



2. Métricas SW

Calidad de software ll

Brenda Juárez Santiago
PROFESORA
Quinto Cuatrimestre



Contenido

DIAGRAMA DE ASIGNATURA	3
CAPITULO 2. Métricas de software.....	4
Ventajas del uso de métricas.....	5
Características de las métricas.....	5
2.1 Clasificación de las métricas de Software.....	6

DIAGRAMA DE ASIGNATURA

CALIDAD EN EL DESARROLLO DE SOFTWARE

MODULOS

1) Introducción a la calidad en el desarrollo de software

2) Métricas de software

3) Proceso personal de desarrollo de software (PSP).

4) Técnicas de estimación

5) Modelos para el aseguramiento de la calidad de software

SUBTEMAS

Generalidades de la calidad

Concepto de calidad en el desarrollo de software

Métricas de software

Tipos de métricas de calidad de software

Proceso personal de desarrollo de software

Plantillas PSP

Puntos de función

Puntos de caso de uso

Diagrama de clases

MOPROSOFT

CMMI

CAPITULO 2. Métricas de software

Michael ['99] define las métricas de software como “La aplicación continua de mediciones basadas en técnicas para el proceso de desarrollo del software y sus productos para suministrar información relevante a tiempo, así el administrador junto con el empleo de estas técnicas mejorará el proceso y sus productos”. Las métricas de software proveen la información necesaria para la toma de decisiones técnicas. Se ilustra una extensión de esta definición para incluir los servicios relacionados al software como la respuesta a los resultados del cliente:

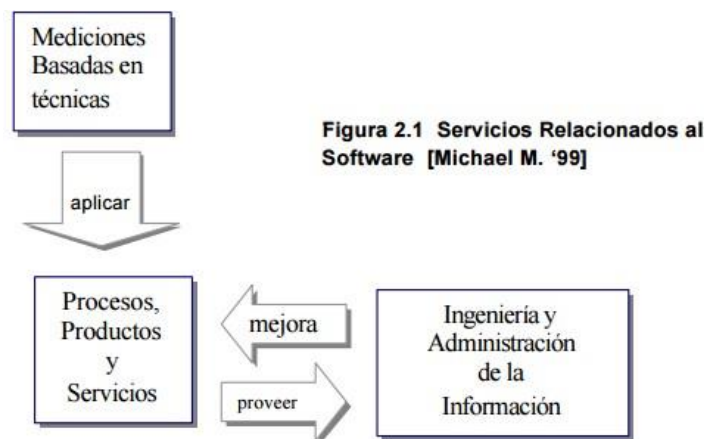


Figura 2.1 Diagrama de métricas

Las métricas son la maduración de una disciplina, que, según Pressman ['98] van a ayudar a la evaluación de los modelos de análisis y de diseño, en donde proporcionarán una indicación de la complejidad de diseños procedimentales y de código fuente, y ayudaran en el diseño de pruebas más efectivas; Es por eso que propone un proceso de medición, el cual se puede caracterizar por cinco actividades:

- (1) Formulación: La obtención de medidas y métricas del software apropiadas para la representación de software en cuestión.
- (2) Colección: El mecanismo empleado para acumular datos necesarios para obtener las métricas formuladas.

- (3) Análisis: El cálculo de las métricas y la aplicación de herramientas matemáticas.
- (4) Interpretación: La evaluación de los resultados de las métricas en un esfuerzo por conseguir una visión interna de la calidad de la representación.
- (5) Realimentación: Recomendaciones obtenidas de la interpretación de métricas técnicas transmitidas al equipo de software.

(Catarina)



Ventajas del uso de métricas

1. Determinar la calidad del producto.
2. Evaluar la productividad de los desarrolladores.
3. Conocimiento cuantitativo de las características del proceso y del producto.
4. Se podrán realizar comparaciones con otros proyectos.
5. Se podrá mejorar el producto ya que las métricas sirven para detectar defectos.

Características de las métricas

1. Exactas
2. Precisas: No se debe perder información en los redondeos ya que la información se desvirtúa.
3. Consistentes: Una medición de un atributo debe dar el mismo valor independientemente de la medición.
4. Comparables: Para ello, debe estar normalizada.

(Marisol Viramontes Aguilar, s.f.)



2.1 Clasificación de las métricas de Software

Según los criterios:

Complejidad: Métricas que definen la medición de la complejidad: volumen, tamaño, anidaciones, y configuración.

Ejemplo:

Costos que tendrá el desarrollo de software. (Estimación)

Calidad: Métricas que definen la calidad del software: exactitud, estructuración o modularidad, pruebas, mantenimiento.

Ejemplo:

Puntos críticos en el diseño.

Competencia: Métricas que intentan valorar o medir las actividades de productividad de los programadores con respecto a su certeza, rapidez, eficiencia y competencia.

Ejemplo:

El ahorrar líneas de código en el desarrollo de un software.

Desempeño: Métricas que miden la conducta de módulos y sistemas de un software, bajo la supervisión del SO o hardware.

Ejemplo:

Eficiencia de ejecución, tiempo, almacenamiento, complejidad de algoritmos computacionales.

Estilizadas: Métricas de experimentación y de preferencia: estilo de código, convenciones, limitaciones, etc.

Ejemplo:

Estilo de código, Identación, las convenciones denominando de datos, las limitaciones.

Según el contexto en que se aplican:

Métricas de proceso: Se recopilan de todos los proyectos, y durante un largo periodo de tiempo caracterizados por:

- Control y ejecución del proyecto.
- Medición de tiempos de las fases.

Ejemplo:

Qué tan eficiente, el proceso de desarrollo está aplicado en términos de puntos de revisión y en objetivos de calidad en el proceso.

Métricas de proyecto: Permiten evaluar el estado del proyecto. Permiten seguir la pista de los riesgos.

Ejemplo:

Identificar cada uno de los puntos de riesgo que pueda tener en el desarrollo.

Métricas de producto: Se centran en las características del software y no en cómo fue producido.

Ejemplo:

- También son productos los artefactos, documentos, modelos, y componentes que conforman el software.
- Se miden cosas como el tamaño, la calidad, la totalidad, la volatilidad, y el esfuerzo.
- *(Betzabeth Pereira, s.f.)*

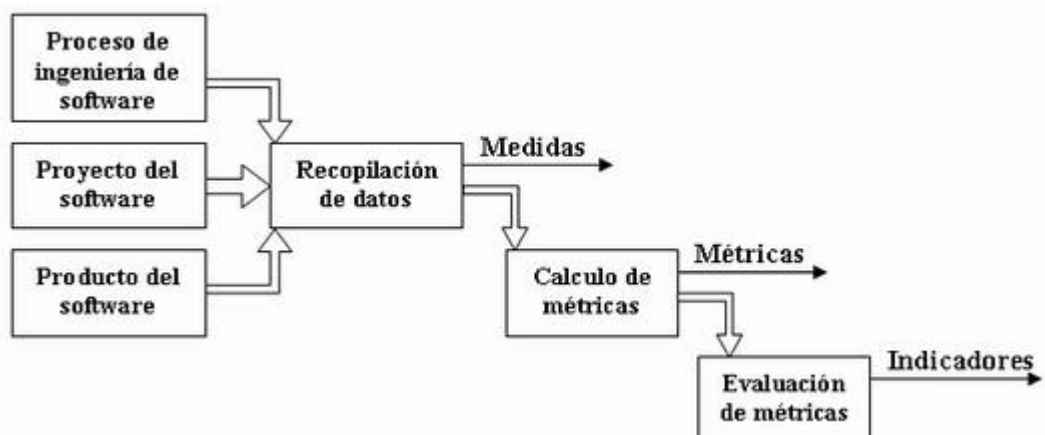


Figura 2.2 Diagrama de aplicación de métricas