.5 *Gestión de Riesgos*

52 Se lleva a cabo la identificación y análisis de riesgos  
53 (lo que puede salir mal, cómo y por qué, y sus  
54 posibles consecuencias), la valoración crítica de

55 riesgos (cuáles son los riesgos más significativos a  
56 los que se está expuestos, sobre cuáles podemos  
57 hacer algo en cuanto a su influencia), la mitigación de  
58 riesgos y la planificación de contingencias  
59 (formulando una estrategia para controlar los riesgos  
60 y gestionar los perfiles de riesgo). Los métodos de  
61 valoración de riesgos (por ejemplo, los árboles de  
62 decisión y los procesos de simulación) deben  
63 utilizarse para resaltar y evaluar riesgos. A estas  
64 alturas se deben determinar las políticas de abandono  
65 de proyectos en conversaciones con todos los otros  
66 contratistas. [Dor02: el v2c7; Pfl01: el c3; Pre04: el  
67 c25; Rei02: el c11; Som05: el c4; Tha97: el c4] La  
68 gestión de riesgos del proyecto debe influir en  
69 aspectos de riesgo únicos en el software, como la  
70 tendencia de los ingenieros del software a agregar  
71 utilidades no deseadas o como el controlador de  
72 riesgos presente en la naturaleza intangible del  
73 software.

## 74 2.6 *Gestión de Calidad*

75 [Dor02: v1c8,v2c3-c5; Pre04: c26; Rei02: c10;  
76 Som05: c24,c25; Tha97: c9,c10]

77 La calidad se define en términos de atributos  
78 pertinentes al proyecto específico y en los de  
79 cualquier producto o productos asociados a ella,  
80 probablemente tanto en términos cuantitativos como  
81 cualitativos. Estas características de la calidad habrán  
82 sido determinadas en la especificación de requisitos  
83 detallados del software. Ver también el KA de los  
84 Requisitos del Software.

85 Los límites de adhesión a la calidad para cada  
86 indicador se colocan de acuerdo a las expectativas  
87 que tenga el contratista sobre el software en cuestión.  
88 Los procedimientos que hacen referencia a lo largo  
89 del proceso a la SQA en curso a lo largo del proceso  
90 y a la verificación y validación del producto  
91 (entregable) también se especifican en esta fase (por  
92 ejemplo, las revisiones técnicas e inspecciones) (ver  
93 también el KA de la Calidad del Software).

## 94 2.7 *Gestión de Planes*

95 [Som05: c4; Tha97: c4]

96 También se ha de planificar cómo se gestionará el  
97 proyecto y cómo se gestionará la planificación. Los  
98 informes, la supervisión y el control del proyecto  
99 deben encajar en el proyecto de ingeniería del  
100 software seleccionado y en las realidades del  
101 proyecto, y deben reflejarse en los varios artefactos  
102 que se usarán para gestionarlo. Pero, en un contexto  
103 en donde los cambios son más una expectativa que un  
104 susto, es de vital importancia que se gestionen los  
105 propios planes. Esto requiere que sistemáticamente se  
106 dirija, supervise, repase, informe y, donde así lo  
107 requiera, revise, la adhesión a los planes. Los planes  
108 asociados a otros procesos de soporte orientados a  
109 gestión (por ejemplo, documentación, gestión de la  
110 configuración del software, y resolución de

problemas) también necesitan gestionarse de esa misma manera.

### 3. Promulgación del Proyecto de Software

A continuación se ejecutan los planes y se promulgan los procesos incluidos en los planes. A lo largo de este proceso todo se centra en la adhesión a los planes, con una expectativa arrolladora de que tal adhesión llevará a la satisfacción plena de los requisitos del contratista y al logro de los objetivos del proyecto. Las actividades actuales de gestión para medir, supervisar, controlar e informar son fundamentales para la promulgación.

#### 3.1 Implementación de Planes

[Pf101: c3; Som05: c4]

Inicia el proyecto y se emprenden las actividades del proyecto según el horario. En el proceso, se utilizan recursos (por ejemplo, esfuerzo del personal, financiación) y se producen entregables (por ejemplo, documentos de diseño de arquitectura, casos de pruebas).

#### 3.2 Gestión de Contratos con Proveedores

[Som05: c4]

Preparar y ejecutar acuerdos con los proveedores, supervisar la actuación del proveedor, y aceptar sus productos, incorporándolos cuando sean adecuados.

#### 3.3 Implementación de Procesos para Medir

[Fen98: c13,c14; Pre04: c22; Rei02: c10,c12; Tha97: c3,c10]

Se promulga el proceso de medición junto con el proyecto del software, asegurándose de que se recogen datos relevantes y útiles (ver también los apartados 6.2 *Planificar el Proceso de Medición* y 6.3 *Realizar el Proceso de Medición*).

#### 3.4 Proceso de Supervisión

[Dor02: v1c8, v2c2-c5,c7; Rei02: c10; Som05: c25; Tha97: c3,c9]

Se evalúa continuamente y a intervalos predeterminados la adhesión a los diferentes planes. Se analizan los resultados y las condiciones de acabado de cada tarea. Se evalúan los entregables en términos de las características que ellos requieren (por ejemplo, por medio de revisiones y auditorías). Se investiga el consumo de fuerzas, la adhesión a horarios, y los costes a día de hoy, y se examina el uso de recursos. Se revisa de nuevo el perfil de riesgo del proyecto y se evalúa la adhesión a los requisitos de calidad.

Se modelan y analizan los datos de medición. Se emprende el análisis de variación basado en la desviación actual de los resultados y valores

esperados. Esto puede darse en forma de desbordamiento de costes, equivocaciones en el horario y similares. Se lleva a cabo la identificación de la desviación y el análisis de calidad y otros datos de medición (por ejemplo, el análisis de la densidad de los defectos). Se recalculan la exposición a riesgos y sus influencias y se ejecutan de nuevo los árboles de decisiones, las simulaciones, etc, a la luz de los nuevos datos. Estas actividades permiten la detección de problemas y la identificación de excepciones basada en la superación de los límites existentes. Se informa de los resultados según se vaya necesitando y sobre todo cuando se hayan superado los límites aceptables.

#### 3.5 Proceso de Control

[Dor02: v2c7; Rei02: c10; Tha97: c3,c9]

Los resultados de las actividades de supervisión del proceso proporcionan la base sobre la que se toman las decisiones para actuar. Se pueden hacer cambios al proyecto cuando se juzgue oportuno y cuando se modele y gestione el impacto y los riesgos asociados a éstos. Esto puede tomar la forma de una acción correctiva (por ejemplo, volviendo a probar ciertos componentes), puede que involucre la incorporación de contingencias para evitar sucesos semejantes (por ejemplo, la decisión de utilizar prototipados para ayudar en la validación de los requisitos del software), y/o puede implicar la revisión de los distintos planes y de otros documentos del proyecto (por ejemplo, la especificación de requisitos) para corregir los resultados inesperados y sus implicaciones.

En algunos casos, puede llevar al abandono del proyecto. En todos los casos, se adhiere al control de cambios y a los procedimientos de gestión de configuración del software (ver también el KA de la Gestión de Configuración del Software) se documentan y comunican decisiones a todos los implicados importantes, se repasan los planes y si es necesario se revisan, y todos los datos importantes se graban en la base de datos central (ver también el apartado 6.3 *Llevar a cabo el Proceso de Revisión*).

#### 3.6 Informes

[Rei02: c10; Tha97: c3,c10]

En períodos específicos y concertados, se informa de la adhesión a los planes dentro de la organización (por ejemplo al comité de dirección de cartera del proyecto) y a los contratistas externos (por ejemplo, clientes, usuarios). Informes de esta naturaleza deben orientarse hacia una adhesión global en oposición a los informes detallados que se requieren frecuentemente dentro del equipo de proyecto.

## 4. Revisión y Evaluación

En puntos críticos del proyecto, se evalúan el progreso global hacia el logro de los objetivos

1 prefijados y la satisfacción de los requisitos del  
2 contratista. De igual modo, en hitos particulares se  
3 llevan a cabo valoraciones sobre la efectividad del  
4 proceso global hasta la fecha, del personal  
5 involucrado, y de las herramientas y métodos  
6 utilizados.

#### 7 4.1 Determinar la Satisfacción de los Requisitos

8 [Rei02: c10; Tha97: c3,c10]

9 Ya que uno de nuestros objetivos principales consiste  
10 en lograr la satisfacción del contratista (usuario o  
11 cliente), es importante que el progreso hacia este  
12 objetivo sea evaluado formal y periódicamente. Esto  
13 ocurre al lograr los principales hitos del proyecto (por  
14 ejemplo, la confirmación de la arquitectura del diseño  
15 de software, la revisión técnica de la integración del  
16 software). Se identifican variaciones a las  
17 expectativas y se llevan a cabo acciones adecuadas.  
18 Al igual que en la actividad de control del proceso  
19 arriba indicada (ver el apartado 3.5 *Proceso de*  
20 *Control*), en todos los casos, se adhiere al control de  
21 cambios y a los procedimientos de gestión de  
22 configuración del software (ver también el KA de la  
23 Gestión de Configuración del Software) se  
24 documentan y comunican decisiones a todos los  
25 implicados importantes, se repasan los planes y si es  
26 necesario se revisan, y todos los datos importantes se  
27 graban en la base de datos central (ver también el  
28 apartado 6.3 *Llevar a cabo el Proceso de Revisión*).  
29 Se puede encontrar más información en el KA de las  
30 Pruebas del Software, en el apartado 2.2 *Objetivos de*  
31 *las Pruebas* y en el KA de la Calidad del Software,  
32 en el apartado 2.3 *Revisiones y Auditorías*.

#### 33 4.2 Revisar y Evaluar la Ejecución

34 [Dor02: v1c8,v2c3,c5; Pfl01: c8,c9; Rei02: c10;  
35 Tha97: c3,c10]

36 Las revisiones periódicas de lo realizado, dirigidas al  
37 personal del proyecto, proporcionan detalles sobre la  
38 probabilidad de ser fiel a los planes así como sobre  
39 las posibles áreas de dificultad (por ejemplo,  
40 conflictos entre miembros del equipo). Se evalúan los  
41 distintos métodos, herramientas y técnicas empleadas  
42 para ver su eficacia y adecuación, y se valora  
43 sistemática y periódicamente el propio proceso para  
44 conocer su relevancia, utilidad y eficacia en el  
45 contexto del proyecto. Cuando se juzga necesario, se  
46 llevan a cabo y se gestionan los cambios.

### 47 5. Cierre

48 El proyecto llega a su fin cuando todos los planes y  
49 procesos implicados se han promulgado y  
50 completado. En esta fase, se repasan los criterios para  
51 el éxito del proyecto. Una vez que se ha establecido  
52 el cierre, se llevan a cabo actividades de archivado,  
53 post mortem y de mejoras de los procesos.

#### 54 5.1 Determinar el Cierre

55 [Dor02: v1c8,v2c3,c5; Rei02: c10; Tha97:  
56 c3,c10]

57 Se han completado las tareas tal y como se  
58 especificaron en los planes, y se confirman los  
59 criterios para lograr un acabado satisfactorio. Todos  
60 los productos planificados han sido entregados con  
61 características aceptables.

62 Se marca y confirma la satisfacción de los requisitos,  
63 se han logrado los objetivos del proyecto. Estos  
64 procesos por lo general involucran a todos los  
65 contratistas y acaban con la documentación tanto de  
66 la aceptación del cliente y como de los informes de  
67 cualquier otro problema pendiente conocido.

#### 68 5.2 Actividades de Cierre

69 [Pfl01: c12; Som05: c4]

70 Tras haberse confirmado el cierre, se archivan los  
71 materiales del proyecto de acuerdo a los métodos,  
72 localización y duración pactados con los contratistas.  
73 La base de datos de medición de la organización se  
74 pone al día con los datos finales del proyecto y se  
75 emprenden análisis post-proyecto. Se inicia un  
76 proyecto post mortem con el fin de analizar los  
77 temas, problemas y oportunidades encontrados  
78 durante el proceso (particularmente por medio de  
79 revisiones y evaluaciones, ver el subárea 4 *Revisión y*  
80 *Evaluación*) y se sacan lecciones del proceso que  
81 luego alimentan los conocimientos de la organización  
82 y los intentos de mejora (ver también el KA del  
83 Proceso de Ingeniería del Software).

### 84 6. Medidas de la Ingeniería del Software

85 [ISO 15939-02]

86 La importancia de la medición y su papel en las  
87 buenas prácticas de gestión está ampliamente  
88 reconocido, y es tal que su importancia sólo puede  
89 aumentar en los próximos años. Medir con eficacia se  
90 ha convertido en una de las piedras angulares de la  
91 madurez de una organización. Las palabras claves  
92 para la medición del software y para los métodos de  
93 medición fueron definidas en [ISO15939-02] basadas  
94 en el vocabulario internacional de metrología ISO  
95 [ISO93]. No obstante, los lectores encontrarán en la  
96 literatura existente diferencias en la terminología; por  
97 ejemplo, el término "métrica" a veces se usa en lugar  
98 de "medición".

99 Este apartado sigue el estándar internacional ISO/IEC  
100 15939, que describe el proceso que define las  
101 actividades y tareas necesarias para implementar un  
102 proceso de medición e incluye, asimismo, un modelo  
103 de medición de la información.

#### 104 6.1 Establecer y Sostener el Compromiso de Medir

105 ♦ Aceptar los requisitos de medición. Cada  
106 tentativa de medición debe estar guiada por  
107 objetivos organizacionales, e impulsada por un

1	conjunto de requisitos de medición establecidos	58	comunicados y revisados por los contratistas
2	por la organización y por el proyecto. Por	59	[ISO 15939-02: 5.2.2].
3	ejemplo, un objetivo organizacional podría ser	60	♦ Seleccionar las mediciones. Se deben elegir las
4	"ser los primeros en salir al mercado con los	61	mediciones candidatas al puesto, claramente
5	nuevos productos." [Fen98: c3,c13; Pre04: c22]	62	vinculadas a las necesidades de información. Las
6	Esto a su vez podría generar un requisito para	63	mediciones deben seleccionarse en base a las
7	que se midan los factores que contribuyen a este	64	prioridades de las necesidades de información y
8	objetivo, y así se puedan gestionar los proyectos	65	otros criterios como el coste de recolección de
9	para hacer frente a este objetivo.	66	datos, el grado de trastorno del proceso durante
10	- Definir el alcance de la medición. Se debe	67	la recolección, la facilidad de análisis, la
11	establecer la unidad organizacional a la que	68	facilidad de obtener datos precisos y
12	se va a aplicar cada requisito de medición.	69	consistentes, etc [ISO15939-02: 5.2.3 y
13	Esto puede consistir en un área funcional, en	70	Apéndice C].
14	un solo proyecto, en un solo sitio, o incluso	71	♦ Definir la recolección de datos, el análisis y los
15	en toda la empresa. Todas las subsiguientes	72	procedimientos para informar. Esto abarca los
16	tareas de medición relacionadas con este	73	procedimientos y horarios de recolección, y la
17	requisito deben encontrarse dentro del	74	gestión de datos de almacenamiento,
18	alcance definido. Además, se debe	75	verificación, análisis, informes y configuración
19	identificar a los contratistas implicados.	76	[ISO15939-02: 5.2.4].
20	- Compromiso de la dirección y del personal	77	♦ Definir los criterios para evaluar los productos de
21	con la medición. El compromiso debe	78	información. Los criterios para evaluar están
22	establecerse formalmente, debe	79	influenciados por los objetivos técnicos y
23	comunicarse, y debe apoyarse en los	80	comerciales de la unidad organizacional. Los
24	recursos (ver el siguiente artículo).	81	productos de información incluyen los asociados
25	♦ Empeñar recursos para la medición. El	82	con el producto que está siendo elaborado, así
26	compromiso de la organización con la medición	83	como los asociados con los procesos utilizados
27	es un factor esencial para el éxito, como	84	para gestionar y medir el proyecto [ISO15939-
28	demuestra la asignación de recursos para llevar a	85	02: 5.2.5 y Apéndices D y E].
29	cabo el proceso de medición. El asignar recursos	86	♦ Revisar, aprobar y proporcionar recursos para las
30	incluye el reparto de responsabilidades para las	87	tareas de medición.
31	diferentes tareas del proceso de medición (tales	88	- El plan de medición debe ser revisado y
32	como usuario, analista y bibliotecario) y el	89	aprobado por los contratistas adecuados.
33	proporcionar una financiación, entrenamiento,	90	Esto incluye todos los procedimientos de
34	herramientas, y apoyo adecuados para dirigir el	91	recolección de datos y los procedimientos de
35	proceso de un modo perdurable.	92	almacenamiento, análisis e informes; los
36	6.2 Planificar el Proceso de Medición	93	criterios de evaluación; horarios y
37	♦ Caracterizar la unidad organizacional. La unidad	94	responsabilidades. Los criterios para revisar
38	organizacional proporciona el contexto para la	95	estos artefactos tendrían que haberse
39	medición así que es importante hacer explícito	96	establecido a un nivel de unidad
40	este contexto y articular las presunciones que	97	organizacional o superior y debieran usarse
41	éste incluye y las restricciones que va	98	como base para estas revisiones. Tales
42	imponiendo. La caracterización puede darse en	99	criterios deben tomar en cuenta experiencias
43	términos de procesos organizacionales, dominios	100	anteriores, disponibilidad de recursos, y
44	de aplicaciones, tecnología e interfaces	101	trastornos potenciales del proceso cuando se
45	organizacionales. Un modelo de proceso	102	proponen cambios a las prácticas actuales.
46	organizacional también es por lo general un	103	La aprobación demuestra que existe un
47	elemento de una caracterización de la unidad	104	compromiso con el proceso de medición
48	organizacional [ISO15939-02: 5.2.1].	105	[ISO15939-02: 5.2.6.1 y Apéndice F].
49	♦ Identificar las necesidades de información. Las	106	- Hay que hacer que los recursos estén
50	necesidades de información se basan en las	107	disponibles para implementar las tareas de
51	metas, restricciones, riesgos y problemas de la	108	medición planeadas y aprobadas. La
52	unidad organizacional. Éstas pueden proceder de	109	disponibilidad de los recursos puede
53	objetivos comerciales, organizacionales,	110	organizarse en aquellos casos en donde los
54	reguladores y/o del producto. Deben ser	111	cambios han de pilotarse antes de un amplio
55	identificadas y deben priorizarse. Debe entonces	112	despliegue. Se debe prestar atención a los
56	seleccionarse un subconjunto para ser cotejado y	113	recursos necesarios para un amplio
57	los resultados deben ser documentados,		



despliegue de los nuevos procedimientos o mediciones [ISO15939-02: 5.2.6.2].

- ♦ Adquirir y desplegar tecnologías de apoyo. Esto incluye una evaluación de las tecnologías de apoyo disponibles, la selección de las tecnologías más adecuadas, la adquisición de esas tecnologías, y el despliegue de esas tecnologías [ISO 15939-02:5.2.7].

6.3 Realizar el Proceso de Medición

- ♦ Integrar los procedimientos de medición con los procesos pertinentes. Los procedimientos de medición, tales como la recolección de datos, deben integrarse en los procesos que están midiendo. Esto puede que implique cambiar los procesos actuales para adaptar la recolección de datos o las actividades de generación. Puede también implicar el análisis de los actuales procesos para minimizar esfuerzos adicionales y evaluaciones del efecto en los empleados, con el fin de asegurarse de que serán aceptados los procedimientos de medición. Se necesita considerar los temas morales y otros factores humanos. Además, los procedimientos de medición deben comunicarse a los proveedores de datos, puede que se tenga que proporcionar entrenamiento, y se debe proporcionar el típico apoyo. De manera similar, el análisis de datos y los procedimientos de información deben tener la típica integración en los procesos organizacionales y/o del proyecto [ISO 15939-02: 5.3.1].
- ♦ Recolectar datos. Se debe recolectar, verificar y almacenar datos [ISO 1539-02: 5.3.2].
- ♦ Analizar datos y desarrollar productos de información. Se pueden agregar, transformar o recodificar datos como parte del proceso de análisis, utilizando un grado de rigor adecuado a la naturaleza de los datos y las necesidades de información. Los resultados de este análisis son indicadores típicos, tales como gráficas, números u otras indicaciones que han de ser interpretadas, dando lugar a conclusiones iniciales que han de presentar los contratistas. Los resultados y conclusiones han de consultarse, utilizando un

proceso definido por la organización (que puede ser formal o informal). Los proveedores de datos y los usuarios de las mediciones deben participar en revisar los datos para asegurar que son significativos y precisos, y que puede llevar a acciones razonables [ISO 15939-02: 5.3.3 y Apéndice G].

- ♦ Comunicar los resultados. Los productos de información deben documentarse y comunicarse a los usuarios y contratistas [ISO 15939-02: 5.3.4].

6.4 Evaluar las Mediciones

- ♦ Evaluar los productos de información. Evaluar los productos de información contrastándolos con los criterios de evaluación específicos y determinar las fuerzas y debilidades de los productos de información. Esto puede realizarlo un proceso interno o una auditoría externa y debe incluir una retroalimentación de los usuarios de las mediciones. Grabar las lecciones aprendidas en una base de datos adecuada [ISO 15939-02: 5.4.1 y Apéndice D].
- ♦ Evaluar el proceso de medición. Evaluar el proceso de medición contrastándolo con los criterios de evaluación específicos y determinar las fuerzas y debilidades de los procesos. Esto puede realizarlo un proceso interno o una auditoría externa y debe incluir una retroalimentación de los usuarios de las mediciones. Grabar las lecciones aprendidas en una base de datos adecuada [ISO 15939-02: 5.4.1 y Apéndice D].
- ♦ Identificar las mejoras potenciales. Tales mejoras pueden consistir en cambios en el formato de los indicadores, cambios en las unidades medidas, o en la reclasificación de las categorías. Determinar los costes y beneficios de las mejoras potenciales y seleccionar las acciones de mejora adecuadas. Comunicar las mejoras propuestas al dueño del proceso de medición y a los contratistas para su revisión y aprobación. Comunicar también la falta de mejoras potenciales si el análisis no identifica ninguna mejora [ISO 15939-02: 5.4.2].

# MATRIZ DE TEMAS VS. MATERIAL DE REFERENCIA

	[Dor02]	[ISO15239-02]	[Fen98]	[Pfi01]	[Pre04]	[Rei02]	[Som05]	[Tha97]
<b>1. Iniciación y Alcance</b>								
1.1 Determinación y Negociación de Requisitos	v2c4			c4	c7		c5	
1.2 Viabilidad, Análisis					c6		c6	
1.3 Construir para Verificar							c6	
<b>2. Planificación de un Proyecto Software</b>								
2.1 Planificación de un Proceso	v1c6, v2c7, v2c8			c2, c3	c2, c21	c1, c3, c5	c3, c4	c3, c4, c6
2.2 Determinar los entregables				c3	c24			c4
2.3 Esfuerzo, Horario y Cálculo del Coste	v2c7		c12	c3	c23, c24	c5, c6	c4, c23	c5
2.4 Reparto de Recursos				c3	c24	c8, c9	c4	c6, c7
2.5 Gestión de Riesgos	v2c7			c3	c25	c11	c4	c4
2.6 Gestión de Calidad	v1c8, v2c3-c5				c26	c10	c24, c25	c9, c10
2.7 Gestión de Planes							c4	c4
<b>3. Promulgación del Proyecto Software</b>								
3.1 Implementación de Planes				c3			c4	
3.2 Gestión de Contratos con Proveedores							c4	
3.3 Implementación de Procesos para medir			c13, c14		c22	c10, c12		c3, c10
3.4 Proceso de Supervisión	v1c8, v2c2-c5, c7					c10	c25	c3, c10
3.5 Proceso de Control	v2c7					c10		c3, c9
3.6 Informes						c10		c3, c10
<b>4. Revisión y Evaluación</b>								
4.1 Determinar la satisfacción de los Requisitos						c10		c3, c10
4.2 Revisar y Evaluar la Ejecución	v1c8, v2c3, c5			c8, c9		c10		c3, c10
<b>5. Cierre</b>								
5.1 Determinar el Cierre	v1c8, v2c3, c5					c10		c3, c10
5.2 Actividades del Cierre				c12			c4	
<b>6. Medida de la Ingeniería del Software</b>		*						
6.1 Establecer y Sostener el compromiso de Medir			c3, c13		c22			
6.2 Planificar el Proceso de Medición		c5, C,D,E,F						
6.3 Realizar el Proceso de Medición		c5, G						
6.4 Evaluar las Mediciones		c5, D						

## 2 REFERENCIAS RECOMENDADAS PARA LA GESTIÓN 3 DEL SOFTWARE

- 4  
5 [Dor02] M. Dorfman and R.H. Thayer, eds., *Software*  
6 *Engineering*, IEEE Computer Society Press, 2002,  
7 Vol. 1, Chap. 6, 8, Vol. 2, Chap. 3, 4, 5, 7, 8.  
8  
9 [Fen98] N.E. Fenton and S.L. Pfleeger, *Software*  
10 *Metrics: A Rigorous & Practical Approach*, second  
11 ed., International Thomson Computer Press, 1998,  
12 Chap. 1-14.  
13  
14 [ISO15939-02] ISO/IEC 15939:2002, *Software*  
15 *Engineering — Software Measurement Process*, ISO  
16 and IEC, 2002.  
17 [Pfl01] S.L. Pfleeger, *Software Engineering: Theory*  
18 *and Practice*, second ed., Prentice Hall, 2001, Chap.  
19 2-4, 8, 9, 12, 13.

- 21 [Pre04] R.S. Pressman, *Software Engineering: A*  
22 *Practitioner's Approach*, sixth ed., McGraw-Hill,  
23 2004, Chap. 2, 6, 7, 22-26.  
24  
25 [Rei02] D.J. Reifer, ed., *Software Management*, IEEE  
26 Computer Society Press, 2002, Chap. 1-6, 7-12, 13.  
27  
28 [Som05] I. Sommerville, *Software Engineering*,  
29 seventh ed., Addison-Wesley, 2005, Chap. 3-6, 23-  
30 25.  
31  
32 [Tha97] R.H. Thayer, ed., *Software Engineering*  
33 *Project Management*, IEEE Computer Society Press,  
34 1997, Chap. 1-10.

**APÉNDICE A. LISTA DE LECTURAS  
ADICIONALES**

- (Adl99) T.R. Adler, J.G. Leonard, and R.K. Nordgren, "Improving Risk Management: Moving from Risk Elimination to Risk Avoidance," *Information and Software Technology*, vol. 41, 1999, pp. 29-34.
- (Bai98) R. Baines, "Across Disciplines: Risk, Design, Method, Process, and Tools," *IEEE Software*, July/August 1998, pp. 61-64.
- (Bin97) R.V. Binder, "Can a Manufacturing Quality Model Work for Software?" *IEEE Software*, September/October 1997, pp. 101-102,105.
- (Boe97) B.W. Boehm and T. DeMarco, "Software Risk Management," *IEEE Software*, May/June 1997, pp. 17-19.
- (Bri96) L.C. Briand, S. Morasca, and V.R. Basili, "Property-Based Software Engineering Measurement," *IEEE Transactions on Software Engineering*, vol. 22, iss. 1, 1996, pp. 68-86.
- (Bri96a) L. Briand, K.E. Emam, and S. Morasca, "On the Application of Measurement Theory in Software Engineering," *Empirical Software Engineering*, vol. 1, 1996, pp. 61-88.
- (Bri97) L.C. Briand, S. Morasca, and V.R. Basili, "Response to: Comments on 'Property-based Software Engineering Measurement: Refining the Additivity Properties,'" *IEEE Transactions on Software Engineering*, vol. 23, iss. 3, 1997, pp. 196-197.
- (Bro87) F.P.J. Brooks, "No Silver Bullet: Essence and Accidents of Software Engineering," *Computer*, Apr. 1987, pp. 10-19.
- (Cap96) J. Capers, *Applied Software Measurement: Assuring Productivity and Quality*, second ed., McGraw-Hill, 1996.
- (Car97) M.J. Carr, "Risk Management May Not Be For Everyone," *IEEE Software*, May/June 1997, pp. 21-24.
- (Cha96) R.N. Charette, "Large-Scale Project Management Is Risk Management," *IEEE Software*, July 1996, pp. 110-117.
- (Cha97) R.N. Charette, K.M. Adams, and M.B. White, "Managing Risk in Software Maintenance," *IEEE Software*, May/June 1997, pp. 43-50.

- (Col96) B. Collier, T. DeMarco, and P. Fearey, "A Defined Process for Project Postmortem Review," *IEEE Software*, July 1996, pp. 65-72.
- (Con97) E.H. Conrow and P.S. Shishido, "Implementing Risk Management on Software Intensive Projects," *IEEE Software*, May/June 1997, pp. 83-89.
- (Dav98) A.M. Davis, "Predictions and Farewells," *IEEE Software*, July/August 1998, pp. 6-9.
- (Dem87) T. DeMarco and T. Lister, *Peopleware: Productive Projects and Teams*, Dorset House Publishing, 1987.
- (Dem96) T. DeMarco and A. Miller, "Managing Large Software Projects," *IEEE Software*, July 1996, pp. 24-27.
- (Fav98) J. Favaro and S.L. Pfleeger, "Making Software Development Investment Decisions," *ACM SIGSoft Software Engineering Notes*, vol. 23, iss. 5, 1998, pp. 69-74.
- (Fay96) M.E. Fayad and M. Cline, "Managing Object-Oriented Software Development," *Computer*, September 1996, pp. 26-31.
- (Fen98) N.E. Fenton and S.L. Pfleeger, *Software Metrics: A Rigorous & Practical Approach*, second ed., International Thomson Computer Press, 1998.
- (Fle99) R. Fleming, "A Fresh Perspective on Old Problems," *IEEE Software*, January/February 1999, pp. 106-113.
- (Fug98) A. Fuggetta et al., "Applying GQM in an Industrial Software Factory," *ACM Transactions on Software Engineering and Methodology*, vol. 7, iss. 4, 1998, pp. 411-448.
- (Gar97) P.R. Garvey, D.J. Phair, and J.A. Wilson, "An Information Architecture for Risk Assessment and Management," *IEEE Software*, May/June 1997, pp. 25-34.
- (Gem97) A. Gemmer, "Risk Management: Moving beyond Process," *Computer*, May 1997, pp. 33-43.
- (Gla97) R.L. Glass, "The Ups and Downs of Programmer Stress," *Communications of the ACM*, vol. 40, iss. 4, 1997, pp. 17-19.
- (Gla98) R.L. Glass, "Short-Term and Long-Term Remedies for Runaway Projects," *Communications of the ACM*, vol. 41, iss. 7, 1998, pp. 13-15.
- (Gla98a) R.L. Glass, "How Not to Prepare for a Consulting Assignment, and Other Ugly Consultancy

- 1 Truths," *Communications of the ACM*, vol. 41, iss.
- 2 12, 1998, pp. 11-13.
- 3
- 4 (Gla99) R.L. Glass, "The Realities of Software
- 5 Technology Payoffs," *Communications of the ACM*,
- 6 vol. 42, iss. 2, 1999, pp. 74-79.
- 7 (Gra99) R. Grable et al., "Metrics for Small Projects:
- 8 Experiences at the SED," *IEEE Software*,
- 9 March/April 1999, pp. 21-29.
- 10
- 11 (Gra87) R.B. Grady and D.L. Caswell, *Software*
- 12 *Metrics: Establishing A Company-Wide Program*.
- 13 Prentice Hall, 1987.
- 14
- 15 (Hal97) T. Hall and N. Fenton, "Implementing
- 16 Effective Software Metrics Programs," *IEEE*
- 17 *Software*, March/April 1997, pp. 55-64.
- 18
- 19 (Hen99) S.M. Henry and K.T. Stevens, "Using
- 20 Belbin's Leadership Role to Improve Team
- 21 Effectiveness: An Empirical Investigation," *Journal*
- 22 *of Systems and Software*, vol. 44, 1999, pp. 241-250.
- 23
- 24 (Hoh99) L. Hohmann, "Coaching the Rookie
- 25 Manager," *IEEE Software*, January/February 1999,
- 26 pp. 16-19.
- 27
- 28 (Hsi96) P. Hsia, "Making Software Development
- 29 Visible," *IEEE Software*, March 1996, pp. 23-26.
- 30
- 31 (Hum97) W.S. Humphrey, *Managing Technical*
- 32 *People: Innovation, Teamwork, and the Software*
- 33 *Process*: Addison-Wesley, 1997.
- 34
- 35 (IEEE12207.0-96) IEEE/EIA 12207.0-
- 36 1996/ISO/IEC12207:1995, *Industry Implementation*
- 37 *of Int. Std. ISO/IEC 12207:95, Standard for*
- 38 *Information Technology-Software Life Cycle*
- 39 *Processes*, IEEE,
- 40 1996.
- 41
- 42 (Jac98) M. Jackman, "Homeopathic Remedies for
- 43 Team Toxicity," *IEEE Software*, July/August 1998,
- 44 pp. 43-45.
- 45
- 46 (Kan97) K. Kansala, "Integrating Risk Assessment
- 47 with Cost 8-12 © IEEE - 2004 Version Estimation,"
- 48 *IEEE Software*, May/June 1997, pp. 61-67.
- 49
- 50 (Kar97) J. Karlsson and K. Ryan, "A Cost-Value
- 51 Approach for Prioritizing Requirements," *IEEE*
- 52 *Software*, September/October 1997, pp. 87-94.
- 53
- 54 (Kar96) D.W. Karolak, *Software Engineering Risk*
- 55 *Management*, IEEE Computer Society Press, 1996.
- 56
- 57 (Kau99) K. Kautz, "Making Sense of Measurement
- 58 for Small Organizations," *IEEE Software*,
- 59 March/April 1999, pp. 14-20.
- 60
- 61 (Kei98) M. Keil et al., "A Framework for Identifying
- 62 Software Project Risks," *Communications of the*
- 63 *ACM*, vol. 41, iss. 11, 1998, pp. 76-83.
- 64
- 65 (Ker99) B. Kernighan and R. Pike, "Finding
- 66 Performance Improvements," *IEEE Software*,
- 67 March/April 1999, pp. 61-65.
- 68 (Kit97) B. Kitchenham and S. Linkman, "Estimates,
- 69 Uncertainty, and Risk," *IEEE Software*, May/June
- 70 1997, pp. 69-74.
- 71
- 72 (Lat98) F. v. Latum et al., "Adopting GQM-Based
- 73 Measurement in an Industrial Environment," *IEEE*
- 74 *Software*, January-February 1998, pp. 78-86.
- 75
- 76 (Leu96) H.K.N. Leung, "A Risk Index for Software
- 77 Producers," *Software Maintenance: Research and*
- 78 *Practice*, vol. 8, 1996, pp. 281-294.
- 79
- 80 (Lis97) T. Lister, "Risk Management Is Project
- 81 Management for Adults," *IEEE Software*, May/June
- 82 1997, pp. 20-22.
- 83
- 84 (Mac96) K. Mackey, "Why Bad Things Happen to
- 85 Good Projects," *IEEE Software*, May 1996, pp. 27-
- 86 32.
- 87
- 88 (Mac98) K. Mackey, "Beyond Dilbert: Creating
- 89 Cultures that Work," *IEEE Software*,
- 90 January/February 1998, pp. 48-49.
- 91
- 92 (Mad97) R.J. Madachy, "Heuristic Risk Assessment
- 93 Using Cost Factors," *IEEE Software*, May/June 1997,
- 94 pp. 51-59.
- 95
- 96 (McC96) S.C. McConnell, *Rapid Development:*
- 97 *Taming Wild Software Schedules*, Microsoft Press,
- 98 1996.
- 99
- 100 (McC97) S.C. McConnell, *Software Project Survival*
- 101 *Guide*, Microsoft Press, 1997.
- 102
- 103 (McC99) S.C. McConnell, "Software Engineering
- 104 Principles," *IEEE Software*, March/April 1999, pp. 6-
- 105 8.
- 106
- 107 (Moy97) T. Moynihan, "How Experienced Project
- 108 Managers Assess Risk," *IEEE Software*, May/June
- 109 1997, pp. 35-41.
- 110
- 111 (Ncs98) P. Ncsi, "Managing OO Projects Better,"
- 112 *IEEE Software*, July/August 1998, pp. 50-60.
- 113
- 114 (Nol99) A.J. Nolan, "Learning From Success," *IEEE*
- 115 *Software*, January/February 1999, pp. 97-105.
- 116
- 117 (Off97) R.J. Offen and R. Jeffery, "Establishing
- 118 Software Measurement Programs," *IEEE Software*,
- 119 March/April 1997, pp. 45-53.
- 120

- (Par96) K.V.C. Parris, "Implementing Accountability," *IEEE Software*, July/August 1996, pp. 83-93.
- (Pfl97) S.L. Pfleeger, "Assessing Measurement (Guest Editor's Introduction)," *IEEE Software*, March/April 1997, pp. 25-26.
- (Pfl97a) S.L. Pfleeger et al., "Status Report on Software Measurement," *IEEE Software*, March/April 1997, pp. 33-43.
- (Put97) L.H. Putman and W. Myers, *Industrial Strength Software — Effective Management Using Measurement*, IEEE Computer Society Press, 1997.
- (Rob99) P.N. Robillard, "The Role of Knowledge in Software Development," *Communications of the ACM*, vol. 42, iss. 1, 1999, pp. 87-92.
- (Rod97) A.G. Rodrigues and T.M. Williams, "System Dynamics in Software Project Management: Towards the Development of a Formal Integrated Framework," *European Journal of Information Systems*, vol. 6, 1997, pp. 51-66.
- (Rop97) J. Ropponen and K. Lyytinen, "Can Software Risk Management Improve System Development: An Exploratory Study," *European Journal of Information Systems*, vol. 6, 1997, pp. 41-50.
- (Sch99) C. Schmidt et al., "Disincentives for Communicating Risk: A Risk Paradox," *Information and Software Technology*, vol. 41, 1999, pp. 403-411.
- (Sco92) R.L. v. Scoy, "Software Development Risk: Opportunity, Not Problem," *Software Engineering Institute*, Carnegie Mellon University CMU/SEI-92-TR-30, 1992.
- (Sla98) S.A. Slaughter, D.E. Harter, and M.S. Krishnan, "Evaluating the Cost of Software Quality," *Communications of the ACM*, vol. 41, iss. 8, 1998, pp. 67-73.
- (Sol98) R. v. Solingen, R. Berghout, and F. v. Latum, "Interrupts: Just a Minute Never Is," *IEEE Software*, September/October 1998, pp. 97-103.
- (Whi95) N. Whitten, *Managing Software Development Projects: Formulas for Success*, Wiley, 1995.
- (Wil99) B. Wiley, *Essential System Requirements: A Practical Guide to Event-Driven Methods*, Addison-Wesley, 1999.
- (Zel98) M.V. Zelkowitz and D.R. Wallace, "Experimental Models for Validating Technology," *Computer*, vol. 31, iss. 5, 1998, pp. 23-31.

**APÉNDICE B. LISTA DE ESTÁNDARES**

(IEEE610.12-90) IEEE Std 610.12-1990 (R2002),  
IEEE Standard Glossary of Software Engineering  
Terminology, IEEE, 1990.  
(IEEE12207.0-96) IEEE/EIA 12207.0-  
1996//ISO/IEC12207:1995, Industry Implementation  
of Int. Std. ISO/IEC 12207:95, Standard for  
Information Technology-Software Life Cycle  
Processes, IEEE, 1996.

(ISO15939-02) ISO/IEC 15939:2002, Software  
Engineering-Software Measurement Process, ISO  
and IEC, 2002.  
(PMI00) Project Management Institute Standards  
Committee, A Guide to the Project Management  
Body of Knowledge (PMBOK), Project Management  
Institute, 2000.