**GUIA DE APRENDIZAJE CLASIFICACIÓN DE METRICAS**

**JENNIFER ANDREA ROMANI JAMAICA**

**JUAN DAVID SABOYÁ JIMENEZ**

**SERVICIO NACIONAL DE APRENDIZAJE (SENA)**

**CENTRO DE DISEÑO Y METROLOGÍA**

**ANÁLISIS Y DESARROLLO DE SISTEMAS DE INFORMACIÓN**

**FICHA 683019**

**BOGOTÁ D.C.**

**2015**

**SOLUCIÓN**

**1) Investigar sobre el modelo de calidad *CMMI*:**

CMMI (Capability Maturity Model Integration) es un modelo de madurez de mejora de los procesos para el desarrollo de productos y de servicios. Consiste en las mejores prácticas que tratan las actividades de desarrollo y de mantenimiento que cubren el ciclo de vida del producto, desde la concepción a la entrega y el mantenimiento.

* Definir cada uno de los niveles del modelo CMMI y de un ejemplo.

Los seis niveles de capacidad, especificados por los números 0 hasta 5, son los siguientes:

0. Incompleto

1. Realizado

2. Gestionado

3. Definido

4. Gestionado cuantitativamente

5. En optimización

**Nivel de capacidad 0: Incompleto**

Un "proceso incompleto" es un proceso que, o bien no se ejecuta, o se ejecuta parcialmente. Al menos una de las metas específicas del área del proceso no se satisface y no existe metas genéricas para ese nivel, ya que no hay ninguna razón para institucionalizar un proceso ejecutado parcialmente.

**Ejemplo:** La transferencia no realiza de un archivo.

**Nivel de capacidad 1: Realizado**

Un proceso de nivel de capacidad 1 se caracteriza cómo un "proceso realizado". Un proceso realizado es un proceso que satisface las metas específicas del área de proceso. Soporta y permite el trabajo necesario para producir los productos del trabajo. Aunque el nivel de capacidad 1 da como resultado mejoras importantes, esas mejoras pueden perderse en el tiempo si no se institucionalizan. La aplicación de la institucionalización (las prácticas genéricas de CMMI en los niveles de capacidad 2 a 5) ayuda a asegurar que las mejoras se mantendrán.

**Ejemplo:** Guardar un archivo satisfactoriamente en el sistema.

**Nivel de capacidad 2: Gestionado**

Un proceso de nivel de capacidad 2 se caracteriza cómo un "proceso gestionado". Un proceso gestionado es un proceso realizado (nivel de capacidad 1) que tiene la infraestructura básica dispuesta para soportar el proceso. Se planifica y ejecuta de acuerdo a políticas; emplea personal con habilidades; tiene los recursos adecuados para producir resultados controlados; involucra a las partes interesadas relevantes; se monitoriza, controla y revisa; y se evalúa la adherencia a su descripción del proceso. La disciplina de proceso reflejada por el nivel de capacidad 2 ayuda a asegurar que las prácticas existentes se mantienen durante tiempo de estrés.

**Ejemplo:** Guardar un archivo con un tipo de extensión especifica como por ejemplo .gif o .doc

**Nivel de capacidad 3: Definido**

Un proceso de nivel de capacidad 3 se caracteriza cómo un "proceso definido". Un proceso definido es un proceso gestionado (nivel de capacidad 2) que se adapta a partir de un conjunto de procesos estándar de la organización, de acuerdo a las guías de adaptación de la organización, y contribuye a los activos de proceso de la organización con productos del trabajo, medidas e información adicional de mejora de procesos. Una distinción crítica entre los niveles de capacidad 2 y 3 es el alcance de los estándares, descripciones de proceso y procedimientos. En el nivel de capacidad 2, los estándares, descripciones de proceso y procedimientos pueden ser bastante diferentes en cada instancia específica del proceso. En el nivel de capacidad 3, los estándares, descripciones de proceso y procedimientos para un proyecto se adaptan a partir del conjunto de procesos estándar de la organización, para ajustarse a un proyecto o unidad organizativa particular, y son, por tanto, más consistentes, excepto para las diferencias permitidas por las guías de adaptación. Otra distinción crítica es que en el nivel de capacidad 3, los procesos se describen normalmente de forma más rigurosa que en el nivel de capacidad 2. Un proceso definido establece claramente el propósito, las entradas, criterios de entrada, actividades, roles, medidas, etapas de verificación, salidas y criterios de salida. En el nivel de capacidad 3, los procesos se gestionan de forma más proactiva utilizando una comprensión de las interrelaciones de las actividades del proceso y de las medidas detalladas del proceso, de sus productos del trabajo y de sus servicios.

**Ejemplo:** Los procesos gestionados o controlados en el administrador de tareas.

**Nivel de capacidad 4: Gestionado cuantitativamente**

Un proceso de nivel de capacidad 4 se caracteriza como un "proceso gestionado cuantitativamente". Un proceso gestionado cuantitativamente es un proceso definido (nivel de capacidad 3) que se controla utilizando técnicas estadísticas y otras técnicas cuantitativas. Se establecen los objetivos cuantitativos de calidad y de ejecución del proceso, y se utilizan cómo criterios para gestionar el proceso. Se comprende la calidad y el rendimiento del proceso en términos estadísticos y se gestionan a lo largo de la vida del proceso.

**Ejemplo:** Un proceso que calcula el tamaño de un archivo.

**Nivel de capacidad 5: En optimización**

Un proceso de nivel de capacidad 5 se caracteriza cómo un "proceso en optimización". Un proceso en optimización es un proceso gestionado cuantitativamente (nivel de capacidad 4) que se mejora en base a una comprensión de las causas comunes de variación inherentes al proceso. El enfoque de un proceso en optimización es mejorar continuamente el rango de la ejecución del proceso mediante mejoras, tanto incrementales cómo innovadoras.

**Ejemplo:** El proceso que mejora un archivo ya sea su calidad en la imagen o reducir su tamaño.

* Realizar un mapa conceptual de la estructura del modelo de calidad CMMI

**MAPA CONCEPTUAL MODELO DE CALIDAD CMMI**

**Control del proceso**

**Satisface metas específicas**

**Proceso poco controlado**

**Nivel 2:**

**Gestionado**

**Nivel 1:**

**Realizado**

**Nivel 0:**

**Incompleto**

**Proceso para la organización**

**Proceso controlado**

**Mejora continua del proceso**

**Nivel 5:**

**En optimización**

**Nivel 4:**

**Gestionado cuantitativamente**

**Nivel 3:**

**Definido**

* Describir las ventajas y desventajas del modelo de calidad CMMI.

**Ventajas**

La gran ventaja de CMMI es que ha demostrado ser una metodología de gran eficacia, que ha permitido mejoras de gran impacto en procesos de desarrollo de productos software, tales como:

* Reducción del coste de desarrollo.
* Localización y resolución de defectos.
* Mejora en la fiabilidad de la planificación, en términos de dedicación y de calendario.
* Aumento de la productividad.
* Reducción de los trabajos derivados de correcciones tras las fases de pruebas.
* Aumento de la efectividad sobre la planificación realizada.
* Mejora en la calidad de producto.
* Reducción del número de defectos y detección en las fases tempranas de su ciclo de vida.
* Mejora de la Imagen de Marca.

**Desventajas**

El problema de CMMI es su falta de adecuación al enfoque a servicio que está experimentando el sector de las TI (procesos de desarrollo de productos de software) en todas sus líneas de actividad, así como el alto esfuerzo de implantación que exige.

**2) Realizar una presentación explicando el modelo de calidad CMMI**

**\***Ver archivo adjunto **Modelo de calidad CMMI.pptx**

**3) Investigar las métricas de calidad de software**

Las Métricas de Calidad proporcionan una indicación de cómo se ajusta el software, a los requerimientos implícitos y explícitos del cliente.

El objetivo de la ingeniería del software es desarrollar y producir software de alta calidad.

Para lograr este objetivo, es fundamental aplicar métodos y herramientas efectivos dentro del contexto de un proceso maduro de desarrollo de software.

* **Corrección**

Es el grado en que el software cumple su función.

La medida más común es: **Defectos por KDLC**(miles de líneas de código)

* **Facilidad de mantenimiento**

Es la facilidad con la que se puede corregir un programa si se encuentra un error.

Se utilizan medidas indirectas como: **Tiempo Medio de cambio (TMC)**

Es decir, el tiempo que se tarda en:

* Analizar una petición.
* Diseñar una modificación.
* Implementar el cambio.
* Probar y realizar el cambio.
* **Integridad**

Mide la capacidad del software para resistir ataques. Se debe tener en cuenta los siguientes atributos:

**-Amenaza**

Es la probabilidad de que un ataque ocurra en un tiempo determinado.

**-Seguridad**

Es la probabilidad de que se pueda repeler el ataque de un tipo determinado.

Se define como: **Integridad = Σ [(1-amenaza) x (1-seguridad)]**

* **Facilidad de uso**

Mide la "amigabilidad " del software con el usuario final.

Se mide en función de:

* Habilidad intelectual o física para aprender el sistema.
* El tiempo requerido para hacer uso eficiente del sistema.
* Aumento de la productividad.
* Valoración subjetiva de la disposición de los usuarios hacia el sistema.

* **Eficacia de la eliminación de defectos**

La eficacia de la eliminación de defectos (EED), es una métrica que permite medir la habilidad de filtrar las actividades de la garantía de calidad y de control, ya que es aplicable a todas las actividades del marco de trabajo del proceso.

Se define de la siguiente forma: **EED = E / (E + D)**

|  |  |
| --- | --- |
| E | Número de errores encontrados antes de la entrega del software |
| D | Número de defectos encontrados después de la entrega |

El valor ideal de EED es 1. No se han encontrado defectos en el software.

**4) Definir calidad**

Según la norma ISO 9000, la **calidad** es el “grado en el que un conjunto de características inherentes cumple con los requisitos”, entendiéndose por **requisito** “necesidad o expectativa establecida, generalmente implícita u obligatoria”.

**5) Definir medida, medición y métrica**

‣ Medida: número o categoría asignada a un atributo de una entidad mediante una medición [ISO 14598 - 1:1999].

‣ Medición: Actividad que usa la definición de la métrica para producir el valor de una medida.

‣ Métrica: es el método de medición definido y la definido y la escala de medición [ISO 14598 n [ISO 14598 -1:1999]. 1:1999].

**6) ¿Que son las métricas de software?**

Podemos definir las Métricas de Software o Medidas de Software como:

La aplicación continúa de técnicas basadas en las medidas de los procesos de desarrollo de Software y sus productos, para producir una información de gestión significativa y a tiempo. Esta información se utilizará para mejorar esos procesos y los productos que se obtienen de ellos.

Las Métricas de Software implican medir: medir involucra números; el uso de números para hacer cosas mejor. Las Métricas de Software pretenden mejorar los procesos de desarrollo de Software y mejorar, por tanto, todos los aspectos de la gestión de aquellos procesos.

Estas medidas son aplicables a todo el ciclo de vida del desarrollo, desde la iniciación, cuando debemos estimar los costos, al seguimiento y control de la fiabilidad de los productos finales, y a la forma en que los productos cambian a través del tiempo debido a la aplicación de mejoras.

Las medidas del Software y los modelos de medida son entonces útiles para estimar y predecir costos y para medir la productividad y la calidad del producto. Un ingeniero del Software recopila medidas y desarrolla métricas para obtener indicadores.

**REFERENCIAS**

<http://www.sei.cmu.edu/library/assets/cmmi-dev-v12-spanish.pdf>

<http://www.slideshare.net/linuxtro/introduccin-niveles-y-evaluacin-cmmi>

<http://www.codejobs.biz/es/blog/2012/09/19/los-niveles-de-capacidad-de-cmmi#sthash.fIOoW8PO.dpbs>

<http://www.ciw.cl/recursos/Charla_Metricas_Indicadores.pdf>