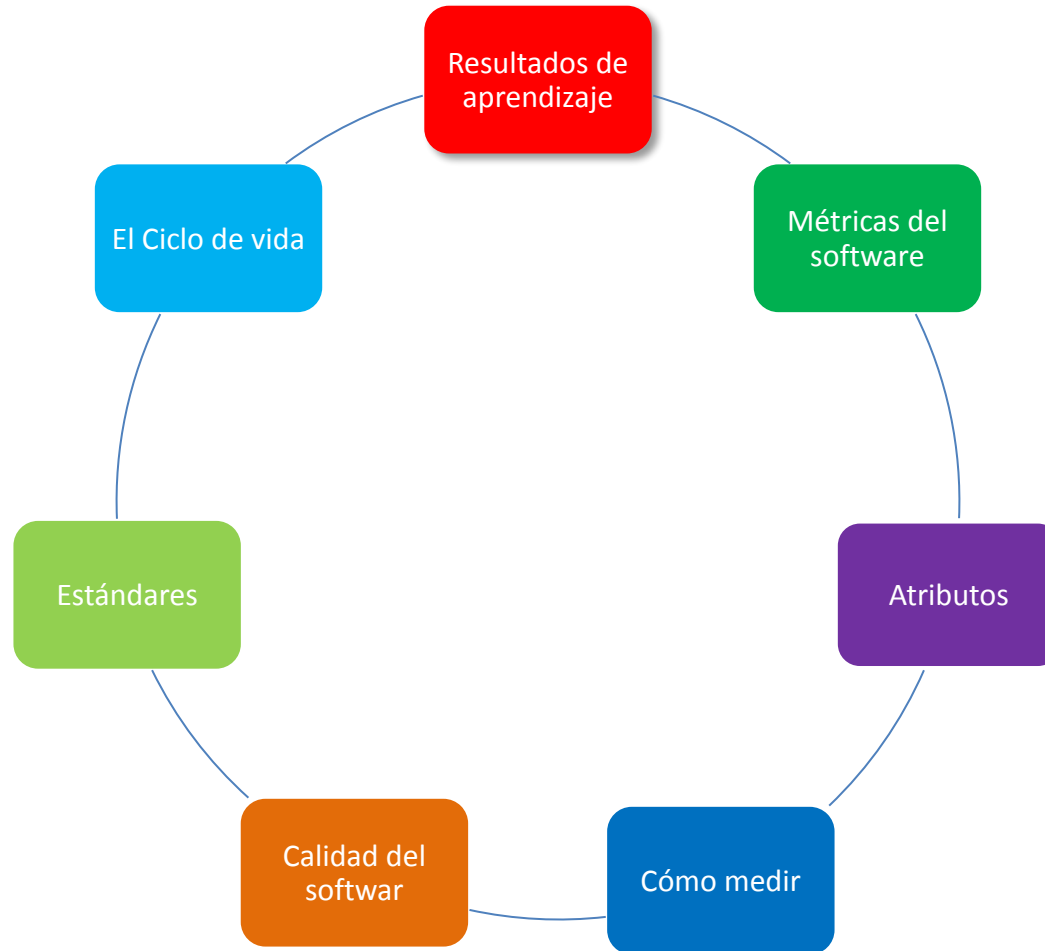


Análisis de Sistemas

CIF 5555

2022-1



CALIDAD DEL SOFTWARE

Resultados de aprendizaje

- El alumno debería:
 - Aplicar el proceso de Ingeniería de Requisitos para la elicitación de las necesidades de los clientes y usuarios

Medir el software

“No se puede controlar aquello que no se puede medir”

- Ejemplos:
 - ¿Cuánto cuesta un determinado proceso?
 - ¿Cual es la productividad de los programadores?
 - ¿Cuán “adecuado” es el código producido?
 - ¿Cuántos programadores se necesitan?
 - ¿Cuánto tiempo se prevé para el desarrollo?
 - ¿Cuán portable es el software producido?

Métricas

- Se usa el término “métrica” para indicar una medida directa o indirecta de cualquier atributo de una entidad de interés.
- Las métricas de software se refieren a actividades de medición pertenecientes a:
 - Medidas y modelos de productividad
 - Estimación de los costos y del esfuerzo
 - Medidas y modelos de confiabilidad
 - Medidas de complejidad
- i.e., **Calidad**

Ejemplos de “métricas”

- Formales \Leftrightarrow informales (subjetivas)
- Confiabilidad
 - Informal: cuanto se cuenta sobre el producto
 - Formal: probabilidad de ausencia de malfuncionamientos en un período de tiempo
- Portabilidad
 - Informal: cuanto es fácil portar el producto
 - Formal: $1 - \text{NEstadNuevos} / \text{NEstadTotal}$

Medición

- Medición: el proceso de asignación de símbolos, normalmente números, para representar un atributo de la entidad de interés, según reglas definidas
- Cuatro elementos:
 - Entidad: el objeto o el evento sobre el que se indaga (una mesa, un viaje..)
 - Atributo: característica del objeto (altura, duración, costo, ..)
 - Forma (de la representación): La altura se mide en centímetros; la ropa con S (“small”), M (“medium”), L (“large”), XL (“extra large”); la bencina con “93”, “95” y “97”.
 - Reglas: para llegar a determinar el valor del atributo (de manera de hacer el proceso repetible, no subjetivo)
- El metro ha sido definido en 1889 (en París está el metro patrón)

Medición (cont.)

- La altura de una persona se mide en cm.
- ¿Cómo se mide la inteligencia?
- Del vino se puede medir la graduación alcohólica, pero su sabor, ¿cómo se mide?
- La exactitud de una medida depende del instrumento y de la forma de la medida (cm, QI)
- Medida directa: cuantificación directa (e.g., medida de la altura)
- Medida indirecta: requiere un cálculo (e.g., medida de la inteligencia)

Medida indirecta

“Hacer medible aquello que no es directamente medible”

- Se hacen medidas directas y se combinan para dar una cuantificación de un atributo no directamente medible.
- Ejemplo: en el decatlón se miden tiempos y longitudes; las medidas son luego combinadas y pesadas de manera de dar lugar a un “puntaje” que identifica el “mejor atleta”

Atributos (y métricas relacionadas)

- Externos (visibles al usuario)
 - Rendimiento (tiempo de respuesta)
 - Costo (\$)
 - Confiabilidad (MTBF, tiempo promedio entre fallas)
 - Usabilidad (???)
- Internos (visibles al diseñador/desarrollador)
 - Dimensión (SLOC, líneas de código)
 - Esfuerzo de producción (MU, unidad de gestión)
 - Funcionalidad (FP, puntos de función)
 - Complejidad (Num ciclomático)
 - Modularidad (???)

La dimensión del código

- La manera más sencilla consiste en medir el número de líneas de código (SLOC).
 - Fácil de medir (salvo ponerse de acuerdo sobre aquello que se mide)
- Se considera que esta medida sencilla sea insuficiente respecto a la cuantificación de esfuerzo, productividad y costo.
 - (la altura no permite de decir si una persona tiene también sobrepeso)

Líneas de código (fuente)

- Es la medida software más antigua (años setenta).
- Gran confusión: Es necesario establecer cómo se cuentan:
 - líneas vacías
 - líneas de comentario
 - declaraciones y otros comandos
 - líneas con más instrucciones
 - líneas programadas o generadas por herramientas
 - copiadas de otra parte
- Normalmente en SLOC entran todas las líneas de programa excepto las líneas vacías y las líneas de comentario
- Si derivan otras medidas

Autodocumentacion NC_SLOC
SLOC

Modelo Constructivo de Costos COCOMO

$$E = aS^bF$$

- E: Esfuerzo (meses/persona)
- S: Dimensión (miles de líneas de código entregadas, KDSI)
- a, b: constantes que varían según el modelo y en base a la modalidad de desarrollo
- F: factor correctivo.

$$T = cE^d$$

- T: tiempo requerido en meses necesario al desarrollo
- c, d: constantes que varían según el modelo y en base a la modalidad de desarrollo

COCOMO

- La cantidad de líneas de código incluye todas las instrucciones fuente entregadas al usuario, excluyendo eventuales instrucciones usadas a propósito para el “debugging”, “testing”, u otro soporte
- Excluye todas las líneas de comentario
- Esfuerzo y tiempo cubren el período intermedio del inicio del análisis del proyecto (hasta la recolección de los requisitos del usuario) hasta la fase de entrega del sistema
- Los valores calculados comprenden las actividades de “management” y la redacción de la documentación, pero excluyen los tiempos de entrenamiento del usuario, los tiempos de instalación y eventual manutención
- La estimación comprende todos los costos directos (programadores, librerías, instrumentos de soporte). Son excluidos los costos indirectos (costos secretaria, costos “top management”)

SLOC no es el mejor modo para medir la prestación

- Sin embargo, proporciona una métrica reveladora de la evolución de un programa informático
- A medida que ha crecido la importancia de la electrónica y el software, también lo ha hecho la complejidad
- Tomemos como ejemplo la explosión del número de líneas de código de software (SLOC) que contienen los automóviles modernos:
 - 2016 150 M
 - 2010 10 M

<https://mondaynote.com/code-on-wheels-a-4715926b2a2>

Calidad de un producto

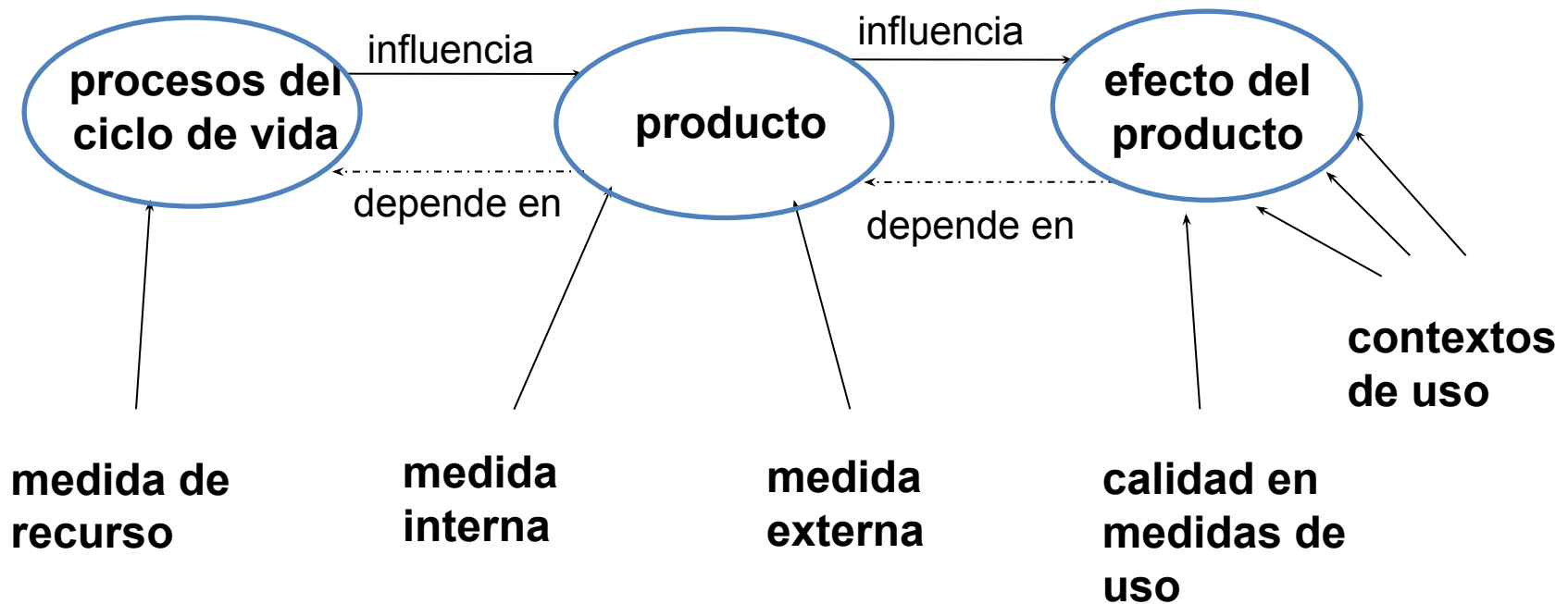
- Su adecuación al uso
- ISO 9000:2015 – “Fundamentals and Vocabulary (definitions)” cita:

“El adjetivo calidad se aplica a los objetos y se refiere al grado en que un conjunto de características inherentes cumple una serie de requisitos.”

Un *objeto* es cualquier entidad concebible o perceptible y una característica inherente es un rasgo que existe en un objeto.”

Calidad del producto software

Calidad del Proceso Calidad del Producto Calidad en el uso



ISO/IEC 9126 → SQuaRE = ISO/IEC 25000

Calidad del proceso software

- Productividad
 - eficiencia del proceso (# líneas/día)
- Estructura
 - procesos/métodos de desarrollo bien codificados
- Puntualidad (temporal)
 - respeto de los tiempos

Calidad del software

- Calidad subjetiva
 - Satisfacción de los requisitos
- Calidad objetiva
 - Rendimiento
 - Funcionalidad
 - Documentación

Calidad del software (cont.)

Percepción de la calidad

- Usuario: Facilidad de uso, eficacia
- Cliente: Solución de un problema a costo aceptable
- Desarrollador: Facilidad de comprensión, manutención, reutilización de código
- Gestor: retorno de la venta

Índices distintos en ámbitos distintos

- Sistemas Informáticos: integridad de los datos, seguridad de accesos
- Sistemas de Tiempo real: tiempo medio de respuesta, latencia máxima

Calidad del software (cont.)

- Interna - Externa
 - Interna: interesa al desarrollador
 - Externa: interesa al usuario
 - La calidad interna tiene influencia sobre la calidad externa
- La calidad del proceso tiene influencia sobre la calidad del producto
- De un buen proceso se espera un buen producto

Estándar de calidad

- ISO 9000: Estándar internacional de calidad
- ISO 9001: Calidad del proceso (genérico)
- ISO SQuaRE, ISO/IEC 25000 : Estándar de calidad del producto software
 - define 8 características de calidad:
Funcionalidad apropiada, Fiabilidad,
Usabilidad, Eficiencia en desempeño,
Mantenibilidad, Compatibilidad, Seguridad

El ciclo de vida (proceso software)

De IEEE Std 1074-2006

- Un ciclo de vida es un período de tiempo que:
 - inicia cuando un producto software es concebido
 - termina cuando el producto software non es más usado
- Contexto organizativo para el desarrollo de un proyecto

De ISO/IEC/IEEE 12207:2017

- Un marco de referencia que contiene
 - los procesos,
 - las actividades y las tareas involucradas en el desarrollo,
 - la explotación y el mantenimiento de un producto de software,
- abarcando la vida del sistema desde la definición de los requisitos hasta la finalización de su uso