



Universidad de San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

**USO DEL PROCESO UNIFICADO PARA ALCANZAR
EL NIVEL 2 DE MADUREZ AL DESARROLLAR SOFTWARE**

Nestor Gildardo Ordóñez Salguero

Asesorado por el Ing. José Ricardo Morales Prado

Guatemala, febrero de 2012

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**USO DEL PROCESO UNIFICADO PARA ALCANZAR
EL NIVEL 2 DE MADUREZ AL DESARROLLAR *SOFTWARE***

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA
FACULTAD DE INGENIERÍA

POR

NESTOR GILDARDO ORDÓÑEZ SALGUERO

ASESORADO POR EL ING. JOSÉ RICARDO MORALES PRADO

AL CONFERÍRSELE EL TÍTULO DE

INGENIERO EN CIENCIAS Y SISTEMAS

GUATEMALA, FEBRERO DE 2012

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
FACULTAD DE INGENIERÍA



NÓMINA DE JUNTA DIRECTIVA

DECANO	Ing. Murphy Olympo Paiz Recinos
VOCAL I	Ing. Alfredo Enrique Beber Aceituno
VOCAL II	Ing. Pedro Antonio Aguilar Polanco
VOCAL III	Ing. Miguel Ángel Dávila Calderón
VOCAL IV	Br. Juan Carlos Molina Jiménez
VOCAL V	Br. Mario Maldonado Muralles
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO

DECANO	Ing. Sydney Alexander Samuels Milson
EXAMINADORA	Inga. Elizabeth Domínguez Alvarado
EXAMINADORA	Inga. Virginia Victoria Tala Ayerdi
EXAMINADOR	Ing. José Ricardo Morales Prado
SECRETARIO	Ing. Carlos Humberto Pérez Rodríguez

HONORABLE TRIBUNAL EXAMINADOR

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

USO DEL PROCESO UNIFICADO PARA ALCANZAR EL NIVEL 2 DE MADUREZ AL DESARROLLAR SOFTWARE

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería en Ciencias y Sistemas, con fecha febrero de 2011.


Nestor Gildardo Ordóñez Salguero

Guatemala, 21 de octubre de 2011


Ingeniero
Carlos Alfredo Azurdia Morales
Coordinador Comisión de Trabajos de Graduación
Dirección de la Escuela de Ciencias y Sistemas
Facultad de Ingeniería
Universidad de San Carlos de Guatemala

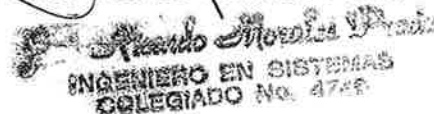
Respetable Ingeniero Azurdia:

Por medio de la presente hago de su conocimiento que he tenido a bien revisar el trabajo de graduación de **NESTOR GILDARDO ORDÓÑEZ SALGUERO** con carné universitario número **9617125**, titulado **"USO DEL PROCESO UNIFICADO PARA ALCANZAR EL NIVEL 2 DE MADUREZ AL DESARROLLAR SOFTWARE"**, por lo cual me permito recomendar dicho trabajo para la respectiva revisión por parte de la comisión de trabajos de graduación de la Escuela de Ciencias y Sistemas.

Sin otro particular, me suscribo

Atentamente,


Ing. José Ricardo Morales Prado


INGENIERO EN SISTEMAS
COLEGIADO No. 4749



Universidad San Carlos de Guatemala
Facultad de Ingeniería
Escuela de Ingeniería en Ciencias y Sistemas

Guatemala, 16 de Noviembre de 2011

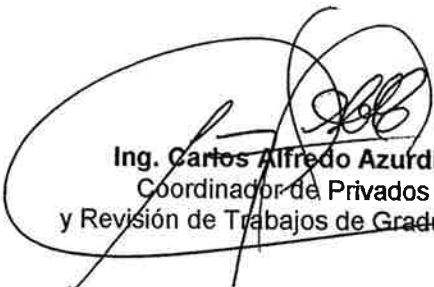
Ingeniero
Marlon Antonio Pérez Turk
Director de la Escuela de Ingeniería
En Ciencias y Sistemas

Respetable Ingeniero Pérez:

Por este medio hago de su conocimiento que he revisado el trabajo de graduación del estudiante **NESTOR GILDARDO ORDOÑEZ SALGUERO** camé 1996-17125, titulado: **“USO DEL PROCESO UNIFICADO PARA ALCANZAR EL NIVEL 2 DE MADUREZ AL DESARROLLAR SOFTWARE”**, y a mi criterio el mismo cumple con los objetivos propuestos para su desarrollo, según el protocolo.

Al agradecer su atención a la presente, aprovecho la oportunidad para suscribirme,

Atentamente,


Ing. Carlos Alfredo Azurdia
Coordinador de Privados
y Revisión de Trabajos de Graduación



E
S
C
U
E
L
A

D
E

C
I
E
N
C
I
A
S

Y

S
I
S
T
E
M
A
S

UNIVERSIDAD DE SAN CARLOS
DE GUATEMALA



FACULTAD DE INGENIERÍA
ESCUELA DE CIENCIAS Y SISTEMAS
TEL: 24767644

*El Director de la Escuela de Ingeniería en Ciencias y Sistemas de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer el dictamen del asesor con el visto bueno del revisor y del Licenciado en Letras, de trabajo de graduación titulado **“USO DEL PROCESO UNIFICADO PARA ALCANZAR EL NIVEL 2 DE MADUREZ AL DESARROLLAR SOFTWARE”**, presentado por el estudiante NESTOR GILDARDO ORDÓÑEZ SALGUERO, aprueba el presente trabajo y solicita la autorización del mismo.*

“ID Y ENSEÑAD A TODOS”



Ing. *Martín Antonio Pérez Turk*
Director, Escuela de Ingeniería Ciencias y Sistemas

Guatemala, 10 de febrero 2012



El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería en Ciencias y Sistemas, al trabajo de graduación titulado: **USO DEL PROCESO UNIFICADO PARA ALCANZAR EL NIVEL 2 DE MADUREZ AL DESARROLLAR SOFTWARE**, presentado por el estudiante universitario **Nestor Gildardo Ordoñez Salguero**, procede a la autorización para la impresión del mismo.

IMPRÍMASE.


Ing. Murphy Olimpo Paiz Recinos
DECANO



Guatemala, febrero de 2012

/cc

ACTO QUE DEDICO A:

Dios	Nuestro Creador, todo poderoso, fuente de sabiduría, amor y misericordia.
Santa María	En su advocación de Nuestra Señora del Sagrado Corazón, abogada de las causas difíciles y desesperadas.
Santa Ana	Apoyo de la vida familiar cristiana.
Mi familia	Iglesia doméstica.

AGRADECIMIENTOS A:

- Dios** Por siempre creer en mí y darme su Espíritu Santo para alcanzar esta meta.
- Mi esposa** Mirla Recinos, por el privilegio de su presencia, brindándome su amor, cariño, comprensión y apoyo.
- Nuestra hija** Andrea Gabriela, por llenar nuestro hogar de felicidad.
- Mis padres** Lidia Salguero de Ordóñez y Santos Ordóñez, por brindarme lo necesario, educarme y guiarme con su ejemplo en los caminos del amor.
- Mis hermanos** Lezztin Anabella, María Lizeth, Erick Eliezar y Carlos Anibal, por permanecer unidos en familia, que Dios las bendiga siempre.
- Mis compañeros** Álvaro Díaz y Claudia de Díaz, Whuendy Chacón y Mario Rodas, Javier Ralda y Carolina de Ralda, Sergio Rodas y Beatriz de Rodas, y Arturo Vásquez, por compartir buenos y malos momentos.

Mis amigos

Walter Bautista, Juan Barillas, Otto Calito, José María, Pablo de León, Bryan Orellana, Carlos Saz, Carlos Castillo, William Guerra, Cristian Girón y todos los que me han apoyado e instado a seguir adelante.

Mi asesor

Ing. Ricardo Morales, por todo su apoyo.

ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES	V
GLOSARIO	VII
RESUMEN.....	XI
OBJETIVOS.....	XIII
INTRODUCCIÓN	XV
1. INGENIERÍA DE <i>SOFTWARE</i>	1
1.1. Definición de Ingeniería de <i>software</i>	1
1.2. <i>Software</i>	1
1.3. Calidad del <i>software</i>	2
1.3.1. La gente	3
1.3.2. La tecnología.....	4
1.3.3. El proceso del <i>software</i>	4
1.4. Proceso de desarrollo de software.....	5
1.4.1. Problemas en el desarrollo de <i>software</i>	5
1.4.2. Causas de los problemas.....	8
1.4.3. Solución de problemas.....	10
1.5. Beneficios al contar con procesos de calidad	11
2. EL MODELO CMMI	13
2.1. Definición del modelo CMMI	13
2.2. Modelo de procesos.....	14
2.3. Representaciones del CMMI.....	15
2.4. Componentes del CMMI	16
2.5. Numeración de metas y prácticas.....	22

2.6.	Niveles del CMMI.....	23
2.6.1.	Niveles de capacidad.....	25
2.6.2.	Niveles de madurez	27
3.	EL PROCESO UNIFICADO.....	31
3.1.	Definición del Proceso Unificado	31
3.2.	Principios del Proceso Unificado	32
3.2.1.	Desarrollo iterativo del <i>software</i>	32
3.2.2.	Administración de requerimientos.....	33
3.2.3.	Uso de arquitecturas basadas en componentes.....	34
3.2.4.	Modelado visual del <i>software</i>	35
3.2.5.	Verificación de calidad del <i>software</i>	36
3.2.6.	Control de cambios	36
3.3.	Estructura del Proceso Unificado.....	37
3.3.1.	Rol	37
3.3.2.	Actividad	38
3.3.3.	Artefacto	39
3.3.4.	Flujos de trabajo	39
3.3.4.1.	Modelado del negocio	40
3.3.4.2.	Requerimientos	40
3.3.4.3.	Análisis y diseño	41
3.3.4.4.	Implementación.....	41
3.3.4.5.	Pruebas	42
3.3.4.6.	Despliegue	42
3.3.4.7.	Gestión del cambio y configuración	43
3.3.4.8.	Gestión del proyecto	43
3.3.4.9.	Entorno	43
3.3.5.	Fases en el ciclo de desarrollo	44
3.3.5.1.	Inicio	44

3.3.5.2.	Elaboración	44
3.3.5.3.	Construcción	45
3.3.5.4.	Transición.....	46
3.3.6.	Iteraciones.....	46
4.	UTILIZANDO EL PROCESO UNIFICADO COMO MARCO DE REFERENCIA PARA ALZANZAR EL NIVEL 2 DE MADUREZ DEL CMMI	49
4.1.	Nivel 2 de madurez del CMMI.....	49
4.2.	Áreas de proceso del nivel 2 de madurez del CMMI.....	50
4.3.	Uso del Proceso Unificado como marco de referencia para alcanzar el nivel 2 de madurez del CMMI	51
4.3.1.	Gestión de requerimientos (REQM)	51
4.3.1.1.	Metas y prácticas específicas	52
4.3.2.	Planificación de proyecto (PP)	56
4.3.2.1.	Metas y prácticas específicas	57
4.3.3.	Monitorización y control del proyecto (PMC).....	60
4.3.3.1.	Metas y prácticas específicas	61
4.3.4.	Gestión de Configuración (CM).....	64
4.3.4.1.	Metas y prácticas específicas	64
4.3.5.	Gestión de acuerdos con proveedores (SAM)	67
4.3.5.1.	Metas y prácticas específicas	67
4.3.6.	Aseguramiento de la calidad de proceso y de producto (PPQA).....	69
4.3.6.1.	Metas y prácticas específicas	69
4.3.7.	Medición y análisis (MA)	71
4.3.7.1.	Metas y prácticas específicas	72

5.	ACTIVIDADES DE UNA ORGANIZACIÓN GUATEMALTECA PARA ALCANZAR EL NIVEL 2 DE MADUREZ DEL CMMI.....	75
5.1.	La organización	75
5.2.	Identificación de necesidades de mejora y tendencia actual	77
5.3.	Creación del proyecto de mejora a nivel organizacional.....	78
5.4.	Adquirir conocimientos externos.....	78
5.5.	Obtención de recursos y contratación de una asesoría.....	79
5.6.	Creación del Plan de evaluación	80
5.7.	Evaluación inicial	81
5.8.	Presentación de resultados de evaluación inicial	81
5.9.	Capacitaciones	84
5.10.	Creación del Comité de mejora	85
5.11.	Elaboración del Plan de mejora.....	86
5.12.	Definición de roles y responsabilidades.....	89
5.13.	Creación de procesos.....	92
5.14.	Implementación de procesos en proyectos pilotos	93
5.15.	Evaluaciones de proyectos pilotos.....	94
5.16.	Siguientes actividades	95
	CONCLUSIONES.....	97
	RECOMENDACIONES	99
	BIBLIOGRAFÍA.....	101

ÍNDICE DE ILUSTRACIONES

FIGURAS

1.	Componentes del CMMI.....	17
2.	Estructura del Proceso Unificado.....	38
3.	Organigrama de la Gerencia de Informática.....	76
4.	Actividades de mejora de la organización.....	77
5.	Porcentaje de avance de madurez del nivel 2 del CMMI.....	83
6.	Modelo IDEAL.....	87
7.	Organigrama de la empresa consultora.....	92

TABLAS

I.	Áreas de proceso del CMMI.....	19
II.	Comparación de representaciones del modelo CMMI.....	24
III.	Distribución típica de fases por envergadura del proyecto.....	47
IV.	Roles y responsabilidades para el plan de mejora.....	89

GLOSARIO

Actividad	Conjunto de operaciones o tareas propias de una persona o entidad.
Área de proceso	Un grupo de prácticas relacionadas en un área que, cuando se implementan colectivamente, satisfacen un conjunto de metas consideradas importantes para hacer mejoras en esa área.
Capacidad	Habilidad que tiene algo o alguien para cumplir con las exigencias, objetivos o metas.
Ciclo de vida del <i>software</i>	Describe el desarrollo de <i>software</i> en período de tiempo, consistente en fases.
Control de configuración	Un elemento de la gestión de configuración que consiste en la evaluación, coordinación, aprobación o rechazo, e implementación de los cambios a los elementos de configuración, con posterioridad al establecimiento formal de su identificación de configuración.
Dato	Información registrada, sin importar la forma o el método de registro, que puede comunicarse, almacenarse y procesarse.

Desarrollo en cascada	También llamado modelo de cascada, es un enfoque metodológico que ordena rigurosamente las etapas del ciclo de vida del <i>software</i> , de tal manera que para iniciar una nueva etapa se debe terminar la anterior.
Estándar	Que sirve como tipo, modelo, norma, patrón o referencia.
Guatecompras	Sistema de información de contrataciones y adquisiciones del estado de Guatemala.
Hito	Es un producto de <i>software</i> concreto cuya obtención supone un logro importante para el proyecto.
Línea base	Un conjunto de productos de trabajo o especificaciones que se ha revisado y acordado formalmente, el cual servirá como la base para desarrollo posterior, y que solamente puede cambiarse mediante procedimientos de control de cambios.
Madurez de la organización	El grado en el cual una organización tiene explícita y consistentemente procesos desplegados que están documentados, gestionados, medidos, controlados y mejorados continuamente.

Métrica	Toda medición realizada tanto sobre los programas y su documentación asociada, como sobre el propio proceso de desarrollo y mantenimiento de <i>software</i> .
Nivel de capacidad	Logro de la mejora de procesos dentro de un área de proceso individual. Un nivel de capacidad se define por las prácticas específicas y genéricas apropiadas para un área de proceso.
Nivel de madurez	Grado de mejora de procesos a través de un conjunto predefinido de áreas de proceso en las que se alcanzan todas las metas del conjunto.
Plan del proyecto	Un plan que proporciona la base para ejecutar y controlar las actividades del proyecto, las cuales tratan los compromisos con el cliente del proyecto.
Proceso	Un conjunto de actividades interrelacionadas que transforman insumos en productos, para lograr un propósito determinado.
Producto de trabajo	Un resultado útil de un proceso, por ejemplo: ficheros, documentos, partes de un producto, servicios, descripciones de proceso, especificaciones y facturas.

Prototipo	Un tipo, forma o instancia preliminar de un producto o componente de producto que sirve como modelo para etapas posteriores o para la versión final y completa del producto.
Proveedor	Una entidad que entrega productos o realiza servicios que han sido adquiridos.
Proyecto	Un conjunto gestionado de actividades y recursos interrelacionados, incluyendo personas, que entregan uno o más productos o servicios a un cliente o usuario final.
Trazabilidad	Una asociación discernible entre dos o más entidades lógicas, tales como requerimientos, elementos de sistema, verificaciones o tareas.
Trazabilidad bidireccional	Una asociación entre dos o más entidades lógicas que es discernible en ambos sentidos, es decir hacia y desde una entidad.

RESUMEN

Los productos de *software* son de gran importancia para la vida moderna, siendo indispensable asegurar su calidad, la cual se ve influida principalmente por: la gente, el proceso y la tecnología. El presente trabajo, enfoca la mejora de la calidad del *software*, en asegurar la calidad de su proceso.

El modelo CMMI es un estándar internacional para la mejora de procesos, que proporciona a las organizaciones los elementos esenciales de procesos efectivos. Permite a las organizaciones acercarse a la mejora de procesos, mediante la representación continua y la representación escalonada. En la representación escalonada se tienen niveles de madurez, que se pueden alcanzar secuencialmente, que van desde el nivel 1 que representa un proceso inmaduro, al nivel 5 que representa un proceso de mayor madurez.

El Proceso Unificado, es un estándar para el desarrollo de *software*, es un proceso de Ingeniería de *software* configurable, que provee un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de la organización de desarrollo. Se basa en varios principios considerados como las mejores prácticas del desarrollo de *software*, que son: desarrollo iterativo del *software*, administración de requerimientos, uso de arquitecturas basadas en componentes, modelo visual del *software*, verificación de la calidad del *software* y control de cambios.

Para alcanzar el nivel 2 de madurez del modelo CMMI, las áreas de proceso que se deben cumplir son: gestión de configuración, medición y análisis, monitorización y control del proyecto, planificación de proyecto, aseguramiento de calidad de proceso y de producto, gestión de requerimientos, y gestión de acuerdos con proveedores.

El Proceso Unificado provee un marco de trabajo que ayuda a alcanzar las áreas de proceso del nivel 2 de madurez del modelo CMMI, alcanzando las metas específicas para dichas áreas de procesos, con excepción del área de proceso gestión de acuerdos con proveedores.

En el caso de estudio, se presentan los esfuerzos que una organización guatemalteca ha realizado para alcanzar el nivel 2 de madurez del modelo CMMI, y las principales actividades realizadas son: creación del proyecto de mejora a nivel organizacional, apoyo de una empresa externa experta, evaluación del estado actual de los procesos, capacitaciones sobre el modelo de mejora, creación de nuevos procesos y mejora a los procesos actuales, implementación y seguimiento de procesos mejorados, para que se conviertan en la forma común de trabajo.

OBJETIVOS

General

Identificar cómo usar el Proceso Unificado para alcanzar el nivel 2 de madurez del modelo CMMI.

Específicos

1. Describir las influencias más significativas en la calidad del *software*.
2. Describir problemas que se dan en el desarrollo de *software*, sus causas y posible solución.
3. Dar a conocer la importancia de tener desarrollos de *software* de calidad.
4. Dar a conocer los principales conceptos del modelo CMMI.
5. Describir el Proceso Unificado.
6. Identificar cómo hacer uso del Proceso Unificado para alcanzar el nivel 2 de madurez del modelo CMMI.
7. Dar a conocer los esfuerzos que una organización guatemalteca está realizando para alcanzar el nivel 2 de madurez del modelo CMMI.

INTRODUCCIÓN

En el presente trabajo de graduación, se darán conceptos relacionados con los productos de *software*, identificando las necesidades de contar con productos de calidad, dando una definición de las tres grandes influencias en dicha calidad, que son: el proceso, la gente y la tecnología.

Se centra en el proceso de desarrollo de *software*, para lo cual se estudiará el Proceso Unificado de desarrollo de *software* o simplemente Proceso Unificado, que es un estándar, el cual implementa muchas de las mejores prácticas concebidas hasta ahora para el desarrollo de *software*.

De la calidad en el desarrollo de *software*, se estudiará el modelo CMMI (*Capability Maturity Model Integration*), que es una marca registrada del SEI (*Software Engineering Institute*) de la Universidad Estadounidense *Carnegie Mellon*.

El CMMI es un modelo para la mejora de los procesos de desarrollo de *software*, el cual tiene dos representaciones que son: escalonada y secuencial. En la representación escalonada, el CMMI identifica niveles de madurez, que van desde el nivel 1 que identifica un proceso inmaduro, hasta el nivel 5 que identifica a un proceso de mayor madurez, los cuales a su vez poseen áreas de proceso, y metas a alcanzar para obtener cada área de proceso.

Se estudia el nivel 2 de madurez del CMMI como objetivo a alcanzar, usando el Proceso Unificado para alcanzarlo, identificando cómo alcanzar las metas de cada área de proceso relacionada al desarrollo de *software*, que pertenecen al nivel 2 del CMMI, en su representación escalonada.

Se hará un estudio en una organización guatemalteca, la cual está actualmente en proceso de obtener el nivel 2 de madurez del CMMI, identificando las actividades realizadas, para que sirvan de referencia a otras organizaciones.

1. INGENIERÍA DE SOFTWARE

1.1. Definición de Ingeniería de *software*

Ingeniería de *software* es la disciplina o área de la informática o ciencias de la computación, que ofrece métodos y técnicas para desarrollar y mantener *software* de calidad, para la resolución de problemas.

Dicha definición es acertada para el presente trabajo, pues contiene ese elemento que en la actualidad se ha tomado con mayor importancia, que es la calidad, pues muchos estudios han identificado que la falta de ella ha ocasionado pérdidas cuantiosas para muchas empresas.

1.2. *Software*

El *hardware* es la parte física de las computadoras, y más ampliamente, es cualquier dispositivo electrónico. El *software* es el conjunto de instrucciones que permiten al *hardware* de la computadora, desempeñar trabajo útil.

El *software* es la meta de la Ingeniería de *software*, ya sea adquiriéndolo, creándolo o manteniéndolo. Haciendo la analogía con cualquier producto de compra-venta, en el presente trabajo se llama al *software* final, como el producto de *software*, o simplemente *software*, que es lo que se entrega al cliente. Otra forma de visualizar al *software* es como servicio.

1.3. Calidad del *software*

En todo producto de ingeniería, al igual que cualquier producto científico, la calidad es de gran importancia, y en ocasiones resulta indispensable. Para la Ingeniería de *software* no es la excepción, pues aunque no siempre se note, el *software* está íntimamente relacionado con la mayoría de actividades en la sociedad moderna, resultando en muchas ocasiones importante para mantener la vida humana.

Para definir la calidad, existen varios puntos de vista:

- La visión trascendental: aquella que se puede reconocer pero no se puede definir.
- La visión de manufactura: donde la calidad es conformidad con la especificación.
- La visión del producto: vinculada a las características inherentes del producto.
- La visión del usuario: que es adecuación al propósito.
- La visión basada en valor: dependiendo la cantidad de dinero que el usuario está dispuesto a pagar.

Cada una de estas visiones de calidad, aplica igualmente al definir la calidad del *software*. La definición dada por R. S. Pressman (1992), presentada a continuación, es adecuada por los elementos que contiene:

“Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo *software* desarrollado profesionalmente”.

Según algunos estudios y esfuerzos, se ha identificado que la calidad del *software* se ve influida principalmente por tres grandes factores, los cuales son:

- La gente
- La tecnología
- El proceso

A continuación una descripción e importancia de cada uno de ellos:

1.3.1. La gente

Las personas involucradas en la Ingeniería de *software*, gracias a sus habilidades naturales, son las responsables en definir, administrar, modificar y comprender, los diferentes procesos que tienen como fin la obtención del producto de *software*. Son las encargadas de aplicar las diferentes herramientas de *software* y utilizar el *hardware* disponible, para la creación o modificación del producto de *software*, aplicando sus conocimientos y habilidades para obtener un producto de *software* de calidad.

En mayor o menor grado, dependiendo el rol desempeñado en todo el proceso que conlleva la Ingeniería de *software*, cada persona involucrada aporta o resta calidad a su trabajo, viéndose reflejada directamente en la calidad del producto de *software*.

Es indispensable que se dé la importancia debida a las personas en la Ingeniería de *software*, ayudándolas en su forma de contrato, horarios de trabajo, capacitaciones, etc., contribuyendo así a tener buena motivación y actitud hacia el proceso.

1.3.2. La tecnología

Tecnología es un concepto amplio, que se define como un conjunto de teorías y técnicas que permiten el aprovechamiento práctico del conocimiento científico. En nuestro caso, interesa la tecnología informática, que es la tecnología de procesamiento de información por medios artificiales, entre los que se incluye a las computadoras.

Otra definición de tecnología informática, indica que es el estudio, diseño, desarrollo, puesta en marcha, ayuda de los sistemas informáticos computarizados, particularmente usos del *software* y *hardware*.

En el desarrollo del *software*, por lo regular al inicio se realiza la mayor inversión en obtener la tecnología informática que será utilizada, y es allí principalmente cuando se debe tener una buena orientación y discernimiento en obtener la adecuada.

1.3.3. El proceso del *software*

Proceso del *software* es el marco de trabajo de las tareas, que son requeridas para construir o mantener *software* de alta calidad.

La calidad de la gente es indispensable, pero el contar con un proceso documentado, entendible, administrable, y en resumen de calidad, ayuda a las personas, para adquirir, desarrollar o mantener el producto de *software* con la calidad deseada.

El proceso orienta a las personas a realizar su trabajo, por ello es indispensable contar con dicha orientación adecuada al propósito.

Gran cantidad de esfuerzo que se ha realizado últimamente, está enfocado a mejorar la calidad del proceso, pues experiencias de varias organizaciones han demostrado que influye grandemente en la calidad del producto de *software* final.

1.4. Proceso de desarrollo de *software*

El desarrollar o mantener *software* se debe realizar haciendo uso de un proceso que sea de calidad. Para obtener dicha calidad es aconsejable que sea definido siguiendo un modelo, logrando así mayor probabilidad de éxito.

A continuación se indican algunos problemas que se dan en el desarrollo de *software*, sus causas, y la solución propuesta.

1.4.1. Problemas en el desarrollo de *software*

Lo primero que se debe realizar, es una identificación adecuada de los problemas al desarrollar *software* en la organización, pues ello es el origen de una mejora.

Los problemas que se presentan, provienen de muchas fuentes, y se ponen en evidencia desde el inicio de dicho proceso. Las organizaciones deben poseer vías de trabajo establecidas para identificar dichos problemas, y no buscar problemas inútilmente. En dicha búsqueda de problemas se deben encontrar los más significativos, tomando en cuenta que esa búsqueda en sí no es el fin último.

Se pueden identificar varios problemas en el proceso de desarrollo de *software*, entre los cuales están:

- No se alcanzan los objetivos: regularmente se intenta alcanzar mucho más de lo que se termina haciendo, al sobreestimar capacidades de recursos, desarrolladores, etc. No se posee adecuada administración de riesgos.
- Inadecuada identificación de requerimientos: no se logra entender lo que se desea en realidad, al no establecerse adecuadamente las necesidades del usuario final. No se conoce al detalle qué debe y qué no debe hacer el sistema.
- Inadecuada identificación de requerimientos de cambio: muchas veces no se miden los alcances de los requerimientos, subestimando o sobreestimando costos, tiempo y recursos requeridos. No se mantiene control de requerimientos nuevos o cambios a los existentes.
- Manejo inadecuado de requerimientos: no se poseen controles específicos para los requerimientos, dejando al lado el seguimiento de los mismos, hasta la entrega final del producto.
- Estimaciones pobres: se es regularmente muy optimista para hacer la entrega a tiempo, además de establecer costos que generalmente quedan muy por debajo de lo que se termina gastando.
- Módulos que no se integran fácilmente: se es muy bueno en reunir los requerimientos, sin embargo malo en diseñar sistemas. Se desarrollan módulos que a la larga no encajan, o no se integran fácilmente, observándose un incremento en el tiempo planificado de desarrollo e integración del sistema.

- Desarrollo de productos *software* de baja calidad: los productos *software* tienen muchos defectos, los cuales pueden ser debido a falta de pruebas, pero usualmente están relacionados a la inhabilidad para capturar y manejar requerimientos, así como diseños deficientes. Con el afán de entregar productos *software* a tiempo, se centra en la funcionalidad, dejando atrás la calidad que se requiere.
- Inaceptables rendimientos de *software*: el tiempo de respuesta del *software* final es tardado, no se posee código ordenado, se verificó la funcionalidad pero no así el rendimiento que se esperaba del sistema.
- Bajo uso: se puede tardar mucho tiempo desarrollando *software* que a la larga puede ser poco usado, por alto nivel de incertidumbre, resistencia al cambio, poca experiencia, relación costo beneficio de la herramienta. Lo que se debió identificar en los requerimientos para la asignación de recursos en el desarrollo de dicho producto de *software*.
- Conflicto entre desarrolladores: hay una falta de control sobre la propiedad y administración de la configuración, por tanto los desarrolladores hacen cambios contrarios estropeando el trabajo de otro. Se tiene una incapacidad de reconstruir quién cambió, qué, cuándo, dónde y porqué.
- Descubrimiento tardío de problemas: un ejemplo de ello se identifica al no dar seguimiento al *software* puesto en marcha, por lo tanto no se pueden detectar problemas que pudieran darse, y que en el desarrollo no fueron encontrados. En el desarrollo se puede dar, entre otras cosas, cuando no se posee un adecuado control de requerimientos, por lo tanto al venir un requerimiento nuevo, no se logra establecer el efecto que tendrá en otros requerimientos o módulos ya realizados.

1.4.2. Causas de los problemas

Identificados los problemas, es necesario identificar sus causas. Entre dichas causas están:

- Insuficiente administración de requerimientos: no se lleva el respectivo control tanto en requerimientos iniciales, nuevos y los cambios a éstos. No se documentan, no hay un adecuado análisis de impacto de los requerimientos.
- Comunicación ambigua e imprecisa: tanto entre desarrolladores, como con el cliente, usuarios e interesados del sistema, que pueden causar conflictos y confusiones.
- Arquitecturas frágiles: desarrollos independientes, los componentes son defectuosos, los componentes no poseen adecuadas interfaces, no se maneja seguridad, no existe reutilización de *software*, los sistemas no son tolerantes a cambios.
- Complejidad agobiante: no se documenta, se desarrolla de una manera desordenada y no entendible, incapacidad para poder encontrar determinado error en la programación.
- Pruebas insuficientes: no existe plan de pruebas, las pruebas se realizan por lo general hasta el final del desarrollo.
- Inconsistencias no detectadas entre requerimientos, diseños e implementaciones: mala administración de requerimientos, aunado en un inadecuado plan de pruebas. El cliente hizo inicialmente sus

requerimientos, y no se le entregó un ejecutable hasta la finalización del sistema.

- Valoración subjetiva del estatus del proyecto: no se sabe a ciencia cierta cómo se encuentra actualmente el proyecto.
- Retardada reducción de riesgos debido a diseño en cascada: el diseño en cascada retrasa actividades de integración y pruebas, no permite puesta en marcha temprana, no identificando problemas debido a que las pruebas se realizan hasta finalizada la elaboración del *software*. No contempla cambios al sistema una vez obtenidos los requerimientos.
- Propagación incontrolable de cambios: no existe historial de cambios ni análisis de impacto en los requerimientos.
- Insuficiente automatización: no se poseen herramientas de *software* o resultan inadecuadas, para ayudar en el manejo de requerimientos, tareas, cambios, defectos, configuración, y demás actividades del desarrollo.
- Inadecuada construcción de interfaces: no hay lineamientos al desarrollar interfaces, todos desarrollan a su manera, hay desarrollo caótico.

1.4.3. Solución de problemas

Identificar los problemas y sus posibles causas, tiene como finalidad darles solución. La solución aquí propuesta para mejorar la calidad final del *software*, es mejorar el desarrollo de *software*, utilizando el denominado Proceso Unificado de desarrollo de *software* o simplemente Proceso Unificado.

Contar con un proceso de desarrollo de *software* probado y aceptado internacionalmente, es un gran paso, pero en la actualidad se está volviendo más indispensable que una organización pueda comprobar que su desarrollo es de calidad, misma que se puede ver a lo externo, en una certificación.

Para certificar a una empresa en desarrollo de *software*, se presentan varias normas y estándares, entre los cuales encontramos el modelo CMMI (*Capability Maturity Model Integration*).

Debido a que nuestro objetivo no es la certificación, utilizaremos el modelo CMMI como orientación para mejorar el proceso de desarrollo.

Se puede identificar la rentabilidad de adoptar un proceso de calidad, al generar por lo menos uno de los siguientes resultados:

- Menores costos
- Menor tiempo de entrega
- Mejoramiento de la calidad del producto o servicio

Se estudia en este trabajo, cómo el Proceso Unificado ayuda a obtener el nivel 2 de madurez del modelo CMMI, el cual es un modelo internacional, del que se puede certificar el proceso de desarrollo de *software* de una organización.

El modelo CMMI nos indica el qué hacer, mientras el Proceso Unificado nos indica cómo hacerlo. De allí la importancia individual de ambos, y la unión de ellos.

1.5. Beneficios al contar con procesos de calidad

Entre los beneficios de contar con un proceso de desarrollo de *software* basado en el modelo CMMI están:

- Entender lo que está pasando.
- Que el personal desarrolle su potencial más eficientemente y más completamente, dentro de la organización.
- Más probabilidad de éxito.
- Hacer a la organización más productiva, pues se basa en la definición, medición y control de los procesos.
- Se incrementa sensiblemente la probabilidad de éxito en la introducción de nuevas y apropiadas tecnologías, técnicas y herramientas en la organización.
- Proporciona un marco de trabajo que permite organizar y priorizar las actividades de mejora de procesos, involucrando al propio producto, a la organización, a la gente y a la tecnología.
- Da soporte a la coordinación de actividades multidisciplinarias que pueden ser necesarias para construir con éxito un determinado producto.
- Enfatiza el alineamiento de los objetivos de la mejora de procesos con los objetivos de negocio de la organización.

- Se implementan prácticas más robustas y maduras en la organización.
- Mejora la satisfacción del cliente.
- Reducción de costos.
- Mejora en el tiempo de entrega.
- Reducción de defectos del *software*.
- Mejor retorno de la inversión.

2. EL MODELO CMMI

2.1. Definición del modelo CMMI

CMMI (*Capability Maturity Model Integration*, o Integración de Modelos de Madurez de Capacidades, por su traducción del inglés), es una marca registrada del SEI (*Software Engineering Institute*), de la universidad estadounidense Carnegie Mellon. CMMI es un enfoque de mejora de procesos que proporciona a las organizaciones los elementos esenciales de procesos efectivos. Es un estándar resultado de la experiencia de la industria, y de la integración de estándares, pues su contenido integra y da relevo a la evolución de sus predecesores.

En la versión 1.3 actual del CMMI, existen tres modelos, los cuales son colecciones de las mejores prácticas que ayudan a la organización a mejorar sus procesos. Las áreas de interés cubiertas por los modelos del CMMI son: desarrollo, adquisición y servicios.

- El modelo CMMI para el desarrollo: ha surgido como una necesidad de desarrollar con calidad, incluyendo buenas prácticas reconocidas, buenas referencias para fijar objetivos y prioridades. Las mejores prácticas en este modelo se centran en las actividades para el desarrollo de productos y servicios de calidad para satisfacer las necesidades de los clientes y usuarios finales. En los predecesores de este modelo la orientación era hacia el desarrollo y mantenimiento de *software*, pero en la versión actual de este modelo las mejores prácticas aplican a cualquier empresa de desarrollo.

- El modelo CMMI para adquisición: proporciona un completo conjunto integrado de directrices para la adquisición de productos y servicios. Las mejores prácticas en este modelo se centran en las actividades para iniciar y gestionar la adquisición de productos y servicios para satisfacer las necesidades de los clientes y usuarios finales.
- El modelo CMMI para servicios: proporciona un completo conjunto integrado de directrices para la prestación de servicios de calidad. Ofrece guías para aplicar las mejores prácticas CMMI en una organización proveedora de servicios. Las mejores prácticas en este modelo se centran en las actividades de prestación de servicios de calidad a los clientes y usuarios finales.

Los tres modelos CMMI son complementarios, pues el comprador debe utilizar el modelo CMMI para adquisición, y el proveedor utilizar el modelo CMMI para el desarrollo o CMMI para servicios, dependiendo si se compra un producto o un servicio respectivamente.

En el presente capítulo se estudia el CMMI, y en los casos que aplique se centra en el modelo CMMI para el desarrollo, aplicado a organizaciones de desarrollo y mantenimiento de sistemas y productos de *software*.

2.2. Modelo de procesos

El CMMI es un modelo de procesos, pues es una colección estructurada de las prácticas que describen las características de los procesos eficaces.

Un modelo de procesos es usado para:

- Establecer objetivos de mejora de procesos y prioridades
- Asegurar la estabilidad, capacidad y madurez de los procesos
- Guiar la mejora de los proyectos y procesos de la organización
- Como un método de evaluación para diagnosticar el estado actual de las prácticas en la organización.

Un modelo de procesos es importante porque provee:

- Un lugar de inicio para mejorar
- Los beneficios que dan las experiencias en una comunidad
- Un lenguaje común con una visión compartida
- Un marco para priorizar acciones
- Una forma de definir lo que significa la mejora de una organización

2.3. Representaciones del CMMI

El CMMI permite acercarse a la mejora de procesos mediante dos representaciones: la representación continua y la representación escalonada.

- A. Representación continua: permite a la organización seleccionar un área de proceso o grupo de áreas de proceso, y mejorar sus procesos relacionados. Esta representación utiliza niveles de capacidad para caracterizar las mejoras relacionadas con un área de proceso.
- B. Representación escalonada: utiliza conjuntos predefinidos de áreas de proceso, para definir el camino de mejora de la organización. Esta ruta de mejora se caracteriza por niveles de madurez. Desde el nivel 2 de

madurez en adelante, cada nivel de madurez provee un conjunto de áreas de proceso que caracterizan diferentes comportamientos en la organización.

Si se conoce el área de proceso que se desea mejorar y su dependencia con el resto en el CCMI, se recomienda la representación continua, que es más flexible para la mejora de procesos.

La representación escalonada ofrece una forma estructurada y sistemática de abordar el modelo, basado en la mejora de procesos de un escalón o nivel a la vez. La representación escalonada se recomienda en la mayoría de casos, pues se avanza progresivamente mediante un camino definido, además contiene un orden aceptado para realizar cambios.

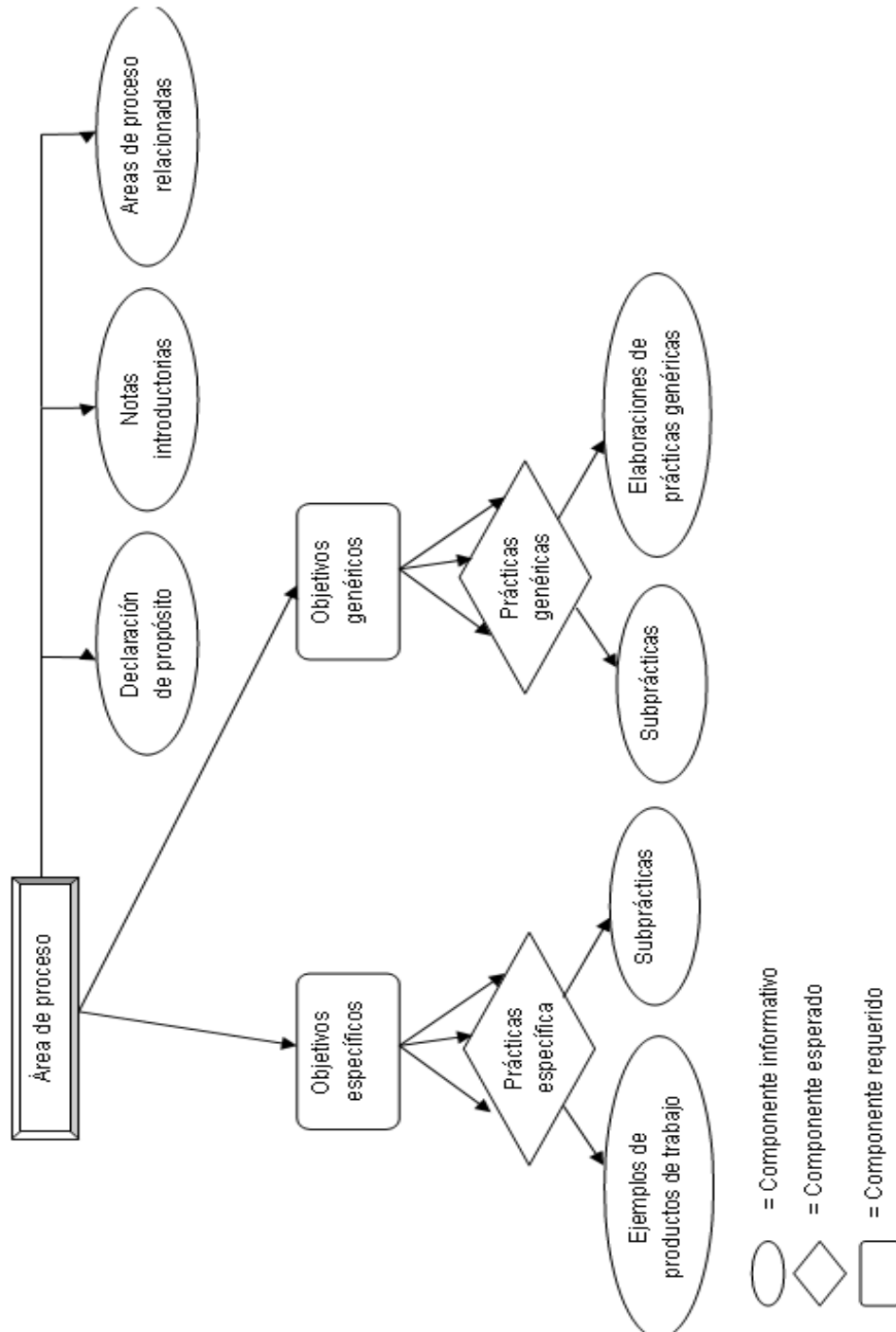
Ambas representaciones proporcionan los medios para aplicar la mejora de procesos, y así alcanzar los objetivos de la organización. Ambas representaciones proporcionan el mismo contenido esencial, y utilizan el mismo modelo de componentes.

2.4. Componentes del CMMI

Los componentes del CMMI se presentan en la figura 1, y se agrupan en tres categorías que son: componentes requeridos, componentes esperados y componentes informativos.

Componentes requeridos: describen lo que se debe alcanzar para satisfacer un área de proceso. Los componentes requeridos son las metas específicas y las metas genéricas.

Figura 1. Componentes del CMMI



Fuente: SEI, CMMI-DEV, <http://www.sei.cmu.edu/reports/10tr033.pdf>. Febrero de 2011.

Componentes esperados: describen lo que se debe alcanzar para satisfacer un componente requerido. Sirven de guía a los que aplican o realizan las mejoras. Los componentes esperados incluyen las prácticas genéricas y las prácticas específicas.

Componentes informativos: proponen la manera de empezar a abordar los componentes esperados y los componentes requeridos. Entre los componentes informativos tenemos: sub prácticas, ejemplos de productos de trabajo, ampliaciones, elaboraciones de las prácticas genéricas, títulos de las metas y prácticas, notas, y las referencias.

A continuación se da una descripción de cada uno de los componentes del CMMI indicados en figura 1.

Área de proceso: en CMMI cada área de proceso es un conjunto de actividades relacionadas, que al momento de aplicarse en su totalidad, cumplen una serie de objetivos considerados importantes para tener una mejora significativa en esa área.

Las áreas de proceso del CMMI para el desarrollo, se agrupan en categorías principales, de manera que puedan apoyar a quienes eligen la representación continua, las cuales son:

- A. Gestión de proyectos: las áreas de proceso agrupadas en ésta categoría son las que cubren la planeación, monitoreo y control de los proyectos.

- B. Gestión de procesos: las áreas de proceso agrupadas en esta categoría son aquellas que contienen actividades cruzadas con los proyectos y que están relacionadas con definir, planificar, desplegar, implementar, monitorear, evaluar, controlar, medir y mejorar los procesos.
- C. Ingeniería: las áreas de proceso agrupadas en ésta categoría cubren el desarrollo y mantenimiento de actividades que son compartidas a través de varias disciplinas de la ingeniería, tales como la ingeniería de *software* o la ingeniería mecánica.
- D. Soporte: las áreas de proceso agrupadas en ésta categoría cubren las actividades que apoyan el desarrollo y mantenimiento del producto.

En el modelo CMMI para el desarrollo, existen 22 áreas de proceso, identificadas mediante acrónimos de sus correspondientes nombres en inglés. En la tabla I se listan ordenadas alfabéticamente según sus acrónimos, las áreas de proceso del CMMI, identificando la categoría y el nivel de madurez al que pertenecen.

Tabla I. **Áreas de proceso del CMMI**

ÁREA DE PROCESO	CATEGORÍA	NIVEL DE MADUREZ
Análisis de causas y resolución (CAR)	Soporte	5
Gestión de configuración (CM)	Soporte	2
Análisis de decisiones y resolución (DAR)	Soporte	3
Gestión integrada de proyectos (IPM)	Gestión de Proyectos	3

Continúa tabla I.

Medición y análisis (MA)	Soporte	2
Definición de proceso organizacional (OPD)	Gestión de procesos	3
Enfoque en procesos de la organización (OPF)	Gestión de procesos	3
Gestión de rendimiento organizacional (OPM)	Gestión de procesos	5
Rendimiento del proceso organizacional (OPP)	Gestión de procesos	4
Formación organizacional (OT)	Gestión de procesos	3
Integración de producto (PI)	Ingeniería	3
Monitorización y control de proyecto (PMC)	Gestión de proyectos	2
Planificación de proyecto (PP)	Gestión de proyectos	2
Aseguramiento de calidad de proceso y de producto (PPQA)	Soporte	2
Gestión cuantitativa de proyecto (QPM)	Gestión de proyectos	4
Desarrollo de requerimientos (RD)	Ingeniería	3
Gestión de requerimientos (REQM)	Gestión de proyectos	2
Gestión de riesgos (RSKM)	Gestión de proyectos	3

Continúa tabla I.

Gestión de acuerdos con proveedores (SAM)	Gestión de proyectos	2
Solución técnica (TS)	Ingeniería	3
Validación (VAL)	Ingeniería	3
Verificación (VER)	Ingeniería	3

Fuente: SEI, CMMI-DEV, <http://www.sei.cmu.edu/reports/10tr033.pdf>. Febrero 2011.

Declaración de propósito: Describe la finalidad del área de proceso.

Notas introductorias: describen los principales conceptos incluidos en el área de proceso.

Áreas de proceso relacionadas: lista las referencias a otras áreas de proceso relacionadas.

Metas específicas: describen las características que deben estar presentes para satisfacer el área de proceso. Se aplican a una única área de proceso.

Prácticas específicas: es la descripción de una actividad considerada importante en la realización de la meta específica a la cual está asociada.

Metas genéricas: describen las características que deben estar presentes para institucionalizar los procesos que implementan un área de proceso. Se llaman genéricas porque se aplican a múltiples áreas de proceso.

Prácticas genéricas: es la descripción de una actividad considerada importante para alcanzar la meta genérica asociada. Se aplican a múltiples áreas de proceso.

Ejemplos de productos de trabajo: lista ejemplos de resultados de una práctica específica.

Subprácticas: es una descripción detallada que sirve de guía para la interpretación e implementación de una práctica genérica o práctica específica.

Elaboraciones de prácticas genéricas: es una guía de cómo la práctica genérica debe aplicarse a una determinada área de proceso.

2.5. Numeración de metas y prácticas

Las metas genéricas y metas específicas están numeradas en forma secuencial. Cada meta específica se identifica con el prefijo SG (*Specific Goal*, por sus siglas en inglés) seguido por un número, por ejemplo SG 1, igualmente cada meta genérica se identifica con el prefijo GG (*Generic Goal*, por sus siglas en inglés) seguido por un número, por ejemplo GG 1.

Cada práctica específica inicia con el prefijo SP (*Specific Practice*, por sus siglas en inglés) seguido por un número en la forma x.y (ejemplo SP 1.1). La “x” es el mismo número que la meta específica mapeada a la práctica específica, mientras que la “y” corresponde al número de secuencia de la práctica específica dentro de la meta específica.

Cada práctica genérica inicia con el prefijo GP (*Generic Practice*, por sus siglas en inglés) seguido por un número en la forma x.y (ejemplo GP 1.1). La “x” es el mismo número que la meta genérica mapeada a la práctica genérica, mientras que la “y” corresponde al número de secuencia de la práctica genérica dentro de la meta genérica.

2.6. Niveles del CMMI

Los niveles se utilizan para describir un camino evolutivo, recomendado para las organizaciones que quieren mejorar los procesos que utilizan para desarrollar y mantener sus productos y servicios.

El CMMI soporta dos caminos para la mejora. En el primer camino de mejora, que es la representación continua, se utiliza el término niveles de capacidad. Para el segundo camino que es la representación escalonada, se utiliza el término niveles de madurez.

Para alcanzar un nivel, la organización debe cubrir la totalidad de las metas correspondientes al área de proceso o conjunto de áreas de proceso a mejorar, independientemente de si se trata de un nivel de capacidad o un nivel de madurez.

Los niveles de capacidad aplican a un proceso de mejora de la organización, logrados en áreas de proceso individuales, siendo cada vez más un medio para mejorar los procesos correspondientes a una determinada área de proceso. En la versión 1.3 actual del CMMI, existen cuatro niveles de capacidad numerados del 0 al 3.

Los niveles de madurez aplican a un proceso de mejora de la organización, a través de múltiples áreas de proceso. Estos niveles de madurez son un medio para predecir el resultado del próximo proyecto emprendido. Actualmente existen cinco niveles de madurez, que van del 1 al 5.

En la tabla II, se presenta una comparación entre los niveles de capacidad y los niveles de madurez del CMMI.

La representación continua se refiere a la selección de un área de proceso particular para mejorar y el nivel deseado de capacidad para el área de proceso. En tal sentido es importante conocer si un determinado proceso es realizado o incompleto, por ello el nivel 0 (cero) con nombre Incompleto da a la representación continua el punto de partida.

Tabla II. Comparación de representaciones del modelo CMMI

Nivel	Representación Continua Niveles de Capacidad	Representación Escalonada Niveles de Madurez
Nivel 0	Incompleto	
Nivel 1	Realizado	Inicial
Nivel 2	Gestionado	Gestionado
Nivel 3	Definido	Definido
Nivel 4		Gestionado cuantitativamente
Nivel 5		En optimización

Fuente: SEI, CMMI-DEV, <http://www.sei.cmu.edu/reports/10tr033.pdf>. Febrero de 2011.

Debido a que la representación escalonada se refiere a la madurez de la organización, mediante múltiples áreas de proceso, en tal sentido si los procesos individuales son incompletos o realizados no es el objetivo principal.

Por lo tanto el nombre de Inicial se da al punto de partida para los niveles en la representación escalonada.

2.6.1. Niveles de capacidad

Los niveles de capacidad caracterizan la mejora de la organización en relación con un área de proceso individual.

- Nivel 0 Incompleto: un proceso en este nivel es aquel que no se realiza o se realiza parcialmente. Una o más metas específicas no se cumplen, además no existen metas genéricas en este nivel pues no hay razón para institucionalizar un proceso que se ejecuta parcialmente.
- Nivel 1 Realizado: se caracteriza por ser un proceso realizado. Un proceso que se realiza es aquel que lleva a cabo el trabajo necesario para producir productos de trabajo, lo que implica que las metas específicas del área de proceso fueron satisfechas. En este nivel no se han institucionalizado las mejoras al proceso o procesos.
- Nivel 2 Gestionado: un proceso gestionado se caracteriza por ser planificado y ejecutado de acuerdo con las políticas, además de contar con personal calificado con recursos necesarios para producir salidas controladas, involucrando a las partes interesadas. Es monitoreado, controlado y revisado.
- Nivel 3 Definido: un proceso definido es un proceso gestionado que se adapta a las directrices de la organización. Se ha mantenido una descripción del proceso. Provee experiencia a procesos relacionados en los activos de procesos de la organización.

Una distinción fundamental entre los niveles de capacidad 2 y 3, es el alcance de los estándares, descripciones de procesos y procedimientos. En el nivel 2 pueden ser muy diferentes en cada instancia específica del proceso. En el nivel 3, los estándares, descripciones de procesos y procedimientos para un proyecto se adaptan al conjunto de procedimientos estándar de la organización, y por tanto es más consistente. Otra diferencia fundamental es que en el nivel 3, los procesos suelen ser descritos con mayor rigor que en el nivel 2.

Para alcanzar un nivel de capacidad de un área de proceso, se deben aplicar las prácticas genéricas o alternativas adecuadas a los procesos asociadas a esa área de proceso.

Alcanzar el nivel 2 de capacidad para un área de proceso, es igual a decir que es una política que indica cómo llevar a cabo el proceso.

Alcanzar el nivel 3 de capacidad para un área de proceso, es igual a decir que un proceso estándar de la organización existe asociado a esa área de proceso, que puede ser adaptado a las necesidades del proyecto. Los procesos de la organización son ahora más definidos y consistentemente aplicados, ya que se basan en procesos estándar con los que cuenta la organización.

2.6.2. Niveles de madurez

Un nivel de madurez consiste en prácticas relacionadas específicas y genéricas para un conjunto predefinido de áreas de proceso, que mejoran el rendimiento general de la organización.

Cada nivel de madurez, madura un subconjunto importante de los procesos de la organización, y lo prepara para el siguiente nivel de madurez.

Los niveles de madurez se miden por el logro de las metas específicas y genéricas asociadas a cada conjunto predefinido de áreas de proceso. Los niveles de madurez se utilizan para caracterizar la mejora organizacional en relación a un conjunto de áreas de proceso.

Los niveles de madurez 2 y 3 utilizan los mismos términos que sus correspondientes niveles de capacidad 2 y 3, intencionalmente porque dichos conceptos son complementarios.

A continuación se presentan los niveles de madurez del modelo CMMI, identificando características de los mismos.

- Nivel 1 Inicial: los procesos suelen ser caóticos, no documentados, el éxito del proyecto se atribuye a una o un grupo reducido de personas, y no al uso de procesos aprobados. En cuanto al éxito de algunos proyectos, regularmente se gasta más de lo previsto, y se desarrollan en más tiempo de lo planeado.

En este nivel las organizaciones se caracterizan por comprometerse demasiado, por abandonar los procesos en momentos de crisis, y ser incapaces de repetir sus éxitos.

Para el nivel inicial no se tienen áreas de proceso definidas, por tanto no se tienen metas específicas o metas genéricas a alcanzar, estando en este nivel aquellas organizaciones que desean seguir escalando en los posteriores niveles de madurez.

- Nivel 2 Gestionado: los procesos se planifican y ejecutan de acuerdo a las políticas. Los proyectos emplean a personal calificado, que dispone de recursos suficientes para producir salidas controladas, que afectan a las partes interesadas. Los procesos se supervisan, controlan y revisan. Se da un mejor manejo en tiempo de estrés. Los planes están documentados.

En este nivel los estatus de los productos de trabajo son visibles a la gestión. Los compromisos se establecen entre los interesados, y son revisados según sea necesario.

Los productos de trabajo son controlados adecuadamente. Los productos de trabajo y servicios satisfacen las descripciones de procesos, estándares y procedimientos.

- Nivel 3 Definido: los procesos están bien caracterizados y entendidos, y se describen en las normas, procedimientos, herramientas y métodos. El conjunto de procesos estándar de la organización se ha definido y mejorado con el tiempo.

Una distinción fundamental entre el nivel 3 y el nivel 2 de madurez, es el alcance de las normas, descripciones de procesos y procedimientos. En el nivel 2, estas pueden ser bastante diferentes en cada instancia del proceso, mientras que en el nivel 3 son más consistentes. Otra diferencia es que en el nivel 3, los procesos suelen ser definidos con mayor rigor que en el nivel 2. Un proceso bien definido posee claramente los estados, propósitos, insumos, criterios de ingreso, actividades, roles, medidas, pasos de verificación, resultados, y los criterios de salida.

- Nivel 4 Gestionado cuantitativamente: en este nivel, en la organización y los proyectos se establecen objetivos cuantitativos para la calidad y desempeño de procesos, los cuales son usados como criterios en la gestión de proyectos. Los objetivos cuantitativos se basan en las necesidades del cliente, la organización, los usuarios finales, y los ejecutores del proceso. La calidad y el rendimiento del proceso se manejan en términos estadísticos y se gestiona toda la vida de los proyectos.

Para subprocesos seleccionados, las medidas específicas de rendimiento del proceso, son recolectadas y analizadas estadísticamente.

Una diferencia fundamental entre el nivel 4 y el nivel 3 de madurez, es la previsibilidad de los resultados del proceso. En el nivel 4 el rendimiento de proyectos y subprocesos seleccionados es controlado usando la estadística y otras técnicas cuantitativas.

- Nivel 5 En optimización: la organización continuamente mejora sus procesos basada en una comprensión cuantitativa de sus objetivos de negocio y sus necesidades de rendimiento. La organización utiliza un enfoque cuantitativo para entender la variación inherente del proceso y las causas de los resultados del proceso.

Este nivel se centra en la mejora continua del rendimiento de procesos mediante mejoras incrementales e innovadoras de procesos y tecnología.

En el nivel 4 de madurez, la organización y los proyectos se centran en la comprensión y el control de los resultados a nivel de subprocesos, y utilizando dichos resultados para administrar proyectos. En el nivel 5 de

madurez, la organización utiliza los datos recolectados de varios proyectos, para ver su rendimiento general.

Los niveles de madurez están orientados a lograr mejoras progresivas, considerando primero el control a nivel de proyecto, para luego mejorar a nivel de organización, mediante datos cualitativos y cuantitativos para la toma de decisiones.

La madurez organizaciones es una forma de predecir los resultados generales del próximo proyecto. Por ejemplo en el nivel 2, la organización ha pasado de producir proyectos de forma caótica a una forma disciplinada mediante la gestión de proyectos. A medida que la organización alcanza las metas específicas y metas genéricas del conjunto de áreas de proceso de un nivel de madurez, aumenta su madurez organizativa y por ende obtiene los beneficios de la mejora de procesos.

Cada nivel de madurez forma una base necesaria para el posterior nivel.

3. EL PROCESO UNIFICADO

3.1. Definición del Proceso Unificado

El Proceso Unificado de desarrollo de *software*, también llamado Proceso unificado de desarrollo, o simplemente Proceso Unificado, es un proceso de ingeniería de *software* configurable, que se adapta a los proyectos variados en tamaño y complejidad.

Provee un enfoque disciplinado para la asignación de tareas, y responsabilidades dentro de la organización de desarrollo. Su objetivo principal es asegurar la producción de *software* de alta calidad, basada en las necesidades y requerimientos del usuario final, dentro de un cronograma y presupuesto predecibles.

El Proceso Unificado es un proceso configurable, y como tal es beneficioso para equipos de desarrollo grandes como para equipos pequeños.

Es un proceso centrado en producir sistemas de calidad, de manera repetible y fiable. Mejora la productividad del equipo de desarrollo y lo unifica al brindar las mejores prácticas en el desarrollo de *software* a todos los miembros del equipo.

3.2. Principios del Proceso Unificado

El Proceso Unificado se basa en varios principios considerados como las mejores prácticas del desarrollo de *software*, para producir productos *software* de calidad, los cuales son:

- Desarrollo iterativo del *software*
- Administración de requerimientos
- Uso de arquitecturas basadas en componentes
- Modelado visual del *software*
- Verificación de la calidad del *software*
- Control de cambios

3.2.1. Desarrollo iterativo del *software*

Los proyectos actuales regularmente son complejos y novedosos, en los cuales entre otras cosas, el cliente inicialmente no expone explícitamente todos los requerimientos, y usar un modelo de desarrollo como el desarrollo en cascada no es realista.

En el desarrollo iterativo, el proyecto se planifica en diferentes bloques temporales, llamados iteraciones, que se pueden entender como pequeños proyectos. En cada iteración se evoluciona el producto, a partir de los resultados completados en las iteraciones anteriores, añadiendo nuevos requisitos o mejorando los que ya fueron completados.

Un proceso iterativo, permite una comprensión creciente de los requerimientos, considerando requerimientos cambiantes, a la vez que considera el desarrollo de versiones cada vez más complejas del *software*,

comenzando desde la funcionalidad base del sistema, con las tareas más riesgosas, teniendo así integración progresiva de elementos.

Con el desarrollo iterativo, se logra tener una versión temprana del *software*, a la vez que se minimizan los riesgos del proyecto, pues en cada iteración se van probando los componentes del proceso, verificando si los riesgos previstos son reales, y obteniendo nuevos riesgos, que podrán ser tratados más fácilmente.

Las iteraciones facilitan la reutilización, ya que es más fácil reconocer piezas comunes que se diseñan o desarrollan parcialmente.

En el desarrollo iterativo, en cada iteración se puede ver el avance que se tiene del proyecto, además de poder observar lo que se puede cambiar en la organización y el proceso, para hacer que la siguiente iteración sea realizada de mejor manera.

3.2.2. Administración de requerimientos

Su importancia se centra en obtener la calidad del producto de *software*, al organizar y medir constantemente los requerimientos funcionales y de rendimiento que el cliente plantee, con lo desarrollado.

Para tener controlado un proyecto, es necesario disponer en todo momento de información que permita analizar el proyecto desde distintos puntos de vista, obteniendo así información vital para la correcta toma de decisiones. Para ello la administración de requerimientos está orientada no solo a contar con estándares de documentación de proyectos, sino que aprovechar la información confiable para evaluar con certeza aspectos como la

factibilidad de un requerimiento, el impacto que un cambio propuesto ejercerá en el proyecto, la prioridad del desarrollo asignado, evitar duplicidad, etc.

El Proceso Unificado describe cómo obtener los requerimientos, cómo organizarlos, cómo documentar los requerimientos funcionales y restricciones, cómo rastrear y documentar decisiones, cómo capturar y comunicar requerimientos del negocio, y cómo detectar más fácilmente las inconsistencias.

3.2.3. Uso de arquitecturas basadas en componentes

La arquitectura de un sistema constituye un amplio marco que describe su forma y estructura, sus componentes y cómo encajan juntos-Jerrold Grochow.

Un componente es un elemento *software* con función y límites claros, y se puede integrar en una arquitectura bien definida. Los componentes estándar para un sistema de información son: subsistemas, módulos, pantallas, informes, programas interactivos, programas por lotes, archivos, instrucciones para objetos, etc.

El Proceso Unificado se basa en diseñar una arquitectura base ejecutable. La arquitectura debe ser flexible, fácil de modificar, comprensible, y que promueva la reutilización de componentes. Las actividades de diseño se centran en la arquitectura.

En el desarrollo basado en componentes, se debe definir una arquitectura modular, al identificar, aislar, diseñar, desarrollar y probar componentes bien formados. Para identificar componentes aptos para ser reutilizables, basta con identificar problemas comunes. En la actualidad existen varias industrias que

apoyan el uso de componentes, vendiendo o regalando componentes estándar que pueden ser adaptados a los existentes para el desarrollo.

Con el desarrollo iterativo se identifican los componentes a ser creados, reutilizados o adquiridos por terceros.

Las pruebas se realizan primeramente sobre componentes unitarios, para luego ampliarlas a componentes más grandes e integrados.

3.2.4. Modelado visual del *software*

Un modelo es una simplificación de la realidad, que ayuda a comprender el problema y sus soluciones, comprendiendo más fácilmente los problemas grandes y complejos.

El Proceso Unificado se basa en el UML (Lenguaje de Modelado Unificado) para el modelado visual, que es un lenguaje gráfico para especificar, visualizar, construir, y documentar los artefactos de un sistema orientado al *software*. En él, se dan los medios para escribir modelos del sistema, cubriendo artículos conceptuales tales como procesos del negocio y funciones del sistema, así como artículos concretos tales como clases escritas en un lenguaje de programación específico, esquemas de la base de datos, y componentes de *software* reutilizables.

Con el modelado visual se permite ocultar detalles, mejorar la comunicación entre el equipo de desarrollo al tener un lenguaje estándar, además de permitir analizar la consistencia entre los componentes, el diseño y la implementación.

El Proceso Unificado propone un modelado visual de la estructura y el comportamiento de la arquitectura y los componentes.

3.2.5. Verificación de calidad del *software*

De un sistema no se espera que solamente funcione bien, también es esencial que el rendimiento y la confiabilidad sean los adecuados. Se debe tener claro que la calidad no es responsabilidad de un grupo específico de desarrolladores, sino de cada miembro de la organización del desarrollo.

La verificación de la calidad del *software* se centra en dos áreas: la calidad del producto como tal, y la calidad del proceso para generar dicho producto.

El Proceso Unificado ayuda a planificar, diseñar, implementar, ejecutar y evaluar pruebas que verifiquen la funcionalidad, el rendimiento y la confiabilidad. Las pruebas son vitales para detectar errores antes de la implementación.

3.2.6. Control de cambios

En un proyecto *software* los cambios se vuelven inevitables, para lo cual es necesario evaluar entre otras cosas, si conviene o no realizarlos, y el impacto que tendrán, para evitar problemas.

El Proceso Unificado indica cómo controlar, investigar y monitorear los cambios dentro del proceso iterativo de desarrollo.

3.3. Estructura del Proceso Unificado

La estructura del Proceso Unificado, se puede describir a lo largo de los siguientes dos ejes:

- A. El eje vertical: representa el aspecto estático del proceso, y está basado en cuatro elementos o términos que son: roles, actividades, artefactos, y flujos de trabajo.
- B. El eje horizontal: representa el tiempo, y muestra el aspecto dinámico del proceso, expresado en términos de: ciclos, fases, iteraciones y metas.

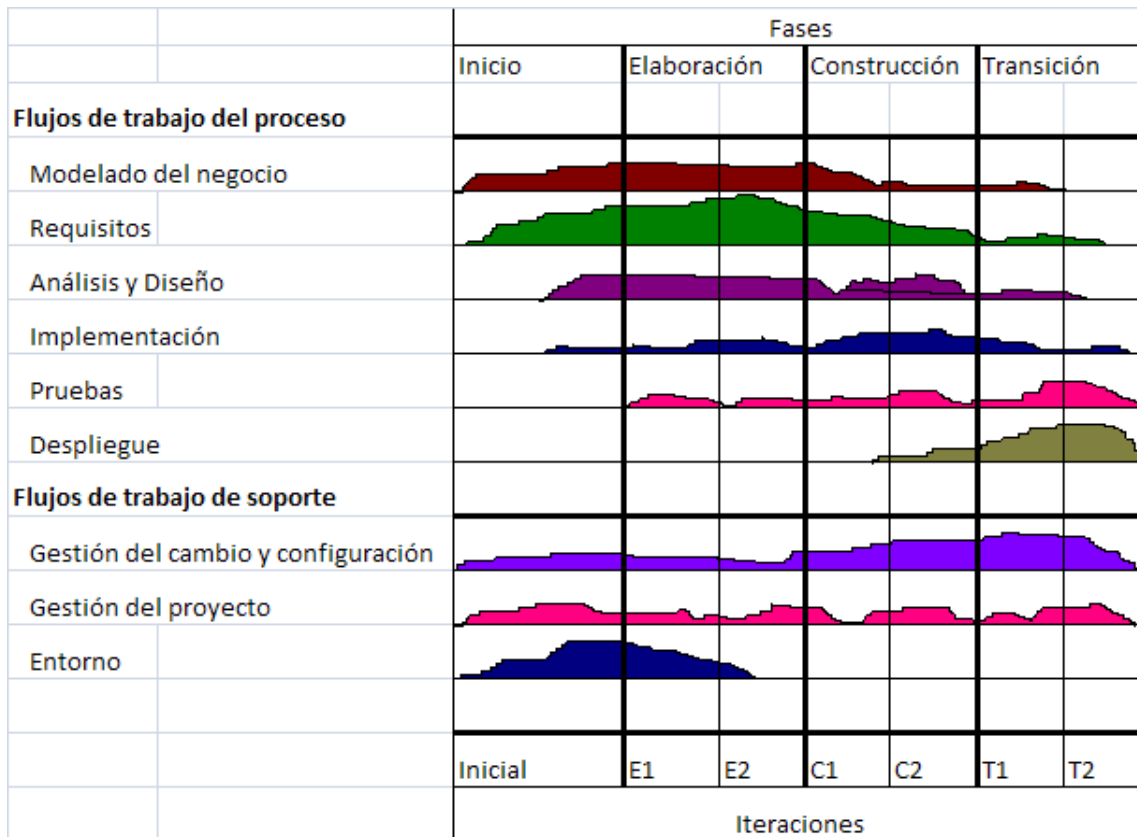
En la figura 2 se muestra la estructura del Proceso Unificado, en la cual aparece una iteración en la fase de inicio, dos iteraciones en la fase de elaboración, dos iteraciones en la fase de construcción y dos iteraciones en la fase de transición, aunque este número de iteraciones en cada una de las fases puede cambiar dependiendo el proyecto.

A continuación se describen los componentes que forman el Proceso Unificado.

3.3.1. Rol

Define el comportamiento y responsabilidades de un individuo, o de un grupo de individuos que trabajan juntos como un equipo. Una persona puede desempeñar distintos roles, y un rol puede ser desempeñado por varias personas. La responsabilidad del rol se centra en llevar a cabo distintas actividades, y ser el dueño de un conjunto de artefactos.

Figura 2. Estructura del Proceso Unificado



Fuente: <http://es.scribd.com/doc/297224/RUP>. Febrero de 2011.

3.3.2. Actividad

Es una unidad tangible de trabajo realizada por un trabajador en un flujo de trabajo, de forma que implica una responsabilidad bien definida para el trabajador, produciendo un resultado bien definido basado en entradas bien definidas, y representa una unidad de trabajo con límites bien definidos.

3.3.3. Artefacto

También llamado producto, es un trozo de información que es producido, modificado o usado en un proceso por los trabajadores al realizar las actividades. Son las cosas o resultados tangibles que se van creando hasta obtener el producto final.

3.3.4. Flujos de trabajo

Con la enumeración de roles, actividades y artefactos no se puede definir el proceso, y por tanto hay que definir la secuencia de actividades realizadas por los diferentes roles, así como la relación entre los mismos, que llevan a obtener resultados observables. Las distintas iteraciones a realizar consistirán en realizar estos flujos de trabajo, con mayor o menor intensidad dependiendo la fase en la que nos encontremos.

En el Proceso Unificado, se definen diferentes flujos de trabajo, también llamados disciplinas, divididos en dos grupos:

A. Flujos de trabajo del proceso:

- Modelado del negocio
- Requerimientos
- Análisis y diseño
- Implementación
- Pruebas
- Despliegue

B. Flujos de trabajo de soporte:

- Administración del proyecto
- Configuración y gestión de cambios
- Entorno

3.3.4.1. Modelado del negocio

Se trata de entender de mejor manera dónde se pondrá en marcha el producto de *software*. Se busca un lenguaje común para el mejor entendimiento entre todos los involucrados. Se pretende asegurar que el producto será de utilidad, y encajará en la organización, además de tener un marco común para los desarrolladores, los clientes y los usuarios finales.

El modelado del negocio se hará dependiendo del tipo de *software* que se esté construyendo.

3.3.4.2. Requerimientos

Es uno de los flujos de trabajo más importantes, al establecer qué es lo que tiene que hacer el sistema exactamente. Aquí los requisitos son el contrato que se debe cumplir, así que debe haber mutua comprensión y aceptación con el usuario final. Se debe establecer el documento de visión, el cual proporciona los requisitos del producto, sus principales características y restricciones.

Se establecen los requerimientos pedidos por el usuario final, tanto los funcionales, como los no funcionales. Además del documento de visión, se establece un documento de glosario con el vocabulario propio del dominio del sistema, los cuales deben estar disponibles en cualquier momento y ser

conocidos por todos los involucrados del sistema, para tener una visión común del mismo.

3.3.4.3. Análisis y diseño

Básicamente se trata de describir qué hace el sistema, lo que se hace en el análisis, y cómo lo hace, que es el diseño. Se trata de traducir los requerimientos, para ser implementados en el sistema.

Se debe contar con la documentación de la arquitectura *software* del sistema, en la cual se identifiquen las vistas necesarias, y cómo se representan. Además se deben establecer divisiones de paquetes y subsistemas.

3.3.4.4. Implementación

Aquí se implementarán los objetos, y definirá un plan para integrarlos. Dicha integración será incremental, realizando pruebas unitarias con los componentes y subsistemas antes de la implementación, y pruebas de implementación después de ésta.

Se debe contar con un modelo de implementación con información general de lo que se implementa, y un apartado por cada iteración, con información de componentes y subsistemas a implementar en dicha iteración, los resultados esperados y las pruebas que serán realizadas sobre los mismos.

3.3.4.5. Pruebas

En este flujo de trabajo, se trata de evaluar la calidad del producto de *software* que se está desarrollando, pero no para aceptarlo o rechazarlo al final del proceso de desarrollo, sino que debe ir integrado en todo el ciclo de vida.

Se establece un plan de prueba global, además de un plan de pruebas por cada iteración, que describe lo que se prueba, el objetivo y cómo se realiza la prueba.

El papel de las pruebas no es asegurar la calidad, pero sí evaluarla, proporcionando una retroalimentación a tiempo, de forma que las cuestiones de calidad puedan resolverse de forma efectiva en tiempo y coste.

3.3.4.6. Despliegue

El objetivo de este flujo de trabajo es producir con éxito las distribuciones del producto, y distribuirlo a los usuarios finales.

Incluye actividades como: prueba al producto en el entorno del usuario, distribuir el *software*, instalar el *software*, proveer ayuda al usuario.

Se desarrolla con mayor intensidad en la fase de transición, ya que cuentan con los mismos objetivos, aunque la ejecución de este flujo de trabajo debe iniciar en fases anteriores, elaborando manuales de usuario, tutoriales, etc.

3.3.4.7. Gestión del cambio y configuración

La finalidad de este flujo de trabajo es mantener la integridad de todos los artefactos que se crean durante el proceso, y el proceso evolutivo que seguirán en el transcurso del ciclo de vida de desarrollo del proyecto.

Cubre tres funciones interdependientes: la gestión de la configuración, la gestión de peticiones de cambio, y las métricas y estatus.

3.3.4.8. Gestión del proyecto

Aquí se establece el plan del desarrollo de *software*, el cual es iterativo, y se planifican las iteraciones a realizar. Se realiza la administración de riesgos, generando un listado de los riesgos posibles. Si se espera vender el sistema, se debe hacer una aproximación de los beneficios a obtener, utilizando un estudio financiero.

Se establece la cantidad de iteraciones que serán realizadas en cada fase del proceso, dependiendo del proyecto a realizar. Se establece la organización del personal, sus roles y actividades que serán realizadas en todo el ciclo de desarrollo. Se debe además monitorizar el proyecto a través de métricas.

3.3.4.9. Entorno

El objetivo de este flujo de trabajo es dar soporte al proyecto con las adecuadas herramientas, procesos y métodos.

Entre sus responsabilidades están: seleccionar y adquirir herramientas *software* para el desarrollo del proyecto, configurarlas para que se ajusten a la

organización, configurar el proceso, mejorar el proceso, dar soporte al entorno durante las iteraciones.

3.3.5. Fases en el ciclo de desarrollo

El Proceso Unificado está dividido en ciclos de desarrollo, en los cuales se obtiene un producto o artefacto al final de cada ciclo. Cada ciclo a su vez, está dividido en cuatro fases que son las siguientes:

3.3.5.1. Inicio

En ésta fase se debe desarrollar el análisis del negocio, para definir la razón de ser del proyecto, justificar la puesta en marcha, y definir el alcance del sistema. Lo importante es hacer lo justo para asegurar que el proyecto sea terminado, explorando el problema antes de decidimos a abordarlo. Se debe crear una idea de la arquitectura que será desarrollada para que soporte el ámbito del sistema.

Entre las principales actividades a desarrollar están: especificación de los criterios de éxito del proyecto, definición de los requerimientos, estimación de los recursos necesarios, planificación inicial de fases y actividades para el resto del proyecto, además de estimar el coste.

3.3.5.2. Elaboración

Aquí ya se tiene una arquitectura propuesta del sistema, se tienen identificados varios riesgos, idealmente se cuenta con un prototipo que muestra la funcionalidad del sistema.

El objetivo de ésta fase es establecer un plan de proyecto justo con una arquitectura completa del sistema. En ésta fase se encontrarán y establecerán prioridades, se estructurarán casi todos los requerimientos funcionales, aunque no sean detallados completamente. Aquí se establece un modelo de análisis sólido, basado en lo establecido en la fase de inicio.

Entre las principales actividades a desarrollar están: análisis del dominio del problema, definición de la arquitectura básica del sistema, análisis de riesgos, planificación del proyecto. Se establecerá una línea base de la arquitectura. Se creará un prototipo del sistema, con un manual de usuario preliminar.

3.3.5.3. Construcción

Se basa en dejar listo el producto de *software*, en una versión inicial funcional, el cual debe poseer la calidad adecuada, y cumplir con los requerimientos. Aquí se detallan todos los requisitos de la fase de elaboración. Se desarrollan y prueban componentes, obteniendo una arquitectura íntegra.

Entre las principales actividades a desarrollar están: análisis, diseño, codificación, pruebas individuales y de integración. Se debe crear el plan de proyecto para la fase de transición. Se debe además poseer un manual preliminar del usuario, que sea lo suficientemente detallado para guiar a los usuarios de la versión inicial.

3.3.5.4. Transición

El objetivo principal es dar a los usuarios interesados, el producto de *software* desarrollado, el cual no necesariamente es perfecto, pero es considerado que puede operar en el entorno del usuario.

Una vez se instale el producto de *software*, el usuario puede descubrir con retraso la necesidad de nuevas características, las cuales deben ser estudiadas por el jefe de proyectos para determinar su impacto en el plan, y que a su vez implicarán nuevos ciclos de desarrollos.

Se realizan pruebas en el entorno del usuario, y las actividades de ésta fase van encaminadas a resolver problemas encontrados en dichas pruebas. Es aconsejable realizar pruebas de regresión, para asegurarse que cambios efectuados al sistema por modificaciones a errores encontrados, no provoquen nuevos errores.

3.3.6. Iteraciones

Una iteración es un mini proyecto, con su propio plan, entregables y evaluaciones, que tiene como resultado una entrega de producto ejecutable, ya sea interno o externo.

Dependiendo la envergadura del proyecto, así será la cantidad de iteraciones que se tendrán, y la distribución de dichas iteraciones en las cuatro fases [inicio, elaboración, construcción, transición]. En la tabla III, se muestra una distribución típica de fases para los proyectos.

Tabla III. **Distribución típica de fases por envergadura del proyecto**

Envergadura del proyecto	Total de iteraciones	Distribución de fases
Poca	3	[0, 1, 1, 1]
Típica	6	[1, 2, 2, 1]
Grande	9	[1, 3, 3, 2]

Fuente: elaboración propia.

La tabla III indica que para un proyecto de poca envergadura, no se tendrán iteraciones en la fase de inicio, se tendrá una iteración en la fase de elaboración, una iteración en la fase de construcción, y una iteración en la fase de transición. Igualmente se pueden leer el conjunto de iteraciones que se tendrán en cada fase dependiente el tipo de proyecto que se desee desarrollar.

4. UTILIZANDO EL PROCESO UNIFICADO COMO MARCO DE REFERENCIA PARA ALCANZAR EL NIVEL 2 DE MADUREZ DEL CMMI

4.1. Nivel 2 de madurez del CMMI

Se trata de repetir éxitos que se dieron anteriormente con otros proyectos que contenían aplicaciones similares, aplicando las disciplinas necesarias para el proceso. En este nivel se establecen los procesos de gestión del proyecto para hacer el correcto seguimiento de coste, de planificación y funcionalidad requeridas.

En este nivel de madurez, los desarrolladores de *software* deben definir actividades tales como el informe del esfuerzo, del tiempo empleado, y el informe de las tareas realizadas.

Este nivel permite la implementación de prácticas mínimas de administración de proyectos, del control de requerimientos, de las versiones del proyecto y de los proyectos que son realizados por subcontratistas. Se trata que el equipo involucrado aproveche la experiencia e inversión realizadas en el proceso de desarrollo de *software* de proyectos anteriores, y puedan aplicarlos al nuevo proyecto.

Aquí se definen los estándares del proyecto del *software*, y la organización se asegura que se sigan correctamente. Los trabajos del proyecto de *software* con sus subcontratistas, si los hay, sirven para establecer una relación fuerte con el cliente.

Al encontrarse en este nivel, no se puede garantizar que todos los proyectos que se desarrollen cuenten con el mismo nivel de madurez, ya que algunos se pueden encontrar aún en el nivel inicial.

4.2. Áreas de proceso del nivel 2 de madurez del CMMI

Existen varias áreas de proceso que se deben cumplir para alcanzar el nivel 2 de madurez del CMMI, llamado Gestionado, en las cuales se deben alcanzar varias metas por cada una de dichas áreas de proceso. Las metas a alcanzar en las áreas de proceso, son planteados a todo el equipo involucrado en el desarrollo, y deben ser cumplidos en todos los niveles organizacionales que apliquen, haciendo notar que si no se cumplen las metas en toda la organización, no se podrá alcanzar este nivel de madurez.

Las áreas de proceso del nivel 2 de madurez del CMMI, son los siguientes:

- Gestión de configuración (CM)
- Medición y análisis (MA)
- Monitorización y control del proyecto (PMC)
- Planificación de proyecto (PP)
- Aseguramiento de calidad de proceso y de producto (PPQA)
- Gestión de requerimientos (REQM)
- Gestión de acuerdos con proveedores (SAM)

4.3. Uso del Proceso Unificado como marco de referencia para alcanzar el nivel 2 de madurez del CMMI

El CMMI describe una trayectoria evolutiva de un inicio caótico de un proceso de desarrollo de *software* no maduro, y su mejora a un proceso maduro y disciplinado.

El CMMI en su representación escalonada, tiene 5 niveles de madurez numerados desde el nivel 1 hasta el nivel 5; cada nivel de madurez desde el nivel 2 en adelante, se compone de áreas de proceso y cada área de proceso identifica un conjunto de características comunes. Cuando están realizadas colectivamente, estas características comunes alcanzan un sistema de metas consideradas importantes para establecer capacidad de proceso en este nivel de madurez.

A continuación se numeran las áreas de proceso del nivel 2 de madurez, dando una descripción del área de proceso, de sus metas específicas, y de las prácticas específicas de cada una de las metas, para luego identificar cómo las características, los métodos, los procedimientos y los artefactos del Proceso Unificado ayudan a alcanzar dichas metas.

4.3.1. Gestión de requerimientos (REQM)

Su propósito es gestionar los requerimientos de los productos y de los componentes del producto del proyecto, e identificar inconsistencias entre esos requerimientos y los planes y productos de trabajo del proyecto.

Los procesos de gestión de requerimientos, gestionan todos los requerimientos recibidos o generados por el proyecto, incluyendo tanto requerimientos técnicos como no técnicos, así como aquellos requerimientos impuestos al proyecto por la organización.

Cuando un proyecto recibe requerimientos de un proveedor de requerimientos aprobado, se revisan éstos con el proveedor de requerimientos para resolver los problemas y para prevenir malentendidos, antes de que los requerimientos se incorporen en los planes del proyecto. Una vez que el proveedor y el receptor de los requerimientos alcanzan un acuerdo, se obtiene un compromiso sobre los requerimientos por parte de los participantes en el proyecto.

4.3.1.1. Metas y prácticas específicas

SG 1 Gestionar los requerimientos: los requerimientos son gestionados y las inconsistencias con los planes y con los productos de trabajo del proyecto son identificadas.

SP 1.1 Comprender los requerimientos: desarrollar una comprensión del significado de los requerimientos con los proveedores de requerimientos.

SP 1.2 Obtener el compromiso sobre los requerimientos: obtener el compromiso de los participantes del proyecto sobre los requerimientos.

SP 1.3 Gestionar los cambios de los requerimientos: gestionar los cambios a los requerimientos a medida que evolucionan durante el proyecto.

SP 1.4 Mantener la trazabilidad bidireccional de los requerimientos: mantener la trazabilidad bidireccional entre los requerimientos y los productos de trabajo.

SP 1.5 Garantizar la alineación entre el trabajo del proyecto y los requerimientos: garantizar que los planes de proyectos y productos de trabajo se mantengan alineados con los requerimientos. Si se encuentran inconsistencias se inician acciones correctivas para resolverlas.

Uno de los principios del Proceso Unificado es ser conducido por casos de uso, que representan un acercamiento sistemático para la reproducción, organización y comunicación de los requerimientos del usuario. Los casos de uso proporcionan la manera de documentar los requerimientos funcionales que sirven como base para el desarrollo del proyecto, las pruebas y la planificación de la iteración. En el Proceso Unificado, los casos de uso se mantienen en un modelo de casos de uso y se refieren constantemente a través del ciclo de vida del proyecto.

Los artefactos del Proceso Unificado que capturan requerimientos en el contexto de la Ingeniería del *software* son:

- Modelo de casos de uso, consistiendo de casos de uso y paquetes de casos de uso.
- Requerimientos complementarios no funcionales
- Revisiones sobre el modelo de casos de uso
- Reporte de casos de uso
- Glosario

Los artefactos del Proceso Unificado que describen los casos de uso y los escenarios (requerimientos) que deben ser desarrollados, según lo utilizado en el contexto de la administración, son:

- Plan de iteración
- Plan de la estructura de la integración
- Plan del proyecto
- Plan de desarrollo del *software*

Todos estos artefactos deben tener una línea base o punto de referencia y estar sujetos a una disciplina (flujo de trabajo) de la administración de cambio.

El Proceso Unificado aboga por el control de la configuración de los productos de *software*, en todos los artefactos de desarrollo, sin embargo los puntos de referencia formales corresponden a los siguientes hitos:

- Hito de los objetivos del ciclo de vida (Fase de inicio)
- Hito de la arquitectura del ciclo de vida (Fase de construcción)
- Hito operacional inicial de la capacidad (Fase de construcción)
- Hito de la entrega del producto (Fase de transición)

Como tal, el Proceso Unificado encaja con el CMMI para el acuerdo en requerimientos, su administración, seguimiento y puntos de referencia.

El paradigma de los casos de uso, es asegurar que las exigencias del usuario se entiendan y capturen adecuadamente. Una vez se capturan los requerimientos, estos deben pasar por varios modelos visuales que el Proceso Unificado incluye, entre los cuales encontramos: casos de uso, diseño, implementación y pruebas, esto para asegurar la consistencia e integridad.

El paradigma del desarrollo iterativo controlado es mitigar los riesgos, porque mientras más temprano sean identificados los riesgos, se plantearán correctamente planes de contingencia para reducirlos, y revisarlos periódicamente. El contar con iteraciones progresivas, que van integrando la funcionalidad requerida, asegura el descubrimiento temprano de riesgos.

Con la identificación temprana de riesgos, el manejo del proyecto se puede redireccionar tempranamente con los requerimientos e incluso realizar cambios tácticos.

Los documentos del Proceso Unificado para la administración de procesos, son los siguientes:

- Caso del negocio
- Plan de desarrollo del *software*
 - Plan de las métricas
 - Lista de riesgos
 - Plan del proyecto
- Plan(es) de iteración(es)
- Evaluación(es) de iteración(es)
- Evaluación(es) de estado

El eficaz control y la administración de los cambios es otra característica del Proceso Unificado, para asegurarse que el *software* esté desarrollado en base a los requerimientos especificados, aceptados y asignados.

El Proceso Unificado influye para que cada proyecto establezca un Comité de Control de Cambios que pueda medir el alcance y el impacto (presupuestario, técnico y de horario) de los cambios o de los defectos propuestos descubiertos durante el curso del proceso de desarrollo. Para asistir a la operación del Comité de Control de Cambios, el Proceso Unificado recomienda el uso de un control fuerte en la administración y configuración de versiones.

4.3.2. Planificación de proyecto (PP)

Su propósito es establecer y mantener planes que definan las actividades del proyecto.

La planificación comienza con los requerimientos que definen el producto y el proyecto, e incluye entre otros: la determinación de recursos necesarios, la negociación de los compromisos, la elaboración de un calendario, la identificación y análisis de los riesgos del proyecto.

El plan del proyecto proporciona la base para realizar y controlar las actividades del proyecto, y usualmente necesitará corregirse a medida que el proyecto progresa, para dar tratamiento a los cambios en los requerimientos y compromisos, las estimaciones inexactas, las acciones correctivas y los cambios del proceso.

4.3.2.1. Metas y prácticas específicas

SG 1 Establecer estimaciones: las estimaciones de los parámetros de planificación del proyecto son establecidas y mantenidas.

SP 1.1 Estimar el alcance del proyecto: establecer una estructura de descomposición del trabajo de alto nivel para estimar el alcance del proyecto.

SP 1.2 Establecer las estimaciones de los atributos del producto de trabajo y de las tareas: establecer y mantener las estimaciones de los atributos de los productos de trabajo y de las tareas.

SP 1.3 Definir las fases del ciclo de vida del proyecto: definir las fases del ciclo de vida del proyecto en las cuales encaje el esfuerzo de la planificación.

SP 1.4 Estimar el esfuerzo y el coste: estimar el esfuerzo y el coste del proyecto para los productos de trabajo y para las tareas, basándose en estimaciones razonables.

SG 2 Desarrollar un plan de proyecto: un plan de proyecto es establecido y mantenido como la base para gestionar el proyecto.

SP 2.1 Establecer el presupuesto y el calendario: establecer y mantener el presupuesto y el calendario del proyecto.

SP 2.2 Identificar los riesgos del proyecto: identificar y analizar los riesgos del proyecto.

SP 2.3 Planificar la gestión de los datos: planificar la gestión de los datos del proyecto.

SP 2.4 Planificar los recursos del proyecto: planificar los recursos necesarios para ejecutar el proyecto.

SP 2.5 Planificar el conocimiento y las habilidades necesarias: planificar las necesidades de conocimiento y habilidades para ejecutar el proyecto.

SP 2.6 Planificar la involucración de las partes interesadas: planificar la involucración de las partes interesadas identificadas.

SP 2.7 Establecer el plan del proyecto: establecer y mantener el contenido del plan global del proyecto.

SG 3 Obtener el compromiso con el plan: los compromisos con el plan del proyecto son establecidos y mantenidos.

SP 3.1 Revisar los planes que afectan al proyecto: revisar todos los planes que afectan al proyecto para comprender los compromisos del proyecto.

SP 3.2 Reconciliar los niveles de trabajo y de recurso: ajustar el plan de proyecto para reconciliar los recursos disponibles y los estimados.

SP 3.3 Obtener el compromiso con el plan: obtener el compromiso de las partes interesadas relevantes, responsables de ejecutar y de dar soporte a la ejecución del plan.

Una de las metas del Proceso Unificado es asegurarse que las expectativas de todos los involucrados estén sincronizadas y consistentes. Esto se asegura con evaluaciones periódicas a través del ciclo de vida del proyecto, y se documenta en el informe de evaluación de estado del proyecto. El informe contiene los siguientes datos: uso de recursos (personal y financiero), riesgos, progreso técnico según lo indicado con las métricas y los resultados importantes del hito.

El Proceso Unificado hace uso de las siguientes clases de métricas:

- Progreso: medido en líneas del código, número de clases, puntos de función por iteración, renovación.
- Estabilidad: tipo de renovación, la volatilidad de requerimientos o implementación.
- Adaptabilidad: costo de renovación.
- Modularidad: grado del impacto de renovación.
- Calidad: índice del descubrimiento de defectos, densidad, profundidad de la herencia, indicador de renovación.
- Madurez: horas de prueba por falla.
- Perfiles del gasto en los recursos: previstos contra actuales.

El Proceso Unificado hace uso de varios documentos, los cuales deben ser constantemente verificados, ya que representan una fuente de información vital para todos los involucrados en el proyecto.

Los documentos usados por el Proceso Unificado son:

- Caso del negocio
- Plan de desarrollo del *software*
 - Plan de las métricas
 - Lista de riesgos
 - Plan del proyecto
- Plan(es) de iteración(es)
- Evaluación(es) de iteración(es)
- Evaluación(es) de estado

4.3.3. Monitorización y control del proyecto (PMC)

El propósito de la monitorización y control del proyecto (PMC) es proporcionar una comprensión del progreso del proyecto, para que se puedan tomar las acciones correctivas apropiadas, cuando el rendimiento del proyecto se desvíe significativamente del plan.

Un plan documentado de proyecto es la base para la monitorización de las actividades, la comunicación del estado y la toma de acciones correctivas.

Cuando el rendimiento se desvía significativamente del plan, una visibilidad apropiada permite las acciones correctivas. Una desviación es significativa si, cuando se deja sin resolver, impide al proyecto cumplir sus objetivos.

4.3.3.1. Metas y prácticas específicas

SG 1 Monitorizar el proyecto frente al plan: el progreso y el rendimiento real del proyecto son monitorizados frente al plan del proyecto.

SP 1.1 Monitorizar los parámetros de planificación del proyecto: monitorizar los valores reales de los parámetros de planificación del proyecto frente al plan del proyecto.

SP 1.2 Monitorizar los compromisos: monitorizar los compromisos frente a aquellos identificados en el plan de proyecto.

SP 1.3 Monitorizar los riesgos del proyecto: monitorizar los riesgos frente a aquellos identificados en el plan de proyecto.

SP 1.4 Monitorizar la gestión de datos: monitorizar la gestión de los datos del proyecto frente al plan de proyecto.

SP 1.5 Monitorizar la involucración de las partes interesadas: monitorizar la involucración de las partes interesadas frente al plan de proyecto.

SP 1.6 Llevar a cabo revisiones de progreso: revisar periódicamente el progreso, el rendimiento y los problemas del proyecto.

SP 1.7 Llevar a cabo revisiones de hitos: revisar los logros y los resultados del proyecto en los hitos seleccionados del proyecto.

SG 2 Gestionar las acciones correctivas hasta su cierre: las acciones correctivas son gestionadas hasta su cierre, cuando el rendimiento o los resultados del proyecto se desvían significativamente del plan.

SP 2.1 Analizar los problemas: recoger y analizar los problemas y determinar las acciones correctivas necesarias para tratarlos.

SP 2.2 Llevar a cabo las acciones correctivas: llevar a cabo acciones correctivas sobre los problemas identificados.

SP 2.3 Gestionar las acciones correctivas: gestionar las acciones correctivas hasta su cierre.

El Proceso Unificado tiene varios niveles de planificación del proyecto, y proporciona un informe del estado en que se encuentra lo planeado contra el funcionamiento real. Este informe, generado para los hitos específicos, es responsabilidad del encargado del proyecto.

Los hitos importantes del Proceso Unificado corresponden al final de una fase (inicio, elaboración, construcción o transición). Las oportunidades de revisión existen en los hitos de menor importancia en el final de cada iteración dentro de una fase, y sirven como los puntos y lecciones de decisión aprendidos para las direcciones futuras. Por ejemplo, las metas de la fase de elaboración son: analizar el dominio del problema, establecer una arquitectura robusta, desarrollar el plan del proyecto, y eliminar los riesgos que generan mayor impacto en el proyecto.

Las decisiones de arquitectura se deben tomar con una comprensión completa del sistema. Esto implica que la mayoría de los casos de uso serán descritos, considerando algunos requerimientos suplementarios.

En la fase final de elaboración, los objetivos y alcances detallados del sistema se evalúan, así como la arquitectura propuesta y la resolución de riesgos importantes.

Las acciones correctivas se definen y manejan cuando los resultados y el funcionamiento reales se desvían significativamente de los planes del *software*.

La lista de riesgos es un artefacto del Proceso Unificado que proporciona una descripción de todos los riesgos identificados del proyecto, y sirve como retroalimentación para el seguimiento y planeamiento del proyecto. Cada riesgo se describe en términos de su impacto y de un plan de contingencia sugerido para reducirlo. La lista de riesgos se desarrolla junto con el caso del negocio para tomar o no tomar acciones en el proyecto. La lista de riesgos se mantiene a través del ciclo de vida del proyecto.

El proceso iterativo controlado del desarrollo, según lo descrito en el Proceso Unificado, se asegura que los involucrados están conscientes del progreso del proyecto y de cualquier cambio que sea necesario realizar al proyecto. Los cambios propuestos son registrados por un Comité de Control de Cambio, para asegurarse de que son realistas y pueden ser acomodados en el cronograma total del proyecto, y en la estructura del mismo.

4.3.4. Gestión de Configuración (CM)

Su propósito es establecer y mantener la integridad de los productos de trabajo utilizando la identificación de configuración, el control de configuración, el registro del estado de configuración y las auditorías de configuración.

4.3.4.1. Metas y prácticas específicas

SG 1 Establecer líneas base: las líneas base de los productos de trabajo identificados son establecidas.

SP 1.1 Identificar elementos de configuración: identificar los elementos de configuración, los componentes y los productos de trabajo relacionados que serán puestos bajo gestión de configuración.

SP 1.2 Establecer un sistema de gestión de configuración: establecer y mantener un sistema de gestión de configuración y de gestión del cambio para controlar los productos de trabajo.

SP 1.3 Crear o liberar líneas base: crear o liberar las líneas base para uso interno y para la entrega al cliente.

SG 2 Seguir y controlar los cambios: los cambios a los productos de trabajo bajo gestión de configuración son seguidos y controlados.

SP 2.1 Seguir la petición de cambio: seguir las peticiones de cambio para los elementos de configuración.

SP 2.2 Controlar los elementos de configuración: controlar los cambios a los elementos de configuración.

SG 3 Establecer la integridad: a integridad de las líneas base es establecida y mantenida.

SP 3.1 Establecer registros de gestión de configuración: establecer y mantener los registros que describen los elementos de configuración.

SP 3.2 Realizar auditorías de configuración: realizar auditorías de configuración para mantener la integridad de las líneas base de configuración.

Según lo descrito en el Proceso Unificado, la administración fuente de la configuración es un elemento esencial en el método iterativo controlado del desarrollo. Puesto que el *software* se desarrolla en etapas; es vital que las versiones del *software*, del esfuerzo precedente del desarrollo, estén disponibles para el desarrollo subsiguiente. Planear el trabajo de *software* que debe ser producido en cada etapa, esta es la base del Proceso Unificado.

El Proceso Unificado tiene dos instrumentos importantes para definir cómo los activos del desarrollo del *software* de un proyecto deben ser administrados, y cómo deben ser integrados:

- El plan de la administración de la configuración, y
- El plan de construcción de la integración.

El plan de la administración de la configuración, que comienza en la fase de inicio, describe lo siguiente:

- Administración y dirección del versionamiento del software.
- Modelos de proceso unificados de desarrollo divididos en artefactos de la configuración.
- Cambios y puestas en marcha usando métodos de control del cambio.

El plan de construcción de la integración proporciona los detalles de los artefactos de la configuración que deben ser construidos, y el orden en que son integrados en una iteración dada.

El plan de la administración de la configuración del Proceso Unificado, involucra una descripción del control de configuración y administración del proceso, para asegurarse que los productos de trabajo sean identificados, controlados, y estén disponibles.

El mantenimiento y administración de los cambios debe ser llevado en un Comité de control de cambios, y debe haber un adecuado sistema de administración de los cambios, que involucre los cambios dados, su costo, cambios requeridos y su seguimiento.

El Proceso Unificado de desarrollo aboga a los requerimientos, diseño e implementación de las líneas base, y el seguimiento entre ellas, con un mantenimiento en formato electrónico. Cualquier cambio en las líneas base son arbitrarios en los diferentes niveles del control del proyecto. El diseño y la implementación de los cambios deben ser revisados a un nivel técnico.

La aprobación y el control de los cambios deben ser comunicados a todo nivel y deben estar descritos en el plan de administración de la configuración y en el plan de desarrollo de *software*.

4.3.5. Gestión de acuerdos con proveedores (SAM)

El propósito de la gestión de acuerdos con proveedores, es gestionar la adquisición de productos y/o servicios de los proveedores.

El alcance de esta área de proceso se refiere a la adquisición de productos, servicios y componentes de productos y servicios, que pueden ser entregados a los clientes del proyecto o incluidos en un sistema de producto o servicio.

4.3.5.1. Metas y prácticas específicas

SG 1 Establecer los acuerdos con el proveedor: los acuerdos con los proveedores se establecen y mantienen.

SP 1.1 Determinar el tipo de adquisición: determinar el tipo de adquisición para cada producto o componente del producto a adquirirse.

SP 1.2 Seleccionar los proveedores: seleccionar los proveedores en base a una evaluación de su capacidad para cumplir los requerimientos especificados y los criterios establecidos.

SP 1.3 Establecer los acuerdos con el proveedor: establecer y mantener los acuerdos con el proveedor.

SG 2 Satisfacer los acuerdos del proveedor: los acuerdos con los proveedores deben satisfacer tanto al proyecto como al proveedor.

SP 2.1 Realizar el acuerdo del proveedor: desarrollar las actividades tal y como se especifican en el acuerdo suscrito con el proveedor.

SP 2.2 Aceptar el producto adquirido: asegúrese de que el acuerdo con el proveedor se cumple antes de aceptar el producto adquirido.

SP 2.3 Garantizar la transición de los productos: garantizar la transición de los productos adquiridos por el proveedor.

Estas metas van más allá del alcance del Proceso Unificado y son dependientes de la organización.

Mientras se subcontrata no se trata específicamente en el Proceso Unificado, sus herramientas, técnicas y mecanismos se asumen bajo los subcontratistas de modo que el proceso siga siendo homogéneo.

Todas las decisiones de subcontratación se deben documentar en el caso del negocio. Subcontratistas que están siguiendo el mismo plan de desarrollo que el contratista fundamental, también participarían en los mismos intercambios técnicos, hitos principales, y evaluaciones de estado.

4.3.6. Aseguramiento de la calidad de proceso y de producto (PPQA)

El propósito del aseguramiento de la calidad de proceso y de producto, es proporcionar al personal y a la gerencia una visión objetiva de los procesos y de los productos de trabajo asociados, durante toda la vida del proyecto.

4.3.6.1. Metas y prácticas específicas

SG 1 Evaluar objetivamente los procesos y los productos de trabajo: el cumplimiento por parte de los procesos ejecutados, los productos de trabajo y los servicios asociados a las descripciones de proceso, estándares y procedimientos aplicables es evaluado objetivamente.

SP 1.1 Evaluar objetivamente los procesos: evaluar objetivamente los procesos ejecutados seleccionados, frente a las descripciones de proceso, estándares y procedimientos aplicables.

SP 1.2 Evaluar objetivamente los productos de trabajo: evaluar objetivamente los productos de trabajo seleccionados, frente a las descripciones de procesos, estándares y procedimientos aplicables.

SG 2 Proporcionar una visión objetiva: las no conformidades son objetivamente seguidas y comunicadas, y su resolución es asegurada.

SP 2.1 Comunicar y resolver las no conformidades: comunicar problemas de calidad y asegurar la resolución de las no conformidades con el personal y con los gerentes.

SP 2.2 Establecer registros: establecer y mantener registros de las actividades de aseguramiento de la calidad.

El Proceso Unificado considera que la calidad es responsabilidad colectiva de todo el personal del proyecto y no viene incorporada a cualquier organización dada por sí misma.

Las tareas de planeamiento de control de calidad del *software*, son una responsabilidad organizacional, sin embargo, el Proceso Unificado tiene un número de cualidades que se prestan a formar un programa de control de calidad eficaz del proyecto.

Cada hito del Proceso Unificado tiene criterios específicos del proyecto, que pueden servir como base para las revisiones. Cada actividad dentro del Proceso Unificado tiene una revisión por separado. Cada revisión está asociada con un conjunto de puntos de chequeo que representan una barrera que debe ser pasada antes que se llegue a la siguiente actividad.

El Proceso Unificado proporciona la dirección a tomar para repasar los artefactos dados. Por ejemplo, los resultados del análisis de objetos a medida que se desarrollan por las necesidades del diseñador, son revisados por un arquitecto independiente, diseñador, diseñador de caso de uso, y revisor de diseño. Dado el Proceso Unificado definido y la revisión del artefacto, el cuerpo objetivo concerniente a la calidad del producto debe fácilmente poder determinar el cumplimiento de los estándares y normas del desarrollo.

Para la evaluación, el Proceso Unificado proporciona las listas de chequeo, documentos y las plantillas necesarias que se pueden aplicar como estándares del proyecto.

Una de las metas del Proceso Unificado es asegurar que las expectativas de todos los involucrados estén sincronizadas y consistentes. Aparte de cualquier entrada de resultados dados por la intervención de control de calidad, el Proceso Unificado invita a la divulgación sobre recursos (humanos y financieros), riesgos, progreso técnico según lo interpretado con las métricas, y los resultados importantes del hito.

El programa del Proceso Unificado a través de las métricas proporciona pautas en el conjunto de métricas siguientes:

- Progreso: medido en líneas del código, número de clases, puntos de función por iteración, renovación.
- Estabilidad: tipo de renovación, la volatilidad de requerimientos o implementación.
- Adaptabilidad: costo de renovación
- Modularidad: grado del impacto de renovación
- Calidad: índice del descubrimiento de defectos, densidad, profundidad de la herencia, indicadores de renovación.
- Madurez: horas de prueba por falla
- Perfiles del gasto en los recursos: previstos contra actuales

El proceso de control de cambios descrito en el Proceso Unificado, permitirá un mecanismo por el que documentar las desviaciones y errores, y proporcionar la resolución.

4.3.7. Medición y análisis (MA)

Su propósito es desarrollar y mantener una capacidad de medición utilizada para apoyar las necesidades de gestión de la información.

Los datos tomados para la medición deben estar alineados con los objetivos de la organización, para proporcionar información útil a la misma. Se mide para tener datos, que se deben analizar y utilizar para tomar buenas decisiones.

4.3.7.1. Metas y prácticas específicas

SG 1 Alinear las actividades de medición y análisis: los objetivos y actividades de medición están alineados con las necesidades de información y los objetivos identificados.

SP 1.1 Establecer los objetivos de medición: establecer y mantener los objetivos de medición que se derivan de las necesidades de información y los objetivos identificados.

SP 1.2 Especificar medidas: especificar las medidas para tratar los objetivos de medición.

SP 1.3 Especificar los procedimientos de recolección y almacenamiento de datos: especificar cómo se obtendrán y almacenarán los datos de la medición.

SP 1.4 Especificar los procedimientos de análisis: especificar cómo se analizarán e informarán los datos de medición.

SG 2 Proporcionar los resultados de la medición: los resultados de la medición que tratan las necesidades de información y los objetivos identificados son proporcionados.

SP 2.1 Obtener los datos de la medición: obtener los datos de la medición especificados.

SP 2.2 Analizar los datos de la medición: analizar e interpretar los datos de la medición.

SP 2.3 Almacenar los datos y los resultados: gestionar y almacenar los datos de la medición, especificaciones de la medición y resultados del análisis.

SP 2.4 Comunicar los resultados: comunicar los resultados de las actividades de medición y análisis a todas las partes interesadas relevantes.

En los flujos de trabajo de soporte, principalmente la gestión del proyecto, el Proceso Unificado apoya la medición y análisis, requiriendo métricas que serán evaluadas por la gerencia, quien toma las decisiones importantes del proyecto.

En la medición y análisis, el Proceso Unificado debe ser mejorado para incluir los detalles de la presentación de informes y análisis.

5. ACTIVIDADES DE UNA ORGANIZACIÓN GUATEMALTECA PARA ALCANZAR EL NIVEL 2 DE MADUREZ DEL CMMI

En el presente capítulo se presenta un caso de estudio sobre las diferentes actividades que ha seguido una organización guatemalteca, para alcanzar el nivel 2 de madurez en desarrollo de *software* del modelo CMMI, el cual tiene como objetivo brindar a la comunidad desarrolladora de *software*, un marco de referencia para poder realizar o no actividades similares, para alcanzar un mejor nivel de madurez en desarrollo de *software*.

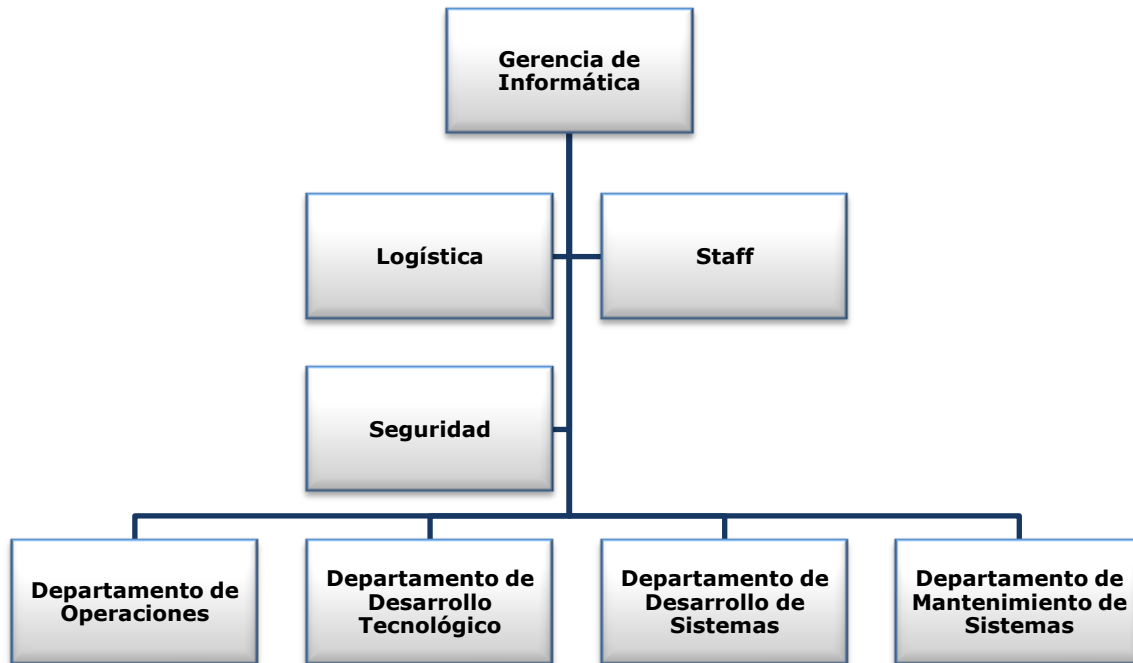
La organización aún no ha alcanzado el nivel 2 de madurez, pero lleva bastante recorrido.

5.1. La organización

El caso de estudio es una organización guatemalteca de gobierno, la cual posee un departamento de informática llamado Gerencia de Informática, que tiene al inicio del proceso de mejora la estructura organizacional presentada en la figura 3.

La Gerencia de Informática está compuesta aproximadamente por 146 colaboradores, y es la encargada de la tecnología de la organización, dando mantenimiento a la plataforma informática, desarrollando y manteniendo las aplicaciones que apoyan las actividades de la organización, dando asesoría en la compra de equipo y asesoría en la compra de soluciones ya implementadas.

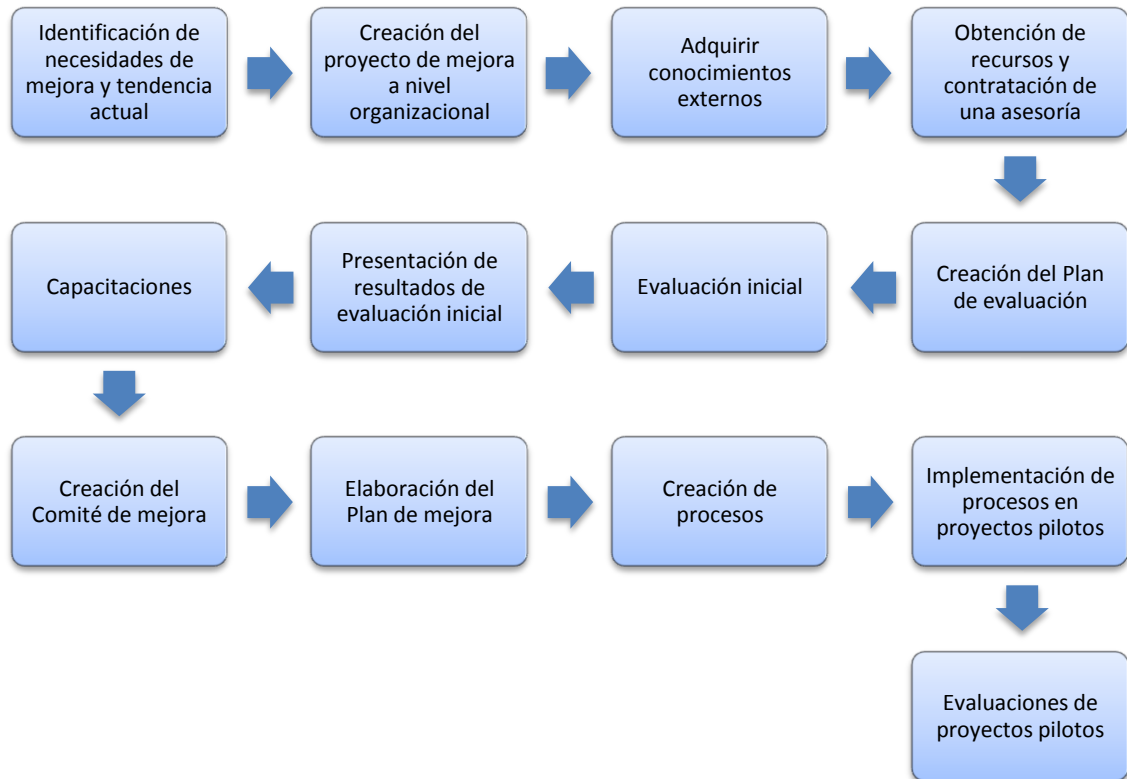
Figura 3. Organigrama de la Gerencia de Informática



Fuente: elaboración propia.

En la figura 4, se presenta el conjunto de actividades para la mejora de desarrollo de *software* que se han desarrollado en la organización, tratando de presentarlas cronológicamente, aunque muchas se desarrollaron en paralelo. Dichas actividades serán descritas en los siguientes subcapítulos.

Figura 4. **Actividades de mejora de la organización**



Fuente: elaboración propia.

5.2. **Identificación de necesidades de mejora y tendencia actual**

Un grupo de personas del departamento de Desarrollo de sistemas, buscando mayor calidad en el *software* que se desarrolla y mantiene, identifica la tendencia actual de mejora en desarrollo de *software* utilizando el modelo CMMI. Luego de identificado el modelo CMMI, se comenta con las autoridades, obteniendo su apoyo con la idea de implementarlo.

En esta actividad se resalta la proactividad que todo departamento de informática debe tener, observando las tendencias actuales en su especialidad.

5.3. Creación del proyecto de mejora a nivel organizacional

El departamento de Desarrollo de sistemas propone el proyecto de mejora a la Gerencia de informática, quien toma la sugerencia y lo agrega como un proyecto de la Gerencia de informática en el Plan organizacional anual de la organización.

El proyecto organizacional toma nombre de: Implementación de procesos en la Gerencia de Informática en el marco del estándar CMMI, que contiene: descripción, propósito, alcances, objetivos, indicadores, fuentes de financiamiento, ubicación, beneficios, responsables, aspectos legales, programación general, análisis costo beneficio, recursos.

Con lo anterior se logra el compromiso de la Gerencia de informática por dicho proyecto, el conocimiento de la alta gerencia en la organización, la asignación de recursos necesarios, y se afinan aspectos no tomados en cuenta desde el inicio.

Desde el inicio del proyecto, el mismo ha ido cambiando año con año, por los conocimientos adquiridos, ya que al inicio era muy optimista buscando la certificación en el nivel 2 de madurez del CMMI, pero se ha ido afinando para acoplarse a los objetivos de la organización.

5.4. Adquirir conocimientos externos

Se realizaron reuniones con diferentes empresas, o sus representantes en Guatemala, que han acompañado a organizaciones en conseguir la certificación en el modelo CMMI para el desarrollo de *software*, obteniendo conocimiento

significativo de las necesidades propias de la organización, para alcanzar su mejora de procesos en desarrollo de *software*.

Se logra obtener poco conocimiento de otras organizaciones que están siguiendo el mismo camino de mejora, en cuanto a cómo lo están haciendo, pues existe celo en dar información de sus actividades de mejora, especialmente en organizaciones privadas.

Se observa que la tendencia en empresas guatemaltecas dedicadas a la fabricación de *software*, es obtener la certificación del CMMI a través del seguimiento con una empresa externa experta en el tema. Tomando como referencia lo anterior, luego de un análisis y experiencias previas, se opta por buscar una empresa experta para apoyar el proceso de mejora en la organización.

5.5. Obtención de recursos y contratación de una asesoría

Colocado el proyecto de mejora en el Plan organizacional anual de la organización, se asignan los recursos necesarios, tanto humanos como financieros, para poder cumplir satisfactoriamente el proyecto, siendo la contratación de una asesoría un requisito indispensable.

Por ser una organización de gobierno y debido al monto aproximado de la contratación, se debió realizar un Concurso público, según el Reglamento de contrataciones del estado, el cual se llamó: Servicio de Asesoría para la Evaluación de los Procesos bajo el estándar CMMI. En el Concurso público se ponen las bases, descripciones de la contratación, etc.

Previo a la contratación de la asesoría, se recibieron varias ofertas, contratándose de acuerdo a lo dispuesto en la legislación correspondiente: Ley de contrataciones del estado, Reglamento de la ley de contrataciones del estado, Guatecompras y el reglamento interno de la organización.

El principal problema presentado en la contratación de la asesoría, fue el tiempo que tardó la elaboración de bases y contratación, y el convencer a los diferentes departamentos involucrados en la organización, los beneficios que se persiguen.

5.6. Creación del Plan de evaluación

Una vez notificada la empresa asesora ganadora del Concurso público, esta se contactó con la Gerencia de informática para iniciar la ejecución de la asesoría, de acuerdo a lo dispuesto en el contrato.

La empresa asesora contratada es guatemalteca, y los consultores expertos en CMMI que trabajaron son de nacionalidad mexicana, panameña y colombiana, observándose una oportunidad para crear o aumentar expertos guatemaltecos en CMMI.

Los primeros acercamientos fueron mediante comunicación a distancia, usando correo electrónico y mensajería instantánea, trabajándose en la creación del documento Plan de evaluación, en el cual se recopila información del negocio de la organización, y se detallan los objetivos, principales participantes, alcance, entregables, calendario, y restricciones de la evaluación inicial, que se realizaría a la organización.

5.7. Evaluación inicial

Los modelos CMMI tienen su método de evaluación llamado SCAMPI (*Standard CMMI Appraisal Method for Process Improvement*).

Las evaluaciones mediante SCAMPI pueden ser de tipo A, B o C. La evaluación inicial a la organización fue mediante SCAMPI tipo C, la cual se caracteriza por ser de corta duración y alcance, y se utiliza para ver el uso de los procesos en la organización y las iniciativas de mejora con relación al modelo CMMI. Al ser muy breve, los resultados permiten identificar tendencias en el uso de procesos, no así un nivel de madurez.

Basado en el Plan de evaluación se realiza la evaluación inicial, que comenzó con la Presentación Inicial, la cual se dio a los evaluados, que contenía información de la empresa asesora, del modelo CMMI, de los objetivos y alcance de la evaluación, de la metodología para la evaluación y el calendario de actividades.

La evaluación inicial se basó en entrevistas, y en revisiones documentales puntuales, tardó una semana y fue realizada por una persona experta.

5.8. Presentación de resultados de evaluación inicial

Como resultado de la evaluación inicial se obtuvo la Presentación de resultados, la cual es una presentación ejecutiva conteniendo: objetivos y alcances de la evaluación, antecedentes, metodología utilizada, nivel de cumplimiento y/o apego al CMMI, principales fortalezas, principales debilidades, principales riesgos y recomendaciones, además de información adicional a considerar en la planificación de las acciones de mejora.

La parte principal de la Presentación de resultados es el nivel de cumplimiento y/o apego al modelo CMMI en la organización, para lo cual se cuenta con un semáforo el cual significa:

Rojo: la evidencia presentada para la cobertura de la práctica se considera insuficiente de acuerdo al modelo.

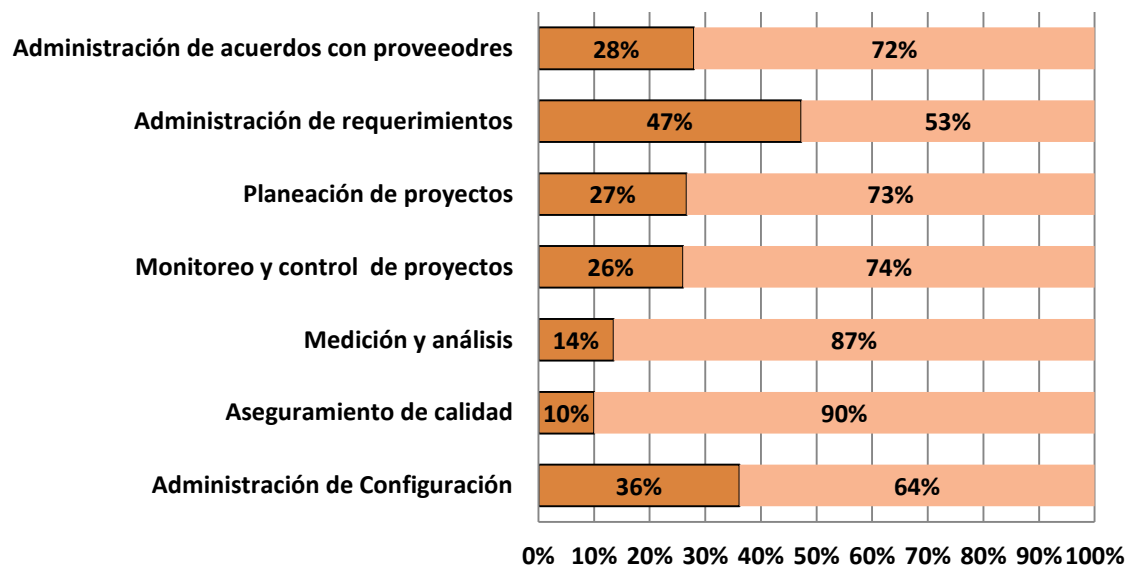
Amarillo: la evidencia presentada muestra un avance sobre la implementación de la práctica y el cumplimiento de las metas establecidas por el modelo, siendo insuficientes para lograr la cobertura total de las metas.

Verde: la evidencia presentada muestra que los esfuerzos por cubrir la práctica se consideran suficientes y adecuados.

Como resultado de la evaluación inicial, el evaluador asignó un color del semáforo a cada práctica específica de las metas específicas de las áreas de proceso evaluadas. Los colores asignados a las prácticas específicas sirven para dar un color a la meta específica asociada. Los colores asignados a las metas específicas sirven para dar un color al área de proceso asociada.

Para identificar el cumplimiento y/o apego de cada área de proceso en la organización, se convierten los colores en porcentajes, obteniendo como resultado lo mostrado en la figura 5 siguiente.

Figura 5. **Porcentaje de avance de madurez del nivel 2 del CMMI**



Fuente: elaboración propia.

Los resultados obtenidos de la gráfica anterior, muestran que en el área de proceso Aseguramiento de calidad, existe un 10% de cobertura en la organización con relación a lo requerido en el modelo CMMI, y de igual manera se interpretan los resultados para cada área de proceso.

El documento presentación de resultados, contiene por cada área de proceso, las oportunidades de mejora encontradas, el informe de apego al modelo mediante los colores del semáforo, por cada práctica específica y meta específica.

El documento presentación de resultados contiene además un resumen global del porcentaje de avance, los siguientes pasos para la mejora, el plan de trabajo propuesto, la cadena de valor identificada, las áreas de oportunidad encontradas, y las principales líneas de acción recomendadas.

5.9. Capacitaciones

Como resultado de la evaluación inicial, y como siguiente paso de la misma, se impartieron capacitaciones del modelo a seguir, para lo cual se realizó una capacitación a nivel ejecutivo, y otra a nivel técnico.

La capacitación ejecutiva tuvo una duración de 20 horas hábiles, y la impartió un consultor para aproximadamente 10 participantes, incluido el gerente, sub-gerente y jefes de área de la Gerencia de informática.

La capacitación ejecutiva estuvo orientada a los beneficios que la organización obtendrá al implementar el modelo CMMI, y como resultado se obtiene mayor conocimiento del modelo CMMI, del proceso de mejora y un aumento considerable en el apoyo al mismo.

Luego de la capacitación ejecutiva se dio la capacitación a nivel técnico con nombre Introducción al modelo CMMI, la cual duró 40 horas, repartidas en 5 días, impartida por un consultor a un grupo de 10 participantes. Para la asistencia a la capacitación a nivel técnico, se seleccionó personal de la Gerencia de informática que tiene conocimientos de procesos, y que está comprometido con la mejora.

La capacitación a nivel técnico fue magistral, con ejercicios y un examen final. En ella se estudiaron todas las áreas de proceso del modelo CMMI, y todas sus prácticas específicas, no limitándose al nivel 2 de madurez del CMMI.

5.10. Creación del Comité de mejora

El Comité de mejora se planificó desde el inicio del proceso, y en contrato se especifica un apoyo de la organización para lograr los objetivos planteados.

La asignación de las personas para asistir a la capacitación a nivel técnico, fue basada en las personas que formarían el Comité de mejora, que se formó luego de finalizadas las capacitaciones.

El Comité de mejora se integró basado en las áreas de proceso del nivel 2 del CMMI, y lo integraron 8 personas de la siguiente manera: 2 para requerimientos, 1 para planificación y monitoreo de proyectos, 1 para medición y análisis y aseguramiento de la calidad, 2 para el proceso de administración de proveedores, 2 para administración de la configuración.

El Comité de mejora ha desempeñado un papel importante en el proceso de mejora, pues es el contacto principal entre la empresa consultora y la organización, para la creación o modificación de los procesos, depurándolos y aprobándolos, y luego de su aprobación es el soporte para resolver dudas y también se encarga del seguimiento, auditorías de proceso, acciones correctivas, creación y presentación de informes, etc.

Las sesiones de trabajo entre integrantes del Comité de mejora y la empresa consultora han sido mediante comunicación a distancia y sesiones presenciales de una hora, un día, pero en general una visita ha durado entre 3 y 5 días.

En contrato con la empresa asesora, se indica que acompañaría a la organización a implementar un área de proceso del nivel 2 de madurez del CMMI, aunque hubo apoyo en todas las áreas de proceso, siendo principalmente el Comité de mejora quien decidió el área de proceso a implementar, con base en los resultados de la evaluación inicial y un análisis de la situación en la organización.

5.11. Elaboración del Plan de mejora

A partir de la evaluación inicial, la empresa asesora elaboró el documento Plan de mejora. Es un plan que se va actualizando, pero quedó más definido al formar el Comité de mejora, pues ellos aportaron información relevante al mismo.

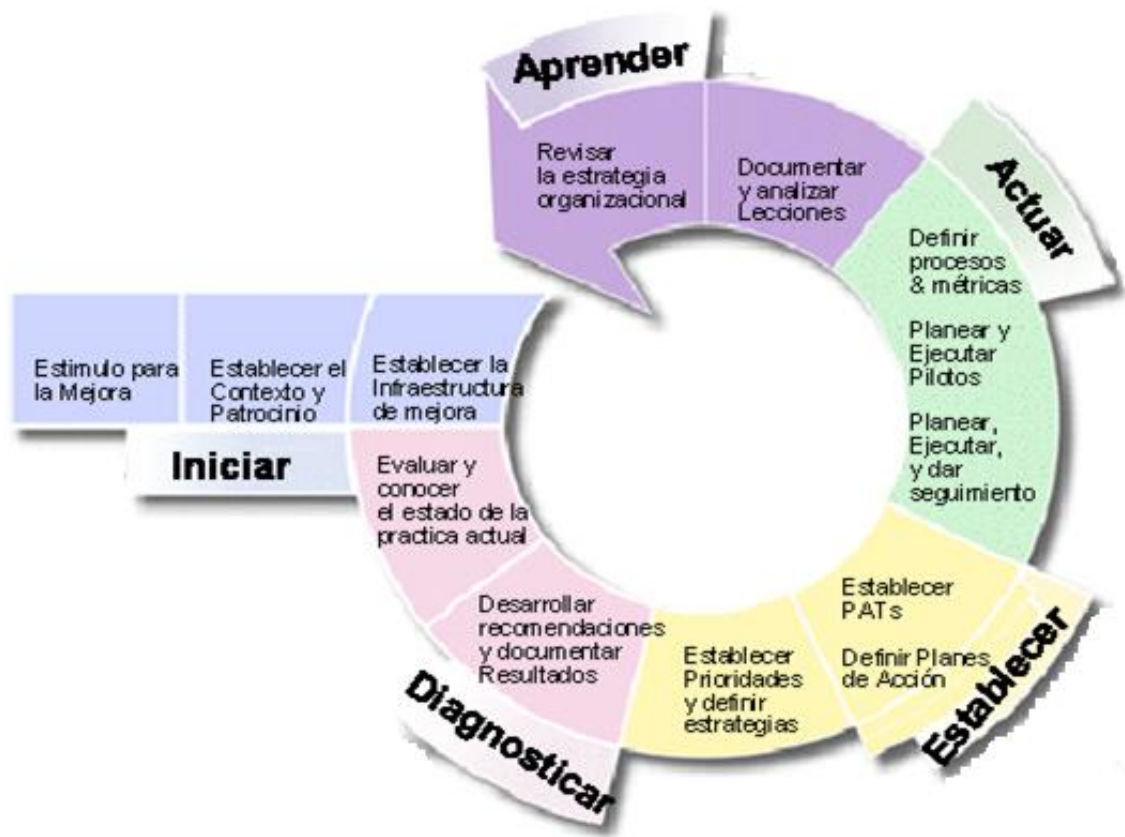
El Plan de mejora contiene: introducción, situación actual del proyecto, objetivos de negocio, el diagnóstico inicial realizado, objetivos de la organización, objetivos de mejora, beneficios esperados con la mejora, supuestos y restricciones, estructura organizacional, roles y responsabilidades, capacitación requerida para el proyecto, desarrollo del proyecto de mejora, recursos humanos, facilidades e instalaciones y acrónimos.

La parte principal del Plan de mejora es el desarrollo del proyecto de mejora, el cual tiene como estrategia general llevar a cabo un ciclo de mejora del modelo IDEAL, que se utilizó para diseñar la estrategia a seguir para el cumplimiento de los objetivos de la mejora de procesos.

El modelo IDEAL es una marca registrada del SEI para guiar el inicio, planificación e implementación de iniciativas de mejora para el proceso de *software* en las organizaciones, aunque su uso se ha ampliado a otras áreas.

El modelo IDEAL está compuesto por 5 etapas, y varias actividades por cada etapa. En la figura 6 se muestra la estructura del modelo IDEAL, la cual es una traducción del inglés del documento fuente. En dicha figura, PAT es acrónimos de *process action teams*.

Figura 6. **Modelo IDEAL**



Fuente: SEI, IDEAL, <http://www.sei.cmu.edu/reports/96hb001.pdf>. Mayo de 2011.

Las actividades realizadas en la organización, para cada una de las cinco etapas del modelo IDEAL para la mejora del proceso *software*, son las siguientes:

Iniciar: se identificó en establecer un plan de mejora basado en un modelo de mejores prácticas reconocido a nivel mundial.

Diagnosticar: se realizó una evaluación para identificar las fortalezas y áreas de oportunidad de la organización para determinar los esfuerzos requeridos y factibilidad para la implementación del modelo CMMI.

Establecer: basados en los resultados de la evaluación inicial se desarrollan los procesos faltantes y se adecuaron los existentes, con el fin de dar cumplimiento a las metas establecidas por el modelo CMMI.

La empresa asesora en conjunto con el personal del Comité de mejora crea los procesos faltantes y documentos de los mismos, adecúa los procesos ya existentes con su respectiva documentación.

Actuar: los procesos creados o actualizados se dieron a conocer a la organización, con la comunicación y capacitación requerida conforme las necesidades detectadas.

La empresa asesora dio soporte a la organización para la ejecución de los procesos y utilización de los activos definidos para tal efecto.

Se ejecutaron auditorías de aseguramiento de calidad durante el desarrollo de los proyectos para revisar su correcta ejecución y se hagan

las adaptaciones pertinentes conforme a las necesidades específicas de los proyectos.

Aprender: al terminar los proyectos se ejecutaron lecciones aprendidas para recopilar las mejores prácticas ejecutadas en un proyecto y detectar las que puedan enriquecer el proceso de la organización, o aquellas en las que el área de calidad deberá poner atención.

5.12. Definición de roles y responsabilidades

Dentro del Plan de mejora se definen los roles y sus responsabilidades, que por su importancia se muestran en la tabla IV.

Tabla IV. **Roles y responsabilidades para el plan de mejora**

ROL	RESPONSABILIDAD
Patrocinador	<ul style="list-style-type: none"> • Participa en las reuniones de seguimiento de avance. • Apoyo en la resolución de problemáticas reportadas que afecten la implementación del modelo CMMI nivel 2.
Líder del proyecto de mejora	<ul style="list-style-type: none"> • Revisar el cumplimiento del Plan de mejora de procesos. • Apoyar al Grupo Consultor en el seguimiento de asuntos y resolución de problemas. • Asignar a los responsables de tareas e involucramiento de los grupos de interés.

Continúa tabla IV.

<p>Líder del proyecto de mejora</p>	<ul style="list-style-type: none"> • Participar en reuniones periódicas con el patrocinador. • Coordinar el equipo de trabajo responsable de sacar adelante el proyecto de acuerdo a los planes. • Participar en las reuniones de seguimiento.
<p>Comité de Procesos</p>	<ul style="list-style-type: none"> • Participar en las reuniones de revisión y aprobación de procesos. • Apoyar en la definición y ajuste de los procesos. • Validar y asegurar que los procesos desarrollados se encuentren alineados a los procesos de negocio. • Apoyo en la toma de decisiones globales que impactan en todo el proyecto. • Ser agentes de cambio dentro de la organización.
<p>Responsables de procesos</p>	<ul style="list-style-type: none"> • Asegurar que los procesos desarrollados se encuentren alineados a los procesos de negocio. • Apoyar en la definición o ajuste de los procesos. • Apoyar en la toma de decisiones globales que impactan en todo el proyecto. • Responsable del repositorio de procesos, administrando su seguridad, estableciendo la estructura de documentos y asegurando sus respaldos. • Ejecutar el conjunto de mejoras al proceso.

Continúa tabla IV.

Grupo de aseguramiento de calidad	<ul style="list-style-type: none">• Participación como responsable del área de Proceso.• Revisar periódicamente la implementación del proceso en los proyectos.• Revisar los productos de trabajo generados en los proyectos.• Participar en el Comité de Procesos.
-----------------------------------	--

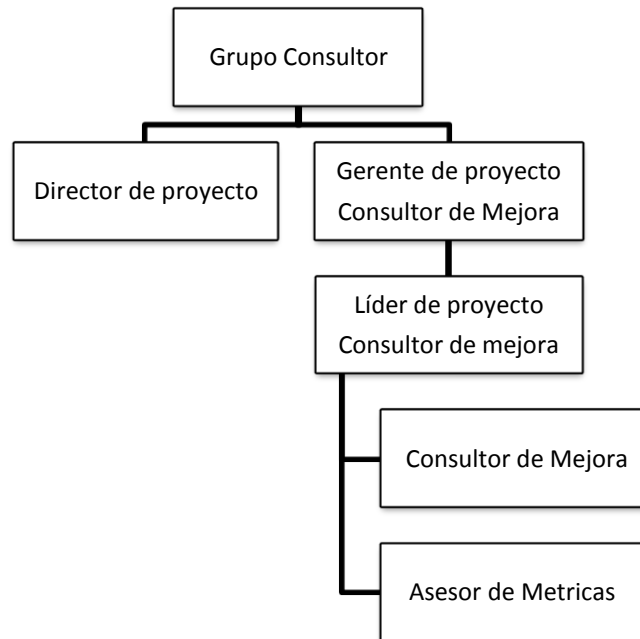
Fuente: elaboración propia.

Todas las personas involucradas en la organización, recibieron una presentación inicial del proceso de mejora, en la cual se explicó el proceso a seguir, y la importancia de ellos en la mejora, específicamente desempeñando su rol dentro del mismo.

El rol de patrocinador fue desempeñado por el subgerente de la Gerencia de informática, el rol de líder del proyecto de mejora fue desempeñado por una persona de la Gerencia de informática, el rol de comité de procesos fue desarrollado por el Comité de mejora (6 personas), el rol responsable de procesos fue desempeñado por varias personas del Comité de mejora, que tenían a su cargo áreas de proceso del nivel 2 del CMMI, y el rol del grupo de aseguramiento de calidad fue desempeñado por dos personas.

El Grupo consultor presentó la estructura organizacional que se muestra en la figura 7, basada en lo definido en contrato, y en los resultados de la evaluación inicial.

Figura 7. **Organigrama de la empresa consultora**



Fuente: elaboración propia.

5.13. Creación de procesos

En la etapa Establecer del modelo IDEAL identificado, la empresa asesora con apoyo del Comité de mejora, genera los procesos que cuentan con los elementos indispensables de procesos adecuados: gente, proceso y tecnología.

Los procesos generados y documentados son: planeación de proyectos, monitoreo y control de proyectos, administración de requerimientos, administración de configuración, medición y análisis, administración de acuerdos con proveedores y aseguramiento de calidad de productos y procesos.

Luego de creados y documentados los procesos, la empresa asesora dio capacitación de los mismos mediante talleres, para que el personal de apoyo en los proyectos pilotos, conozca los objetivos, actividades, roles y responsabilidades dentro de dichos procesos definidos.

5.14. Implementación de procesos en proyectos pilotos

En la etapa Actuar del modelo IDEAL identificado, se generó un documento con los proyectos por implementar en la Gerencia de informática, seleccionando cuatro representativos de los tipos de proyecto de la organización, para asegurar la efectividad y beneficios del modelo de mejora.

Los cuatro proyectos seleccionados se nombraron pilotos, los cuales se iniciaron en paralelo, con una duración aproximada de tres meses para ser piloteados. La empresa asesora dio soporte a los proyectos pilotos para asegurar la correcta implementación de las prácticas en los proyectos, de todas las áreas de proceso del nivel 2 del CMMI, principalmente del área de proceso que la organización seleccionó para alcanzar.

El Grupo consulto genera el documento Plan de implementación, que contiene: información del documento, introducción, alcance, situación actual, selección de proyectos pilotos, roles y responsabilidades, y cronograma de actividades de consultoría.

Por parte del Grupo consultor se da apoyo en todas las áreas de proceso del nivel 2 de madurez del CMMI, pero se documentan únicamente las áreas de proceso que se decide pilotear según Contrato público, que al final fueron las siguientes dos: Planeación de proyectos y Monitoreo y control de proyectos.

En esta etapa se realizaron revisiones quincenales para verificar el avance y cumplimiento de los procesos en los proyectos pilotos, para descubrir fortalezas y debilidades detectadas en el proceso. Se generaron reportes mensuales del desempeño en los proyectos piloto.

Los criterios de éxito de la implementación, son los siguientes:

- A. Que el proceso sea conocido por el personal de la Gerencia de informática mediante talleres de proceso.
- B. Que el proceso sea utilizado por los proyectos pilotos y se detecten oportunidades de mejora.
- C. Que el proceso permita cumplir adecuadamente los objetivos del proyecto.
- D. Que el proceso colabore para identificar desviaciones en el proyecto y controlar los esfuerzos.
- E. Que el proceso esté adecuadamente documentado, sea claro, esté completo y la audiencia lo pueda ejecutar.

5.15. Evaluaciones de proyectos pilotos

Los proyectos pilotos fueron evaluados dos veces, mediante SCAMPI tipo B, pero dichas evaluaciones no fueron oficiales, pues en Contrato público no se especificó de esa manera, para evitar que el costo del contrato se elevara.

Ambas evaluaciones fueron realizadas con un mes de diferencia, y la finalidad de la primera evaluación es identificar problemas en implementación de los procesos, y poder corregirlos a tiempo, para que en segunda evaluación dichos inconvenientes ya estuvieran resueltos.

Al final de las actividades de seguimiento a los proyectos pilotos, se realizó la segunda evaluación de avance, y se evaluaron todas las áreas de proceso del nivel 2 de madurez del CMMI, dando énfasis a las dos áreas de proceso seleccionadas por la organización, relacionadas a la administración de proyectos, implementadas en los 4 proyectos pilotos.

Los resultados obtenidos de la segunda evaluación, fueron presentados a los participantes de los pilotos y a los jefes, en ellos se destaca el alcance según la segunda evaluación a los proyectos pilotos, de las áreas de proceso Planeación de proyectos y Monitoreo y control de proyectos, que corresponde al logro del objetivo planteado en el Contrato público.

5.16. Siguietes actividades

A la fecha, se está realizando el plan de trabajo que incluye las acciones correctivas para realizar los ajustes necesarios de acuerdo con los hallazgos detectados en los proyectos pilotos, para todas las áreas de proceso.

Entre las siguientes actividades, también se destaca el seguir implementando y madurando los procesos en proyectos posteriores.

CONCLUSIONES

1. Los principales factores que influyen en la calidad del *software* son: la gente, el proceso y la tecnología, y actualmente gran cantidad de esfuerzos en mejorar la calidad del *software* van orientados en mejorar el proceso.
2. A las empresas que desarrollan *software*, con base al estudio realizado en el presente trabajo, una opción para alcanzar el nivel 2 de madurez en desarrollo de *software* basado en el modelo CMMI, es la utilización del Proceso Unificado.
3. El Proceso Unificado apoya en alcanzar las metas requeridas para las áreas de proceso del nivel 2 de madurez del modelo CMMI, con excepción del área de proceso Gestión de acuerdos con proveedores.
4. Para implementar el modelo CMMI puede existir resistencia al cambio, principalmente por el control y cantidad de documentos requeridos, para lo cual se debe iniciar con los documentos más importantes, unificando si es posible información de varios documentos, e incluyendo al personal de la organización activamente en la definición de los procesos.
5. Para iniciar un proyecto de mejora en desarrollo de *software*, se debe tener en cuenta el apoyo de una empresa experta en el tema, para que el tiempo de implementación sea menor y exista mayor probabilidad de éxito, además de contar con el apoyo activo de la dirección de la organización.

6. En un proceso de mejora utilizando el modelo CMMI, no debe ser indispensable una evaluación oficial o certificación, sino alinear el proceso de mejora con los objetivos del negocio.

RECOMENDACIONES

1. Para mejorar la calidad del producto de *software* final, a las empresas que desarrollan *software*, mejorar sus procesos de desarrollo de *software* mediante estándares como el Proceso Unificado.
2. Para que una empresa u organización que desarrolla *software* pueda alcanzar el nivel 2 de madurez en desarrollo de *software* basado en el modelo CMMI, con base a las investigaciones realizadas en el presente trabajo, puede utilizar el Proceso Unificado.
3. A las empresas o departamentos de informática que deseen implementar procesos de mejora continua, para mejorar la calidad en sus productos de *software*, tener de referencia un modelo reconocido internacionalmente como el modelo CMMI.
4. A las empresas dedicadas al desarrollo de *software*, certificarse en el modelo CMMI, para tener reconocimiento y mayor competitividad a nivel nacional e internacional, por la experiencia obtenida en empresas ya certificadas.
5. A las empresas guatemaltecas dedicadas a la asesoría en desarrollo de *software*, tener personal capacitado en el modelo CMMI, para que en Guatemala se aproveche de mejor manera este mercado de mejora.

6. A los jefes de departamentos informáticos, apoyar activamente los procesos de mejora en desarrollo de *software*, por los múltiples beneficios que se pueden obtener.
7. A las empresas o departamentos informáticos que han iniciado o desean iniciar un proyecto de mejora en procesos de desarrollo de *software*, apoyarse con una empresa experta en el tema, pues aunque puede ser costoso se verán los beneficios en el tiempo de implementación.
8. A los departamentos informáticos que han iniciado o desean iniciar un proyecto de mejora, alinear dicho proyecto con los objetivos de la organización.
9. A las empresas o departamentos informáticos que han iniciado o desean iniciar un proyecto de mejora de desarrollo de *software*, para vencer la potencial resistencia al cambio, conformar un grupo de mejora que participe activamente en todo el proceso, y brindarle la ayuda necesaria.

BIBLIOGRAFÍA

1. BETH, Mary; KONRAD, Mike; SHRUM, Sandy. *CMMI Guía para la integración de procesos y la mejora de productos*. Cátedra de Mejora de Procesos de Software en el Espacio Iberoamericano (trad.). 2a ed. Madrid: Pearson Educación, 2009. Disponible en Web: <<http://www.sei.cmu.edu/library/assets/cmmi-dev-v12-spanish2.pdf>> [Consulta: 21 de junio de 2011].
2. JACOBSON, Ivar; BOOCH, Grady; RUMBAUGH, James. *El proceso unificado de desarrollo de software*. Madrid: Pearson Educación, 2000. 464 p. ISBN: 84-7829-036-2.
3. PFLEEGER, Shari Lawrence. *Ingeniería de software, teoría y práctica*. Quiroga, Elvira (trad.). Buenos Aires: Pearson Education, 2002. 792 p. ISBN: 987-9460-71-5.
4. PRESSMAN, Roger S. *Ingeniería del software, un enfoque práctico*. 4a ed. Madrid: McGRAW-HILL, 1998. 581 p. ISBN: 84-481-1186-9.
5. Software Engineering Institute; *CMMI for Development, Version 1.3* [en línea]. Noviembre 2010. Disponible en Web: <<http://www.sei.cmu.edu/reports/10tr033.pdf>> [Consulta: 21 de febrero del 2011].