



Conference Paper

Structural Analysis of CMMI with the Software Development Process, Project Portfolio Management and PMBOK

Gloria Arcos-Medina¹, Alejandra Oñate-Andino², and Danilo Pástor²¹Escuela Superior Politécnica de Chimborazo, Grupo de Investigación en Ingeniería de Software²Escuela Superior Politécnica de Chimborazo, Facultad de Informática y Electrónica

Abstract

CMMI is one of the most used models in the software industry to certify the quality of its processes and products, reduces or eliminates rework, increases product and process reuse, reduces costs due to multiple evaluations and process improvement programs. On the other hand, companies that develop software apply traditional models or agile models, which manage the software development process. They also apply development standards such as ISO/IEC 12207 as well as project portfolio management standards and project management standards. The purpose of this paper is to relate the traditional and agile software development processes, project portfolio management and PMBOK with the CMMI standard. The results of this work allow us to determine the aspects that are missing from these standards and the best practices to reach a high level of CMMI maturity. It is concluded that for a company it is important to apply both agile processes and traditional processes, as well as all the standards that contribute improving the management of its projects. None of these practices and norms are opposed to the CMMI process areas, but rather are complemented to achieve a high level of quality in the final product and in achieving the objectives of the company.

Keywords: Software Development, Agile Practices, Management Project, PMBOK, CMMI, Project Portfolio.

1. Introducción

Las Tecnologías de la Información (TI) han dejado de ser una herramienta de soporte para convertirse en algo totalmente necesario para cualquier empresa. Hoy en día es impensable concebir una empresa que no use las tecnologías de la información para la gestión del día a día. Sin embargo, el gestionar estas tecnologías de la información se ha convertido en una tarea compleja que presenta varios problemas, principalmente relacionados a cómo lograr que las TI no sean vistas como un gasto necesario sino

Corresponding Author:

Gloria Arcos-Medina
garcos@esPOCH.edu.ec

Received: 28 July 2017

Accepted: 5 September 2017

Published: 30 January 2018

Publishing services provided by
Knowledge E

© Gloria Arcos-Medina et al. This article is distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use and redistribution provided that the original author and source are credited.

Selection and Peer-review under the responsibility of the SIIPRIN Conference Committee.



como una inversión cuyo principal beneficio sea la ventaja competitiva que estas tributan a la organización. Con el propósito de disminuir esta problemática, se han creado diversos marcos de trabajo y mejores prácticas que se han convertido en estándares de la industria: ISO 12207, PMBOK, Capability Maturity Model Integration (CMMI), modelos ágiles como SCRUM, entre otros; que son motivo de estudio en este paper; tal es así que su implantación se ha convertido en los últimos años en una necesidad para aquellas empresas que deseen gestionar las TI adecuadamente y a través de ellas, lograr ventajas de negocio.

De acuerdo con [1], en el año 2012 el 18% de los proyectos de software fracasan, el 43% tiene problemas de planificación, funcionalidad o expectativas de los usuarios y únicamente el 39% tiene éxito; si se considera el porcentaje total de los proyectos que han tenido problemas en su desarrollo, éste llega al 61%; estos datos permiten determinar la tendencia mínima que existe en la actualidad a reducir los proyectos no exitosos, desde el 2004 al 2012, la reducción ha sido del 10%. Una de las razones para obtener este nivel de fracaso, es la mala gestión de proyectos TI.

Para poder resolver esta temática es importante conocer el porqué del fracaso de estos proyectos, Coello [2], menciona los siguientes problemas: a) Falta de compromiso y apoyo de la alta dirección, b) Toma de requerimientos y definición de alcance equivocado o incompleto, c) Carencia de un sistema de control de cambios, d) Jefes de proyecto improvisados, etc.

Desde el año 1987 aparece CMM que en 2000 evolucionó a CMMI, Integración de Modelos de Madurez de Capacidades, que es un modelo para la mejora y evaluación de procesos para el desarrollo, mantenimiento y operación de sistemas de software [3]. En la actualidad, las ventas de la industria del software se han incrementado, en los últimos años se ha registrado un crecimiento en el mundo desde el 2004 del 11% convirtiéndose en un mercado potencial [4]; sin embargo de acuerdo a la publicación CMMI® SCAMPI SM.Class A Appraisal Results. 2012 Mid-Year [5], Asia y Norte América son los continentes que lideran el número de certificaciones de calidad CMMI seguidos por Europa, lo que nos hace pensar que el software producido en Sur América en su gran parte no está sujeto a evaluación de calidad, por lo que pierden competitividad frente a otros países del mundo.

Por otra parte, el proceso de desarrollo de software está dividido en dos grandes grupos: modelos tradicionales de desarrollo de software o también llamados procesos guiados por un plan y los modelos ágiles que surgieron en febrero de 2001. Cada modelo implica una serie de etapas y tareas que se ejecutan entre la idea inicial de

desarrollo de software y la entrega del producto final. La mayoría de las organizaciones de TI necesitan lo mejor de ambos mundos el software que soporta trabajo en constante cambio de las necesidades del negocio, y un proceso para la entrega de software que es predecible, entrenable, y auditable [6]. Cualquiera que sea el modelo aplicado es importante garantizar la calidad del proceso de software.

Uno de los principales estándares que guían el ciclo de vida del desarrollo de software es el ISO/IEC Standard 12207:2008, que es un marco de referencia que contiene los procesos, las actividades y las tareas involucradas en el desarrollo, explotación y mantenimiento de un producto software, abarcando la vida del sistema desde la definición de requisitos hasta que se deja de utilizar [7]. Todo producto de software implica la realización de un proyecto que debe ser priorizado a través de un portafolio de proyectos que a su vez debe ser correctamente gestionado a través de estándares como PMBOK.

El presente trabajo tiene como objetivo, relacionar la estructura de CMMI con el proceso de desarrollo de software tradicional y ágil, con Project Portfolio Management (PPM) y Project Management Body of Knowledge (PMBOK), de tal forma que, independientemente del modelo que apliquen las empresas para desarrollar software, se consideren los aspectos principales para garantizar la calidad del software desarrollado, de tal forma que el producto pueda ser fácilmente certificado.

El contenido de este artículo que se presenta a continuación está dividido en 6 secciones: la Sección 2 describe los aspectos relacionados a la gestión de portafolios de proyectos y PMBOK, la Sección 3 describe las fases de capacidad y madurez de CMMI, la Sección 4 describen los Modelos de desarrollo de software, la Sección 5 presenta la relación de CMMI con los modelos de desarrollo, PPM y PMBOK y la Sección 6 presenta las conclusiones del trabajo.

2. CMMI. Capability Maturity Model Integration

Existen varias referencias en la industria que enfocan de forma directa o indirecta el tema de la gestión de procesos de la organización: por ejemplo, si la empresa tiene un sistema de calidad ISO 9001, hay una mejora continua de procesos instaurada y funcionando; si la gestión de servicios está basada en ITIL, sabemos que los procesos de gestión de dichos servicios están basados en unas mejores prácticas, etc. [8]

Las siglas de CMMI responden a Capability Maturity Model Integration, que significa Integración de Modelos de Madurez de las Capacidades, es un conjunto de modelos basados en las mejores prácticas en la gestión de los procesos, desarrollados a través

de un proyecto conjunto en el que participaron el SEI (Software Engineering Institute), el gobierno estadounidense y algunos miembros de la industria. Dicho modelo establece cinco niveles de ‘madurez’ de las organizaciones en función de si tienen o no una serie de características que detalla cada modelo. Las organizaciones pueden ser evaluadas y, en función de dicha evaluación, se las puede otorgar un nivel de madurez del 1 al 5. Es decir, a través de CMMI, podemos saber el grado de ‘madurez’ de los procesos que tiene una organización, de acuerdo a un modelo de buenas prácticas.

De acuerdo con [9], *Capability* es empresa esperados que pueden ser alcanzados siguiendo un proceso, *Maturity* es el grado de propiedad, de mejora de los procesos por medio de niveles, *Model*, provee asistencia para desarrollo de procesos, no son descripciones de procesos, *Integration* significa alcance, expectativa de establecer todos los procesos usando CMMI es la combinación de 3 modelos: SW-CMM Software, SE-CMM Ingeniería de Software y IPD-CMM Desarrollo integrado de productos. En síntesis CMMI es un modelo de evaluación de los procesos de una organización. En ingeniería de software y el desarrollo organizacional es un enfoque de mejora de procesos.

Uno de los principales aspectos de CMMI es la clasificación de las organizaciones en maduras e inmaduras y, luego, la prescripción del camino a seguir por una organización inmadura para evolucionar y convertirse en una organización madura. Una organización inmadura es aquella que lleva adelante sus proyectos sin una definición previa de los procesos a seguir, mientras que las organizaciones maduras el modelo entiende a aquellas que desarrollan sus proyectos en forma planeada.

CMMI propone 5 niveles de madurez, los mismos que se indican en la Figura 1.

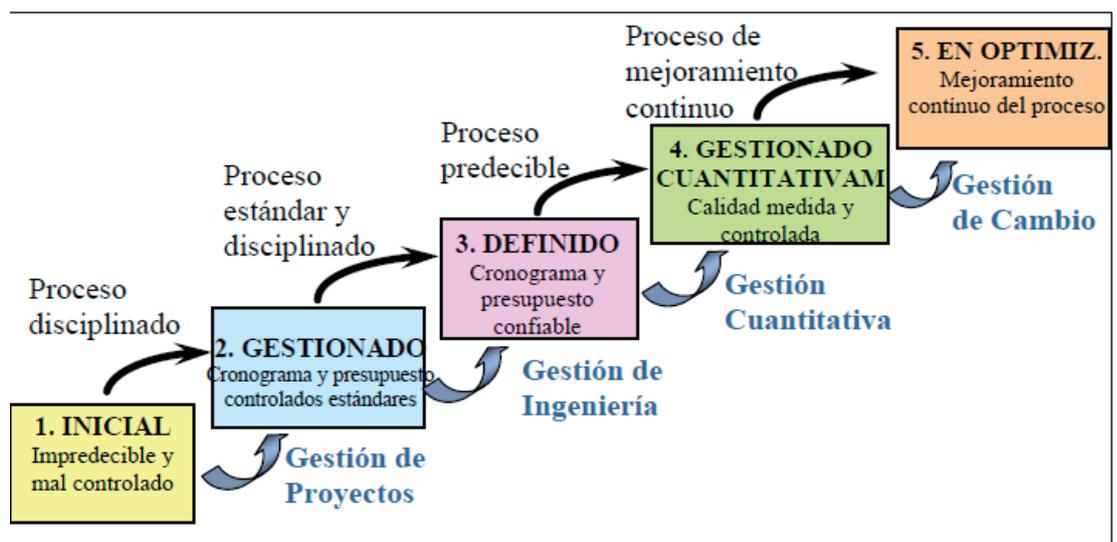


Figura 1: Niveles de Madurez CMMI [10].

El modelo propone además 22 áreas de proceso que se organizan en 4 categorías: *Ingeniería, Administración de proyectos, Administración de procesos y Soporte*. Cada área de proceso consta de uno a tres objetivos concretos y de los tres objetivos genéricos. Para cada objetivo, se esperan normalmente varios procedimientos para alcanzar ese objetivo. La Tabla 1, muestra la agrupación de áreas de proceso.

TABLA 1: Áreas de procesos CMMI [11].

Nivel de madurez	Áreas de procesos
5	CAR: Análisis de causas y resolución OPM: Administración del rendimiento de las organizaciones
4	OPP: Rendimiento de los procesos organizativos QPM: Administración cuantitativa de proyectos
3	RD - desarrollo de requisitos TS: Solución técnica PI: Integración de productos VER: Comprobación VA: Validación IPM: Administración integrada de proyectos RSKM: Administración de riesgos OPF: Enfoque de los procesos organizativos OPD: Definición de procesos organizativos OT: Aprendizaje organizativo DAR: Análisis de decisiones y resolución Control de calidad de los procesos y productos
2	CM: Administración de configuraciones MA: Medición y análisis SAM: Administración de acuerdos con proveedores PP: Planeación de proyectos PMC: Supervisión y control de proyectos RM: Administración de requisitos
1	No hay áreas de proceso dentro del nivel 1 del modelo. Este representa un proceso indefinido sin capacidad de definir un proceso o de repetir un resultado con la descripción del proceso que lo generó. Técnicamente, en una valoración según CMMI, una organización que no cumpla los objetivos de las áreas de proceso en el nivel 2 del modelo está todavía en el nivel 1.

Para cada área de proceso, se puede evaluar un nivel de capacidad. Se definen cuatro niveles de capacidad en CMMI: 0. Incompleto, 1. Realizado, 2. Administrado y 3. Definido.

3. Proceso de Desarrollo de Software

El proceso de desarrollo de software puede ser analizado desde dos puntos de vista: el proceso de desarrollo tradicional, representado por el estándar 12207 y el proceso de desarrollo ágil, que se describen a continuación.

3.1. Estándar ISO/IEC 12207

Esta norma establece un marco de referencia común para los procesos del ciclo de vida del software, con una terminología bien definida a la que puede hacer referencia la industria del software. Contiene procesos, actividades y tareas para aplicar durante

la adquisición de un sistema que contiene software, un producto software puro o un servicio software, y durante el suministro, desarrollo, operación y mantenimiento de productos software. Esta norma incluye también un proceso que puede emplearse para definir, controlar y mejorar los procesos del ciclo de vida del software.

La ISO 12207 define un **modelo de ciclo de vida** como un marco de referencia que contiene los procesos, actividades y tareas involucradas en el desarrollo, operación y mantenimiento de un producto software, y que abarca toda la vida del sistema, desde la definición de sus requisitos hasta el final del uso [12].

Esta norma agrupa las actividades que pueden llevarse a cabo durante el ciclo de vida del software en cinco procesos principales, ocho procesos de apoyo y cuatro procesos organizativos. Cada proceso del ciclo de vida está dividido en un conjunto de actividades; cada actividad se subdivide a su vez en un conjunto de tareas como se indica en la Figura 2.

Los **procesos principales del ciclo de vida**, son cinco procesos que dan servicio a las partes principales durante el ciclo de vida del software. Los procesos **de apoyo al ciclo de vida**, son procesos que apoyan a otros procesos como parte esencial de los mismos, con un propósito bien definido, y contribuyen al éxito y calidad del proyecto software. Un proceso de apoyo se emplea y ejecuta por otro proceso según sus necesidades. Los **procesos organizativos del ciclo de vida**, se emplean por una organización para establecer e implementar una infraestructura construida por procesos y personal asociado al ciclo de vida, y para mejorar continuamente esta estructura y procesos.

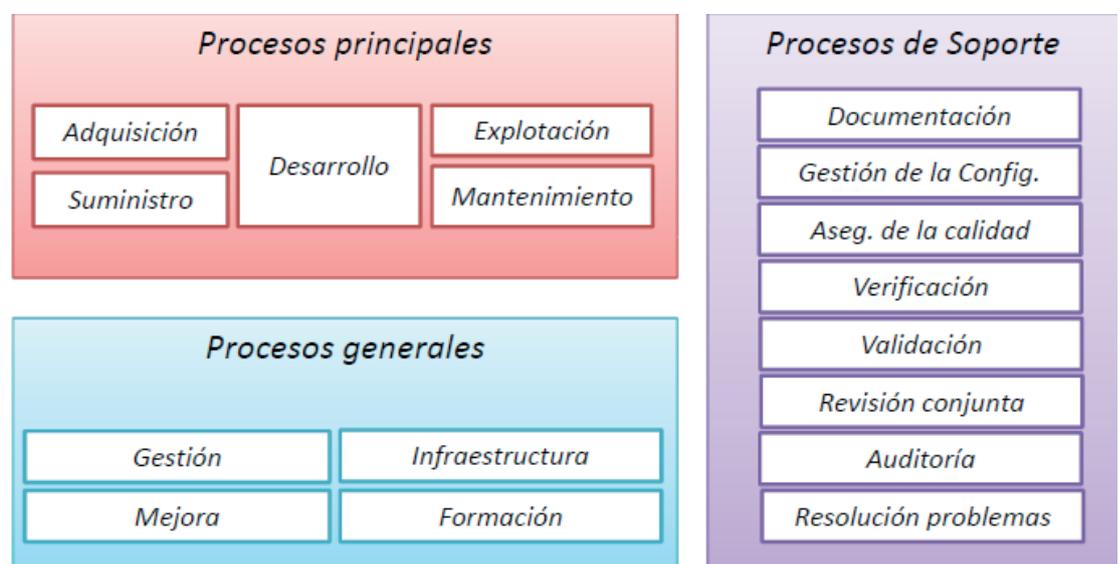


Figura 2: Estructura de la norma ISO/IEC 12207:2008 (Adaptado de [7]).

3.2. Modelos ágiles de desarrollo de software

Los métodos ágiles de desarrollo dependen de un enfoque iterativo para la especificación, desarrollo y entrega del software y principalmente fueron diseñados para apoyar al desarrollo de aplicaciones de negocios donde los requerimientos del sistema normalmente cambiaban rápidamente durante el proceso de desarrollo, están pensados para entregar software funcional de forma rápida a los clientes, quienes piden que se incluya iteraciones posteriores del sistema o nuevos requerimientos o cambios en el mismo.

En febrero de 2001, nace el término “ágil” aplicado al desarrollo de software. En esta reunión participan un grupo de 17 expertos de la industria del software. Su objetivo fue proponer los valores y principios que deberían permitir a los equipos desarrollar software rápidamente y respondiendo a los cambios que puedan surgir a lo largo del proyecto [13]. El punto de partida fue el Manifiesto Ágil, un documento que resume la filosofía ágil. El Manifiesto Ágil estableció un conjunto común de valores y principios dominantes para todas las metodologías ágiles individuales en el momento; aquí se detallan cuatro valores clave para habilitar equipos de alto rendimiento: 1) Los individuos e interacciones sobre procesos y herramientas, 2) Entregar software que funciona sobre documentación comprensiva, 3) Colaboración del cliente sobre negociación de contratos, y 4) Responder al cambio sobre desarrollar un plan.

Aunque los creadores e impulsores de las metodologías ágiles más populares han suscrito el manifiesto ágil y coinciden con sus principios, cada metodología tiene características propias y hace hincapié en algunos aspectos más específicos. Entre las metodologías más representativas se encuentran: Extrem Programming, SCRUM, Crystal Methodologies, Dynamic Systems Development Method (DSDM), Adaptive Software Development (ASD), Adaptive Software Development (ASD), Feature-Driven Development (FDD), Lean Development (LD) [14].

Estos modelos ágiles ofrecen una solución casi adecuada para una gran cantidad de proyectos; sus principales ventajas son: poca documentación, simplicidad, análisis como una actividad constante, diseño evolutivo, testeos diarios; sin embargo, estas ventajas pueden ser las causantes de varios problemas en el proceso de desarrollo de software; como cualquier otra metodología también surgen problemas al implementarlas como por ejemplo: a) Falta de documentación del diseño. b) Da énfasis al desarrollo del código, dejando de lado el respeto a planificación, el proceso ágil asume una disciplina propia de cada miembro del equipo [15]. b) El proyecto carece de una gestión centralizada [16], c) El código no puede tomarse como una documentación.

En sistemas de tamaño grande se necesitan leer los cientos o miles de páginas del listado de código fuente. d) Falta de calidad. Probar el código de forma constante no genera productos de calidad, sólo revela falta de análisis y diseño. e) Fuerte dependencia de las personas. Como se evita en lo posible la documentación y los diseños convencionales, los proyectos ágiles dependen críticamente de las personas. f) Sobre costos y retrasos derivados de la refactorización continua. g) Los modelos de datos son “pesados” y no pueden cambiarse fácilmente solo porque el cliente desea incorporar más funciones al sistema. h) Problemas derivados del fracaso de los proyectos ágiles. Si un proyecto ágil fracasa no hay documentación o hay muy poca; lo mismo ocurre con el diseño. La comprensión del sistema se queda en las mentes de los desarrolladores.

Ambler [17] y Santos [18], determinaron varias buenas prácticas utilizadas en el desarrollo de software ágil, que servirán de base para el presente estudio. Estas prácticas son: Cliente on-site, casos de usuario, escenarios, documento de la visión, historias de usuario, soluciones vía Spike, modelo de dominio, estudio del objetivo de negocio, diagramas UML, refactoring, programación en parejas, propiedad colectiva del código, detección de errores, pruebas posteriores al desarrollo, propiedad individual de código, pruebas funcionales, desarrollo guiado por pruebas, pruebas unitarias, pruebas automatizadas, integración continua, iteraciones fijas, gráficos burndown, Kanban, planificación, velocidad, estimaciones realizadas por características, informes de progreso.

4. Portafolios y gestión de proyectos

Rothman [19], considera a un portafolio como la organización de proyectos ordenados de acuerdos a la fecha y valor, que los comités de la organización planifican, en cierto modo, el portafolio es un gran diseño visible, el cual debe ayudar a decidir lo siguiente: a) Cuando comprometerse con un proyecto para que el equipo desarrolle el producto, para comenzar o continuar un proyecto. b) Cuando es el momento para finalizar el proyecto y liberar al equipo de trabajo para realizar otras actividades en otro proyecto. c) Cuando transformar un proyecto y comprometerse con el proyecto modificado. d) Cuando decidir por seleccionar un proyecto entre otros proyectos, el portafolio proporciona una herramienta visual que ayuda a negociar cuales proyectos a ejecutar y cuando.

La gestión del portafolio de proyectos describe la gestión de una organización de portafolios de iniciativas de negocio, ésta es una colección ordenada de procesos estratégicos y decisiones que juntas habilitan un balance más efectivo del cambio organizacional y del negocio [20]. La gestión de proyectos TI es importante ya que

las organizaciones de TI exitosas son aquellas que ven más allá de las necesidades de un solo sistema. Las actividades fundamentales de esta disciplina incluyen la identificación y selección de proyectos, el monitoreo y gobierno de proyectos, y la gestión del inventario de activos de TI [21].

Por otro lado, la gestión de la demanda TI, se define como a un megaproyecto, en el que se determina que proyectos de TI deben ser habilitados y ejecutados por el departamento de TI [22], [23], y se realiza a través de portafolios, que se constituye a su vez en procesos del negocio. Es importante considerar que la gestión de la demanda es una manera de alcanzar rentabilidad para satisfacer las necesidades de los clientes y al mismo tiempo garantizar servicio y calidad [24], [25].

La demanda TI, puede provenir del nivel estratégico, del nivel táctico o del nivel operativo de la empresa; la demanda estratégica se gestiona a través del portafolio de proyectos, es decir, la demanda de nuevos proyectos que introducen la innovación y activan nuevos negocios, productos y servicios.

La gestión del portafolio de proyectos ofrece un proceso basado en hechos para evaluar, priorizar y monitorizar proyectos. PPM une procesos de *planificación estratégica, recursos, la asignación de presupuestos, la selección e implementación de proyectos y las métricas "post-mortem"* de proyectos.

En el estándar del PMI® [26], para la dirección de portafolio podemos encontrar un modelo de procesos muy detallado que nos lleva de la estrategia al portafolio y de éste a los programas y proyectos. Un modelo más simple aparece en la Figura 3 adaptada de Archer y Ghasemzadeh [27]. Este esquema de procesos nos relaciona los tres niveles, estrategia, portafolio y proyecto.

El modelo empieza con la propuesta del proyecto y su análisis individual. Este análisis, pretende hacer una evaluación individual del riesgo asociado a la realización del proyecto donde se usan criterios financieros tales como VAN, TIR y ROI y criterios de valoración del alineamiento estratégico. Algunos proyectos son ya descartados en esta etapa.

Los proyectos que pasan los criterios individuales entran en el proceso de selección de proyecto donde se comparan los proyectos entre ellos, tanto los proyectos que están en ejecución como los proyectos recién propuestos. La selección se basa en la evaluación de varios criterios, Estos criterios, al igual que en el análisis individual de proyectos miden riesgo, beneficio y alineamiento estratégico. Con los proyectos seleccionados se realiza un balanceo y priorización de los proyectos. En base a los recursos disponibles y a la valoración previa, los proyectos se categorizan y priorizan y se asignan recursos a ellos. El seguimiento de los proyectos se realiza en base a esta

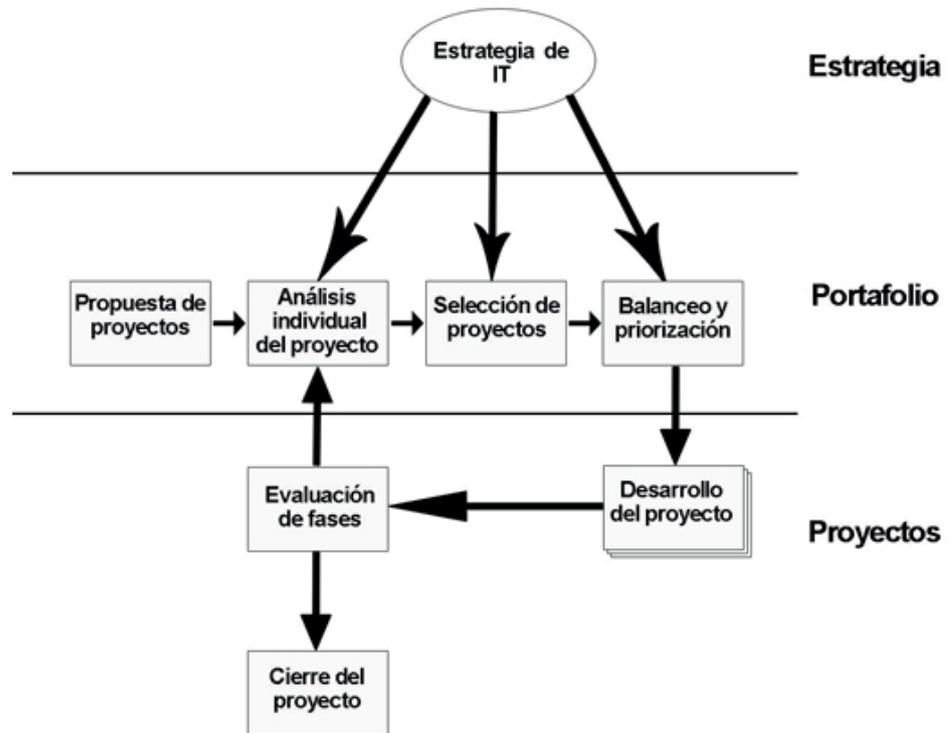


Figura 3: Modelo de procesos [27].

priorización y categorización. El resultado de este proceso supone una actualización de los planes de proyectos individuales, ajustándolos a las nuevas prioridades. A partir de este punto el proceso se convierte en iterativo, los proyectos se van ejecutando de acuerdo al plan actualizado y, a medida que determinadas etapas son desarrolladas, se evalúa continuamente el proyecto de manera individual y respecto al resto del portafolio, hasta su finalización o cancelación.

La gestión de proyecto basado en el marco de trabajo PMBOK, creado por el Project Management Institute PMI, es el modelo más difundido y aceptado para la gestión de proyectos en general (no solo proyectos de TI). Dicho modelo se basa en un conjunto de buenas prácticas divididas en 9 áreas de conocimiento, cada una de las cuales se sub-divide en actividades (siendo 44 en total). Este modelo nos brinda un esquema de trabajo para gestionar cada aspecto de un proyecto: desde gestión del alcance hasta gestión de las adquisiciones. Es importante mencionar también que cada organización debe determinar que partes del marco de trabajo de PMBOK es aplicable a la compañía. Esto dependerá de la envergadura y nivel de detalle y control que se desee tener de cada proyecto. Por ello, se debe pensar en PMBOK como un conjunto de lineamientos generales, de los cuales la organización se puede alimentar para establecer una metodología de trabajo propia.

Según PMI®, la gestión de proyectos se estructura según el estándar ANSI/PMI 99-001-2018, conocido como PMBOK. En su versión 4 el cuerpo de conocimientos necesario para dirigir proyectos consta de 42 procesos, 9 áreas de conocimiento y 5 grupos de procesos. Los grupos de procesos son los siguientes: **Inicio**: Definición del nuevo proyecto o fase mediante la obtención de la autorización para comenzar. **Planificación**: Establecimiento del alcance del proyecto, refinamiento de los objetivos y definición del curso de las acciones necesarias para alcanzar los objetivos. **Ejecución**: Realización del trabajo definido en el plan. **Seguimiento y Control**: Seguimiento, análisis, y regulación del progreso del proyecto, identificando áreas en las que el plan requiera cambios e inicio de los cambios correspondientes. **Cierre**: Finalización formal y ordenada de todas las actividades del proyecto o fase.

5. Relaciones entre CMMI, el portafolio de proyectos y el proceso de desarrollo

El presente trabajo realiza una relación entre el ciclo de vida del desarrollo de software y CMMI, para lograr este objetivo, hemos abordado dos enfoques: 1) desde el punto de vista de los modelos tradicionales de desarrollo de software, a través del estándar ISO/IEC 12207, y 2) Desde el punto de vista de los modelos ágiles.

5.1. CMMI y el estándar ISO 12207

Para realizar este análisis nos hemos basado tanto en las áreas de procesos de CMMI como en los niveles de madurez propuestos por esta norma y la hemos relacionado con los procesos establecidos en el estándar ISO/IEC 12207; tomando en cuenta que éste estándar orienta el desarrollo de software desde un punto de vista de los modelos tradicionales del ciclo de vida. En la Tabla 2 se muestran los resultados, se han identificado como PP a los procesos principales del estándar, PS a los procesos de soporte y PG a los procesos generales.

De la tabla anterior se puede determinar que de acuerdo al estándar ISO/IEC 12207 los Procesos Principales de *Operación y Mantenimiento*, no son considerados por CMMI; de la misma manera, el Proceso de Documentación es implícito para CMMI para todas las áreas de procesos. En lo que tiene que ver con los Procesos Generales, la *Infraestructura y la Formación*, no son enfocados por CMMI.

Por otro lado, existen 7 áreas del proceso de CMMI, que no son cubiertas por el estándar ISO/IEC 12207, la mayoría de ellas está relacionada con la administración y

TABLA 2: Relación de CMMI e ISO/IEC 12207.

Nivel	Áreas de procesos	ISO/IEC 12207:2008
5	CAR: Análisis de causas y resolución	PS. Solución de problemas
	OPM: Administración del rendimiento de las organizaciones	---
4	OPP: Rendimiento de los procesos organizativos	---
	QPM: Administración cuantitativa de proyectos	PG. Gestión
3	RD: Desarrollo de requisitos	PP. Desarrollo
	TS: Solución técnica	PP. Suministro
	PI: Integración de productos	PP. Desarrollo
	VER: Comprobación	PP. Revisiones Conjuntas PS. Auditoría PS. Verificación PS. Validación
	VA: Validación	PS. Validación
	IPM: Administración integrada de proyectos	PG. Gestión
	RSKM: Administración de riesgos	---
	OPF: Enfoque de los procesos organizativos	---
	OPD: Definición de procesos organizativos	---
	OT: Aprendizaje organizativo	---
	DAR: Análisis de decisiones y Resolución	PS: Solución de problemas
	Control de Calidad de los procesos y productos	PS. Aseguramiento de la Calidad PG. Mejora
2	CM: Administración de configuraciones	PS. Gestión de la Configuración
	MA: Medición y análisis	---
	SAM: Administración de acuerdos con proveedores	PP. Adquisición PP. Suministro
	PP: Planeación de proyectos	PG. Gestión
	PMC: Supervisión y control de proyectos	PG. Gestión
	RM: Administración de requisitos	PP. Desarrollo
1	No hay áreas de proceso dentro del nivel 1 del modelo.	

establecimiento de procesos a nivel organizativo, considerando que son 22 áreas del proceso CMMI, el 31.81% de ellas no es considerado por el mencionado estándar.

Se puede establecer que, si bien ambos están orientados a lograr un software de calidad, una empresa que aplica ISO/IEC 12207, puede complementar sus procesos y tener una certificación de calidad CMMI.

5.2. CMMI y los Modelos Ágiles de Desarrollo

Marçal [28], en su trabajo presenta un mapeo entre CMMI Scrum, pero solamente aborda las áreas de Procesos de Dirección de Proyectos de CMMI; a diferencia de esto, para determinar la relación entre CMMI y los modelos ágiles, en esta investigación se

consideraron todas las áreas claves del negocio propuestas por CMMI a las que se les ha organizado por el enfoque organizativo a los que pertenecen; la relación con los modelos ágiles está basada en las prácticas ágiles determinadas por [18]. La Tabla 3, muestra la relación indicada.

Las buenas prácticas aplicadas en los modelos ágiles se aplican al 91% de las 22 áreas de procesos definidas por CMMI; sin embargo, existen dos áreas del negocio que no tienen un fuerte soporte de prácticas aplicadas, como es el caso del *análisis a las decisiones y resolución, el análisis de causas y resolución, rendimiento de los procesos administrativos*. No existen prácticas que soporten la *Administración de riesgos y la implementación e innovación administrativas*.

CMMI es un modelo no una metodología. CMMI se centra en el **qué** se espera encontrar en una organización, mientras que metodologías y métodos ágiles se centran en el **cómo** elaborar productos del ciclo de vida del software. En CMMI no se establece orden en la ejecución de los procesos, ni determina un ciclo de vida, son las metodologías quienes determinan este punto, recomendando, por ejemplo, el ciclo de vida iterativo e incremental.

5.3. CMMI, PPM y PMBOK

La relación entre CMMI y PPM, se realizaron a través de las áreas de procesos CMMI y las fases de PPM, desde la propuesta de proyectos hasta la priorización de los mismos. En lo que se refiere a PMBOK se han considerado las 9 áreas del conocimiento que éste estándar propone.

De la Tabla 4 se puede determinar que la Gestión de Portafolios de Proyectos no es considerada en su totalidad por las áreas de procesos CMMI, de las cuatro fases establecidas: Propuesta de proyectos, Análisis individual de proyectos, Selección de proyectos y Balance y Priorización de proyectos, CMMI llega hasta la fase 3, la última fase no es considerada por CMMI. Por otra parte, CMMI contempla la administración de procesos administrativos que no es considerada por PPM. Se puede determinar que el 75% de las fases de PPM son cubiertas por CMMI

En la relación de CMMI y PMBOK, se observa que, las 9 áreas del conocimiento propuestas por PMBOK, están consideradas por PMBOK, el *grupo de procesos de seguimiento y control* es el que relaciona todas las áreas de PMBOK con el proceso de *Comprobación y Supervisión y Control de Proyectos* de CMMI.

TABLA 3: Relación CMMI y prácticas ágiles.

Enfoque organizativo	Área de procesos	Prácticas ágiles
Ingeniería	Desarrollo de requisitos	Cliente on -siteHistorias de usuarioCasos de usuarioEscenarios
	Solución técnica	Modelo de dominioDiagramas UML
	Integración de productos	Integración Continua
	Comprobación	Pruebas posteriores al desarrolloPruebas funcionalesdesarrollo guiado por PruebasPruebas unitariasPruebas automatizadasIntegración continua
	Validación	Pruebas posteriores al desarrolloPruebas funcionalesPruebas automatizadasIntegración continua
Administración de proyectos	Planeación de proyectos	Iteraciones FijasPlanificación de iteracionesKanban
	Supervisión y control de proyectos	Gráficos BurndownEstimaciones realizadas por característicasInformes de progreso
	Administración integrada de proyectos	Kanban
	Administración de acuerdos con proveedores	Soluciones vía Spike
	Administración de requisitos	Cliente on -siteHistorias de usuarioCasos de usuarioEscenarios
	Administración de riesgos	--
Administración de procesos	Administración cuantitativa de proyectos	Gráficos burndown Kanban
	Enfoque de los procesos organizativos	Documento de la VisiónEstudio del objetivo de negocio
	Definición de procesos organizativos	Documento de la VisiónEstudio del objetivo de negocio
	Aprendizaje organizativo	Programación en parejasPropiedad colectiva del código
	Rendimiento de los procesos organizativos	Velocidad
Compatibilidad	Innovación e implementación organizativas	--
	Administración de configuraciones	Integración Continua
	Control de calidad de los procesos y productos	Pruebas funcionalesDesarrollo guiado por pruebasPruebas unitariasPruebas automatizadas
	Medición y análisis	Iteraciones FijasKanban
	Análisis de decisiones y resolución	Soluciones vía Spike
	Análisis de causas y resolución	Soluciones vía Spike

TABLA 4: Relación CMMI y PPM - PMBOK.

Nivel	Áreas de procesos CMMI	Fase PPM	Área del conocimiento PMBOK
5	CAR: Análisis de causas y resolución	Análisis individual del proyecto	—
	OPM: Administración del rendimiento de las organizaciones	—	Gestión de la Integración del proyecto
4	OPP: Rendimiento de los procesos organizativos	—	Gestión de la Integración del proyecto
	QPM: Administración cuantitativa de proyectos	—	Gestión de la Integración del proyecto
3	RD: Desarrollo de requisitos	—	Gestión del alcance del proyecto
	TS: Solución técnica	Propuesta de proyectos	Gestión del alcance del proyecto
	PI: Integración de productos	—	Gestión de la Integración del proyecto
	VER: Comprobación	—	Todas las áreas
	VA: Validación	—	Gestión del alcance del proyecto
	IPM: Administración integrada de proyectos	—	Gestión de la Integración del proyecto
	RSKM: Administración de riesgos	—	Gestión de riesgos del proyecto
	OPF: Enfoque de los procesos organizativos	—	—
	OPD: Definición de procesos organizativos	—	—
	OT: Aprendizaje organizativo	—	—
	DAR: Análisis de decisiones y Resolución	Selección de proyectos	—
	Control de Calidad de los procesos y productos	—	Gestión de la calidad del proyecto
2	CM: Administración de configuraciones	—	—
	MA: Medición y análisis	—	Todas las áreas
	SAM: Administración de acuerdos con proveedores	—	Gestión de las adquisiciones del proyecto
	PP: Planeación de proyectos	Propuestas de proyectos	—
	PMC: Supervisión y control de proyectos	—	Todas las áreas
	RM: Administración de requisitos	—	Gestión del alcance del proyecto
1	No hay áreas de proceso dentro del nivel 1 del modelo.	—	—

El grupo de procesos de Planificación de PMBOK es sumamente detallado, en él se consideran aspectos como: gestión del tiempo, alcance, costos, comunicaciones, recurso humano, interesados; que no son abordados completamente por CMMI.

6. Conclusiones

Existen 7 áreas del proceso de CMMI, que no son cubiertas por el estándar ISO/IEC 12207, la mayoría de ellas están relacionadas con la administración y establecimiento de procesos a nivel organizativo, es decir el 31.81% de CMMI no es considerado por el mencionado estándar. Se puede establecer que, si bien ambos están orientados a lograr un software de calidad, una empresa que aplica ISO/IEC 12207, puede complementar sus procesos y tener una certificación de calidad CMMI.

CMMI muestra áreas de proceso, no procesos en sí, muestra tipologías de proceso, que luego en cada organización pueden instanciarse de diferente manera, y pueden existir numerosas correspondencias entre "áreas de proceso" y "procesos" de la organización, etc.

Independientemente del modelo de ciclo de vida que se utilice, sea modelos tradicionales o ágiles; lo importante es que la empresa cumpla los requerimientos de calidad que CMMI exige, lo cual incrementará el nivel de competitividad de sus productos.

En este trabajo se muestra que CMMI y los modelos ágiles pueden ser combinados con éxito ya que el 91% de las prácticas ágiles son cubiertas por las áreas de procesos de CMMI.

CMMI contempla las fases de PPM: propuesta de proyectos, análisis individual de proyectos, selección de proyectos; siendo el balanceo y priorización, la única fase no considerada en CMMI. El grupo de procesos de Planificación de PMBOK es sumamente detallado, en él se consideran aspectos como: gestión del tiempo, alcance, costos, comunicaciones, recurso humano, interesados; que no son abordados completamente por CMMI.

Considerando que PPM es instrumento para gestionar la cartera de proyectos de una empresa a fin de obtener rentabilidad, PMBOK permite establecer buenas prácticas en relación a la gestión, administración y dirección de proyectos y el desarrollo de proyectos utilizan metodologías tradicionales o ágiles; es importante aplicar un estándar de calidad como CMMI que asegure la calidad de los proyectos de desarrollo de software desde el momento de su formulación hasta su implementación.

Referencias

- [1] The Standish Group: Chaos Manifiesto 2013, <http://www.versionone.com/assets/img/files/CHAOSManifiesto2013.pdf>, (2013).
- [2] ITIL, COBIT, CMMI, PMBOK: Como integrar y adoptar los estándares para un buen Gobierno de TI, <http://helkyncoello.wordpress.com/2008/12/08/itil-cobit-cmmi-pmbok-como-integrar-y-adoptar-los-estandares-para-un-buen-gobierno-de-ti/>.
- [3] CMMI Institute: CMMI Institute - the home of Capability Maturity Model Integration, <http://cmmiinstitute.com/>.
- [4] AESOFT: aesoft - Estudio de Mercado Sector Software y Hardware 2011, <http://aesoft.com.ec/www/index.php/118-slideshow/154-http-www-slideshare-net-aesoft-ot-20489-microsoftfolleto>.
- [5] SEI: CMMI for SCAMPI SM Class A Appraisal Results 2012Mid-Year Update. Carnegie Mellon, USA (2012).
- [6] Baker, S.W.: Formalizing agility, part 2: How an agile organization embraced the CMMI. In: Agile Conference, 2006. p. 8–pp. IEEE (2006).
- [7] ISO: ISO/IEC 12207:2008, <https://www.iso.org/obp/ui/#iso:std:iso-iec:12207:ed-2:v1:en>.
- [8] José Sánchez del Río: Qué es CMMI y para qué sirve -, <http://www.cantabriatic.com/que-es-cmmi-y-para-que-sirve/>, (2014).
- [9] SEI: CMMI® for Development, Version 1.3. Software Engineering Process Management Program (2010).
- [10] Larrondo Petrie, M., Hugo, V., García, M., Giraldo, G.: Modelo de Registro y Acreditación de Instituciones de Educación Superior basado en el Modelo CMMI. In: San Cristóbal, Venezuela June 2-5, 2009 7 th Latin American and Caribbean Conference for Engineering and Technology WE1- 1 Seventh LACCEI Latin American and Caribbean Conference for Engineering and Technology (LACCEI'2009) "Energy and Technology for the Americas: Education, Innovation, Technology and Practice." Venezuela (2009).
- [11] Anderson, D.: Kanban. Blue Hole Press (2010).
- [12] ISO/IEC 12207: ISO/IEC 12207:2008 - Systems and software engineering – Software life cycle processes, http://www.iso.org/iso/catalogue_detail?csnumber=43447.
- [13] Agile Alliance: Agile Alliance?:: The Twelve Principles of Agile Software, <http://www.agilealliance.org/the-alliance/the-agile-manifesto/the-twelve-principles-of-agile-software/>.

- [14] Letelier, P., Penadés, M.C.: Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP), <http://www.cyta.com.ar/tao502/v5n2a1.htm>.
- [15] Minna Pikkarainen, X.W.: An investigation of agility issues in scrum teams using agility indicators.
- [16] Esfahani, H.C., Cabot, J., Yu, E.: Adopting agile methods: Can goal-oriented social modeling help? In: 2010 Fourth International Conference on Research Challenges in Information Science (RCIS). pp. 223-234 (2010).
- [17] Ambler, S.: Quality in an agile world. *Softw. Qual. Prof.* 7, 34 (2005).
- [18] Santos, M. de A.: Agile Practices: An Assessment of Perception of Value of Professionals on the Quality Criteria in Performance of Projects. *J. Softw. Eng. Appl.* 04, 700-709 (2011).
- [19] Rothman, J.: *Manage Your Project Portfolio: Increase Your Capacity and Finish More Projects*. Pragmatic Bookshelf, Raleigh, N.C. (2009).
- [20] Sowden, R.: *Portfolio, Programme and Project Management Maturity Model*, (2008).
- [21] Scott Ambler: *Gestión Ágil del Portafolio de Proyectos*. *Cut. IT J.* (2009).
- [22] Baschab, J., Piot, J.: *The Executive's Guide to Information Technology*. John Wiley & Sons Inc (2007).
- [23] Aguilar Alonso, I., Carrillo Verdún, J., Tovar Caro, E.: Importancia de la Gestión del Proceso de la demanda de TI, (2008).
- [24] Thomason, D.: Strategic, tactical, operational [demand management]. *Manuf. Eng.* 83, 34-37 (2004).
- [25] Aguilar Alonso, I., Carrillo Verdún, J., Tovar Caro, E.: Description of the structure of the IT demand management process framework. *Int. J. Inf. Manag.* 37, 1461-1473 (2017).
- [26] Project Management Institute: *Estandares en la gestión de Portafolios del PMI y PMBOK*. (2006).
- [27] F. Ghasemzadeh, N.P. Archer: Project portfolio selection through decision support. *Decis. Support Syst.* 29, 73-88 (2000).
- [28] Marcal, A.S.C., de Freitas, B.C.C., Furtado Soares, F.S., Belchior, A.D.: Mapping CMMI project management process areas to SCRUM practices. In: *Software Engineering Workshop, 2007. SEW 2007. 31st IEEE*. pp. 13-22. IEEE (2007).