

Capability Maturity Model (CMM)

Reingeniería de Procesos

Universidad Santa María

27/01/2020

Brian Arriaga C.I.: 23.694.578

Kevin Cabrera C.I.: 21.282.157

**ÍNDICE**

[**Capability Maturity Model (CMM)** 3](#_Toc30938556)

[**Niveles de madurez de CMM de procesos de software** 4](#_Toc30938557)

[**Áreas clave del proceso o Key Process Areas (KPA’s)** 5](#_Toc30938558)

[**Nivel 1: Inicial** 7](#_Toc30938559)

[**Nivel 2: Repetible** 7](#_Toc30938560)

[**Nivel 3: Definido** 8](#_Toc30938561)

[**Nivel 4: Administrado** 8](#_Toc30938562)

[**Nivel 5: Optimización** 9](#_Toc30938563)

[**Referencias** 10](#_Toc30938564)

**Capability Maturity Model (CMM)**

CMM fue desarrollado por el Instituto de Ingeniería de Software (Software Engineering Institute o SEI) en la Universidad Carnegie Mellon en 1987.

* No es un modelo de proceso de software. Es un framework que se utiliza para analizar el enfoque y las técnicas seguidas por cualquier organización para desarrollar un producto de software.
* También proporciona pautas para mejorar aún más la madurez de esos productos de software.
* Se basa en la retroalimentación profunda y las prácticas de desarrollo adoptadas por las organizaciones más exitosas del mundo.
* Este modelo describe una estrategia que debe seguirse moviéndose a través de 5 niveles diferentes.
* Cada nivel de madurez muestra un nivel de capacidad de proceso. Todos los niveles, excepto el nivel 1, se describen con más detalle en las áreas de proceso clave (KPA).

El Modelo de Madurez de Capacidades (CMM) es una metodología utilizada para desarrollar y refinar el proceso de desarrollo de software de una organización. El modelo describe un camino evolutivo de cinco niveles de procesos cada vez más organizados y sistemáticamente más maduros. CMM fue desarrollado y es promovido por el Software Engineering Institute (SEI), un centro de investigación y desarrollo patrocinado por el Departamento de Defensa de los Estados Unidos (U.S. Department of Defense o DoD). SEI se fundó en 1984 para abordar problemas de ingeniería de software y, en un sentido amplio, para avanzar en las metodologías de ingeniería de software. Más específicamente, SEI se estableció para optimizar el proceso de desarrollo, adquisición y mantenimiento de sistemas altamente dependientes del software para el DoD. Debido a que los procesos involucrados son igualmente aplicables a la industria del software en su conjunto, SEI aboga por la adopción del CMM en toda la industria.

El CMM es similar a ISO 9001, una de las series de normas ISO 9000 especificadas por la Organización Internacional de Normalización (International Organization for Standardization o ISO). Las normas ISO 9000 especifican un sistema de calidad efectivo para las industrias manufactureras y de servicios; ISO 9001 se ocupa específicamente del desarrollo y mantenimiento de software. La principal diferencia entre los dos sistemas radica en sus respectivos propósitos: ISO 9001 especifica un nivel de calidad mínimo aceptable para los procesos de software, mientras que el CMM establece un framework para la mejora continua del proceso y es más explícito que el estándar ISO para definir los medios a emplear con ese fin.

**Niveles de madurez de CMM de procesos de software**

* En el nivel *inicial*, los procesos son desorganizados, incluso caóticos. Es probable que el éxito dependa de los esfuerzos individuales, y no se considera que sea repetible, porque los procesos no estarían suficientemente definidos y documentados para permitir su replicación.
* En el nivel *repetible*, se establecen técnicas básicas de gestión de proyectos y se pueden repetir los éxitos, ya que los procesos necesarios se habrían establecido, definido y documentado.
* En el nivel *definido*, una organización ha desarrollado su propio proceso de software estándar a través de una mayor atención a la documentación, la estandarización y la integración.
* En el nivel *administrado*, una organización supervisa y controla sus propios procesos a través de la recopilación y el análisis de datos.
* En el nivel de *optimización*, los procesos se mejoran constantemente a través del monitoreo de los comentarios de los procesos actuales y la introducción de procesos innovadores para satisfacer mejor las necesidades particulares de la organización.

**Áreas clave del proceso o Key Process Areas (KPA’s)**

Cada uno de estos KPA define los requisitos básicos que debe cumplir un proceso de software para satisfacer el KPA y alcanzar ese nivel de madurez.

Conceptualmente, las áreas clave del proceso forman la base para el control de gestión del proyecto de software y establecen un contexto en el que se aplican métodos técnicos, se producen productos de trabajo como modelos, documentos, datos, informes, etc., se establecen hitos, se garantiza la calidad y el cambio se gestiona adecuadamente.

Sin KPA

Planificación del Proyecto

Gestión de la Configuración

Gestión de Requisitos

Gestión de Subcontratos

Garantía de Calidad del Software

Revisiones por Pares

Coordinación

Definición del Proceso de la Organización

Enfoque del Proceso de la Organización

Programas de Capacitación

Gestión de la Calidad del Software

Gestión Cuantitativa

Gestión de Cambio de Procesos

Gestión del Cambio Tecnológico

Prevención de Defectos

Nivel 1

Nivel 2

Nivel 3

Nivel 4

Nivel 5

**INICIAL**

**REPETIBLE**

**DEFINIDO**

**ADMINISTRADO**

**OPTIMIZACIÓN**

Los 5 niveles de CMM son los siguientes:

**Nivel 1: Inicial**

* No hay KPA definidos.
* Los procesos seguidos son improvisados e inmaduros y no están bien definidos.
* Entorno inestable para el desarrollo de software.
* No hay base para predecir la calidad del producto, el tiempo de finalización, etc.

**Nivel 2: Repetible**

* Se enfoca en establecer políticas básicas de gestión de proyectos.
* La experiencia con proyectos anteriores se utiliza para gestionar nuevos proyectos similares.

KPA’s:

* Planificación del Proyecto: incluye la definición de los recursos necesarios, objetivos, limitaciones, etc. para el proyecto. Presenta un plan detallado que debe seguirse sistemáticamente para completar con éxito un software de buena calidad.
* Gestión de la Configuración: el objetivo es mantener el rendimiento del producto de software, incluidos todos sus componentes, durante todo el ciclo de vida.
* Gestión de Requisitos: incluye la gestión de las revisiones y comentarios de los clientes que dan como resultado algunos cambios en el conjunto de requisitos. También consiste en acomodar esos requisitos modificados.
* Gestión de Subcontratos: se centra en la gestión eficaz de contratistas de software calificados, es decir, gestiona las partes del software desarrolladas por terceros.
* Garantía de Calidad del Software: garantiza un producto de software de buena calidad siguiendo ciertas reglas y pautas estándar de calidad durante el desarrollo.

**Nivel 3: Definido**

* En este nivel, se lleva a cabo la documentación de las pautas y procedimientos estándar.
* Es un conjunto integrado bien definido de proyectos de ingeniería y procesos de gestión de software específicos.

KPA’s:

* Revisiones por Pares: en este método, los defectos se eliminan mediante el uso de una serie de métodos de revisión, como tutoriales, inspecciones, chequeo de amigos, etc.
* Coordinación Intergrupal: consiste en interacciones planificadas entre diferentes equipos de desarrollo para garantizar el cumplimiento eficiente y adecuado de las necesidades del cliente.
* Definición del Proceso de la Organización: su enfoque principal es el desarrollo y mantenimiento de los procesos de desarrollo estándar.
* Enfoque del Proceso de la Organización: incluye actividades y prácticas que deben seguirse para mejorar las capacidades del proceso de una organización.
* Programas de Capacitación: se enfoca en mejorar el conocimiento y las habilidades de los miembros del equipo, incluidos los desarrolladores, y garantizar un aumento en la eficiencia del trabajo.

**Nivel 4: Administrado**

* En esta etapa, se establecen objetivos de calidad cuantitativos para la organización para productos de software y procesos de software.
* Las mediciones realizadas ayudan a la organización a predecir el producto y la calidad del proceso dentro de algunos límites definidos cuantitativamente.

KPA’s:

* Gestión de la Calidad del Software: incluye el establecimiento de planes y estrategias para desarrollar un análisis cuantitativo y la comprensión de la calidad del producto.
* Gestión cuantitativa: se centra en controlar el rendimiento del proyecto de forma cuantitativa.

**Nivel 5: Optimización**

* Este es el nivel más alto de madurez del proceso en CMM y se enfoca en la mejora continua del proceso en la organización utilizando retroalimentación cuantitativa.
* El uso de nuevas herramientas, técnicas y evaluación de procesos de software se realiza para evitar la recurrencia de defectos conocidos.

KPA’s:

* Gestión de Cambio de Procesos: se centra en la mejora continua de los procesos de software de la organización para mejorar la productividad, la calidad y el tiempo de ciclo del producto de software.
* Gestión del Cambio Tecnológico: consiste en la identificación y el uso de nuevas tecnologías para mejorar la calidad del producto y disminuir el tiempo de desarrollo del mismo.
* Prevención de Defectos: se centra en la identificación de las causas de los defectos y para evitar que se repitan en proyectos futuros mejorando el proceso definido por el proyecto.

**Referencias**

What is Capability Maturity Model (CMM)? - Definition from WhatIs.com. (2020). <https://searchsoftwarequality.techtarget.com/definition/Capability-Maturity-Model>. 26 de enero 2020

Software Engineering | Capability maturity model (CMM) - GeeksforGeeks. (2018). <https://www.geeksforgeeks.org/software-engineering-capability-maturity-model-cmm/>. 26 de enero 2020