

Universidad Autónoma del Estado de Hidalgo

Escuela Superior Huejutla





Área Académica: Licenciatura Sistemas Computacionales

Materia: Gestión de Proyectos

Profesor: Lic. Virginia Arguelles Pascual

Periodo: Julio-Diciembre

Tema: El proceso de software y métricas del proyecto.





Resumen

El presente trabajo esta enfocado a la unidad II de la materia Gestión de Proyectos donde podemos ubicar temas como medidas métricas e indicadores tanto al software como al proyecto.





Topic: Software Process and Project Metric.

Abstract

This work is focused on unit 2 project management matter where we can locate and indicators action items as much as to the software project.

Keywords: Software Metrics, Project





Introducción

La medición es fundamental para cualquier disciplina de ingeniería, y la ingeniería del software no es una excepción. La medición nos permite tener una visión más profunda proporcionando un mecanismo para la evaluación objetiva. Lord Kelvin en una ocasión dijo:

Cuando pueda medir lo que está diciendo y expresarlo con números, ya conoce algo sobre ello; cuando no pueda medir, cuando no pueda expresar lo que dice con números, su conocimiento es precario y deficiente: puede ser el comienzo del conocimiento, pero en sus pensamientos, apenas está avanzando hacia el escenario de la ciencia.





Las *métricas del software* se refieren a un amplio elenco de mediciones para el software de computadora. La medición se puede aplicar al proceso del software con el intento de mejorarlo sobre una base continua. Se puede utilizar en el proyecto del software para ayudar en la estimación, el control de calidad, la evaluación de productividad y el control de proyectos. Finalmente, el ingeniero de software puede utilizar la medición para ayudar a evaluar la calidad de los resultados de trabajos técnicos y para ayudar en la toma de decisiones táctica a medida que el proyecto evoluciona.

En el contexto de gestión de proyectos de software, existe una gran preocupación por las métricas de productividad y de calidad -medidas «de salida» (finalización) del desarrollo del software, basadas en el esfuerzo y tiempo empleados, y medidas de la «utilidad» del producto obtenido.





Park, Goethert y Florac tratan en su guía de la medición del software las razones por las que medimos:

Hay cuatro
Razones para
medir los
procesos del
software, los
productos y
los recursos:

Caracterizar

Evaluar

Predecir

Mejorar





Caracterizamos para comprender mejor los procesos, los productos, los recursos y los entornos y para establecer las líneas base para las comparaciones con evaluaciones futuras.

Evaluamos para determinar el estado con respecto al diseño. Las medidas utilizadas son los sensores que nos permiten conocer cuándo nuestros proyectos y nuestros procesos están perdiendo la pista, de modo que podamos ponerlos bajo control.



Predecimos para poder planificar. Realizar mediciones para la predicción implica aumentar la comprensión de las relaciones entre los procesos y los productos y la construcción de modelos de estas relaciones, por lo que los valores que observamos para algunos atributos pueden ser utilizados para predecir otros.

Hacemos esto porque queremos establecer objetivos alcanzables para el coste, planificación, y calidad de manera que se puedan aplicar los recursos apropiados.



Medimos para **mejorar** cuando recogemos la información cuantitativa que nos ayuda a identificar obstáculos, problemas de raíz, ineficiencias y otras oportunidades para mejorar la calidad del producto y el rendimiento del proceso.





Medidas métricas e indicadores

Aunque los términos *medida*, *medición* y *métricas* se utilizan a menudo indistintamente, es importante destacar las diferencias sutiles entre ellos. Como los términos «medida» y «medición» se pueden utilizar como un nombre o como un verbo, las definiciones de estos términos se pueden confundir.

Dentro del contexto de la ingeniería del software, una *medida* proporciona una indicación cuantitativa de la extensión, cantidad, dimensiones, capacidad o tamaño de algunos atributos de un proceso o producto.

La *medición* es el acto de determinar una medida.





Un ingeniero del software recopila medidas y desarrolla métricas para obtener indicadores.

Un *indicador* es una métrica o una combinación de métricas que proporcionan una visión profunda del proceso del software, del proyecto de software o del producto en sí.

Un indicador proporciona una visión profunda que permite al gestor de proyectos o a los ingenieros de software ajustar el producto, el proyecto o el proceso para que las cosas salgan mejor.





Métricas en el proceso y dominio del proyecto.

La medición es algo común en el mundo de la ingeniería.

Se mide el consumo de energía, el peso, las dimensiones físicas, la temperatura, el voltaje, la relación señal-ruido..., la lista es casi interminable. Por desgracia, la medición es mucho menos común en el mundo de la ingeniería del software.

Existen problemas para ponerse de acuerdo sobre qué medir y las medidas de evaluación de problemas recopilados.





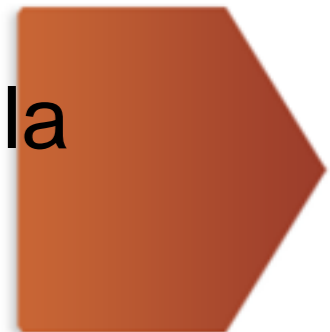
Se deberían recopilar métricas para que los indicadores del proceso y del producto puedan ser ciertos. **Los *indicadores de proceso*** permiten a una organización de ingeniería del software tener una visión profunda de la eficacia de un proceso ya existente (por ejemplo: el paradigma, las tareas de ingeniería del software, productos de trabajo e hitos).

También permiten que los gestores evalúen lo que funciona y lo que no. Las métricas del proceso se recopilan de todos los proyectos y durante un largo período de tiempo. Su intento es proporcionar indicadores que lleven a mejoras de los procesos de software a largo plazo.





- Los *indicadores de proyecto* permiten al gestor de proyectos del software:
- (1) Evaluar el estado del proyecto en curso;
 - (2) Seguir la pista de los riesgos potenciales;
 - (3) Detectar las áreas de problemas antes de que se conviertan en «críticas»;
 - (4) Ajustar el flujo y las tareas del trabajo, y
 - (5) Evaluar la habilidad del equipo del proyecto en controlar la calidad de los productos de trabajo del software.





En algunos casos, se pueden utilizar las mismas métricas del software para determinar tanto el proyecto como los indicadores del proceso. En realidad, las medidas que recopila un equipo de proyecto y las convierte en métricas para utilizarse durante un proyecto también pueden transmitirse a los que tienen la responsabilidad de mejorar el proceso del software. Por esta razón, se utilizan muchas de las mismas métricas tanto en el dominio del proceso como en el del proyecto.





Mediciones del software

Entre las medidas directas del proceso de la ingeniería del software se incluyen el coste y el esfuerzo aplicados. Entre las medidas directas del producto se incluyen las líneas de código (LDC) producidas, velocidad de ejecución, tamaño de memoria, y los defectos informados durante un período de tiempo establecido. Entre las medidas indirectas se incluyen la funcionalidad, calidad, complejidad, eficiencia, fiabilidad, facilidad de mantenimiento y muchas otras «capacidades».



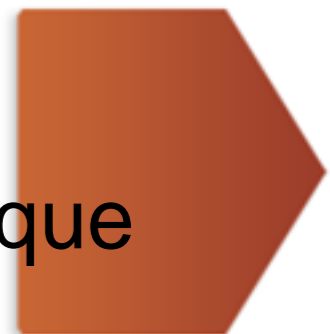


El dominio de las métricas del software se divide en: **métricas de proceso, proyecto y producto.**

También se acaba de destacar que las métricas de **producto** que son privadas para un individuo a menudo se combinan para desarrollar métricas del **proyecto** que sean públicas para un equipo de software.

Las métricas del proyecto se consolidan para crear métricas de **proceso** que sean públicas para toda la organización del software.

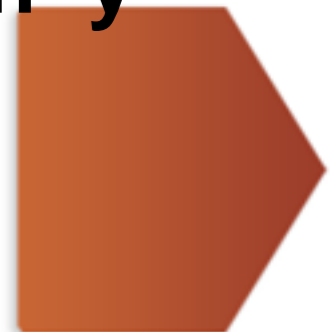
Pero ¿cómo combina una organización métricas que provengan de particulares o proyectos?





Métricas orientadas al tamaño

Las métricas del software orientadas al tamaño provienen de la normalización de las medidas de calidad y/o productividad considerando el «tamaño» del software que se haya producido. Si una organización de software mantiene registros sencillos, se puede crear una tabla de datos orientados al tamaño, como la que muestra la Figura 4.4. La tabla lista cada proyecto de desarrollo de software de los últimos años y las medidas correspondientes de cada proyecto. Debe tenerse en cuenta que el esfuerzo y el coste registrados en la tabla incluyen todas las actividades de ingeniería del software (**análisis, diseño, codificación y prueba**) y no sólo la codificación.





Proyecto	LDC	Esfuerzo	Coste £(000)	Pag. Doc.	Errores	Defectos	Personas
Alfa	12,100	24	168	365	134	29	3
Beta	27,200	62	440	1224	321	86	5
Gamma	20,200	43	314	1050	256	64	6

FIGURA 4.4. Métricas orientadas al tamaño.



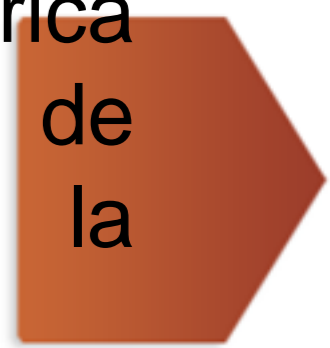


Métricas orientadas a la función

Las métricas del software orientadas a la función utilizan una medida de la funcionalidad entregada por la aplicación como un valor de normalización. Ya que la «funcionalidad» no se puede medir directamente, se debe derivar indirectamente mediante otras medidas directas.

Las métricas orientadas a la función fueron propuestas por primera vez por Albretch, quien sugirió una medida llamada *punto defunción*.

Los puntos de función se derivan con una relación empírica según las medidas contables (directas) del dominio de información del software y las evaluaciones de la complejidad del software.

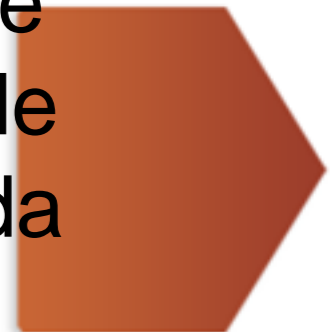




Número de entradas de usuario. Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas se deberían diferenciar de las peticiones, las cuales se cuentan de forma separada.

Número de salidas de usuario. Se cuenta cada salida que proporciona al usuario información orientada a la aplicación. En este contexto la salida se refiere a informes, pantallas, mensajes de error, etc.

Número de peticiones de usuario. Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva.





Número de archivos. Se cuenta cada archivo maestro lógico (esto es, un grupo lógico de datos que puede ser una parte de una gran base de datos o un archivo independiente).

Número de interfaces externas. Se cuentan todas las interfaces legibles por la máquina (por ejemplo: archivos de datos de cinta o disco) que se utilizan para transmitir información a otro sistema.

Una vez que se han recopilado los datos anteriores, a la cuenta se asocia un valor de complejidad. Para calcular puntos de función (PF), se utiliza la (4.1) relación siguiente:






Parámetros de medición	Cuenta	Factor de ponderación			=	[]
		Simple	Medio	Complejo		
Número de entradas de usuario	[]	x 3	4	6	=	[]
Número de salidas de usuario	[]	x 4	5	7	=	[]
Número de peticiones de usuario	[]	x 3	4	6	=	[]
Número de archivos	[]	x 7	10	15	=	[]
Número de interfaces externas	[]	x 5	7	10	=	[]
Cuenta total						[]

FIGURA 4.5. Cálculo de puntos de función.

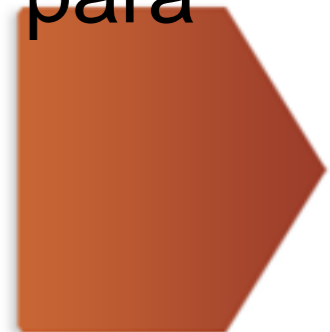




Métricas para la calidad del software

Para lograr este objetivo, los ingenieros del software deben aplicar métodos efectivos junto con herramientas modernas dentro del contexto de un proceso maduro de desarrollo de software. Además, un buen ingeniero del software (y buenos gestores de la ingeniería del software) deben medir si la alta calidad se va a llevar a cabo.

La calidad de un sistema, aplicación o producto es tan bueno como los requisitos que describen el problema, el diseño que modela la solución, el código que conduce a un programa ejecutable, y las pruebas que ejercitan el software para detectar errores.





Bibliografía:

- Roger S. Pressman . Ingeniería del software. Un enfoque practico. Sexta edición. Mc Graw Hill.

