**FACTORES DE CALIDAD DEL SOFTWARE**

**GARVIN**

Funcionalidad: Grado en el que el software satisface las necesidades planteadas según las establecen los atributos siguientes, adaptabilidad, exactitud, interoperabilidad, cumplimiento y seguridad.

Confiabilidad: ¿El software proporciona todas las características y capacidades sin fallar?

¿Está disponible cuando se necesita? ¿Entrega funcionalidad libre de errores?

Confiabilidad: Grado en el que se espera que un programa cumpla con su función y con la precisión requerida

Durabilidad: ¿El software puede recibir mantenimiento (cambiar) o corregirse (depurarse)

sin la generación inadvertida de eventos colaterales? ¿Los cambios ocasionarán que la tasa

de errores o la confiabilidad disminuyan con el tiempo?

Calidad del desempeño: ¿El software entrega todo el contenido, las funciones y las características especificadas como parte del modelo de requerimientos, de manera que da valor al usuario final?

Calidad de las características: ¿El software tiene características que sorprenden y agra-

dan la primera vez que lo emplean los usuarios finales?

Conformidad: ¿El software concuerda con los estándares locales y externos que son relevantes para la aplicación? ¿Concuerda con el diseño de facto

y las convenciones de código?

Por ejemplo, ¿la interfaz de usuario está de acuerdo con las reglas aceptadas del diseño

para la selección de menú o para la entrada de datos?

Durabilidad: ¿El software puede recibir mantenimiento (cambiar) o corregirse (depurarse)

sin la generación inadvertida de eventos colaterales? ¿Los cambios ocasionarán que la tasa de errores o la confiabilidad disminuyan con el tiempo?

Servicio: ¿Existe la posibilidad de que el software reciba mantenimiento (cambios) o correcciones (depuración) en un periodo de tiempo aceptablemente breve? ¿El equipo de apoyo puede adquirir toda la información necesaria para hacer cambios o corregir defectos? Douglas Adams [Ada93] hace un comentario irónico que parece pertinente: “La diferencia entre algo que puede salir mal y algo que posiblemente no salga mal es que cuando esto último sale mal, por lo general es imposible corregirlo o repararlo.

Estética: No hay duda de que todos tenemos una visión diferente y muy subjetiva de lo que es estético. Aun así, la mayoría de nosotros estaría de acuerdo en que una entidad estética posee cierta elegancia, un flujo único y una “presencia” obvia que es difícil de cuantificar y que, no obstante, resulta evidente. El software estético tiene estas características.

Percepción: En ciertas situaciones, existen prejuicios que influirán en la percepción de la

calidad por parte del usuario. Por ejemplo, si se introduce un producto de software elaborado por un proveedor que en el pasado ha demostrado mala calidad, se estará receloso y

la percepción de la calidad del producto tendrá influencia negativa. De manera similar, si

un vendedor tiene una reputación excelente se percibirá buena calidad, aun si ésta en realidad no existe.

Robustez: capacidad de un programa para ejecutar diversos procesos de manera simultanea sin generar fallos o bloquearse.

Compatibilidad: Capacidad de dos o más sistemas o componentes para intercambiar información y/o llevar a cabo sus funciones requeridas cuando comparten el mismo entorno hardware o software. Esta característica se subdivide a su vez en las siguientes su característica

Facilidad de uso: La interfaz amigable del sistema y el orden lógico en el que se encuentran los menús favorecen el uso intuitivo del software.

Integridad: salvaguarda la precisión y completitud de la información y sus métodos de proceso

MCCALL

Corrección: Grado en el que un programa satisface sus especificaciones y en el que cumple con los objetivos de la misión del cliente

Confiabilidad: Grado en el que se espera que un programa cumpla con su función y con la precisión

Requerida

Eficiencia: Cantidad de recursos de cómputo y de código requeridos por un programa para llevar acabo su función.

Integridad: Grado en el que es posible controlar el acceso de personas no autorizadas al software o a

los datos.

Usabilidad: Esfuerzo que se requiere para aprender, operar, preparar las entradas e interpretar las

salidas de un programa.

Facilidad de recibir mantenimiento: Esfuerzo requerido para detectar y corregir un error en un programa

Flexibilidad: Esfuerzo necesario para modificar un programa que ya opera.

Susceptibilidad de someterse a pruebas.

Esfuerzo que se requiere para probar un programa a fin de

garantizar que realiza la función que se pretende

Portabilidad: Esfuerzo que se necesita para transferir el programa de un ambiente de sistema de hardware o software a otro.

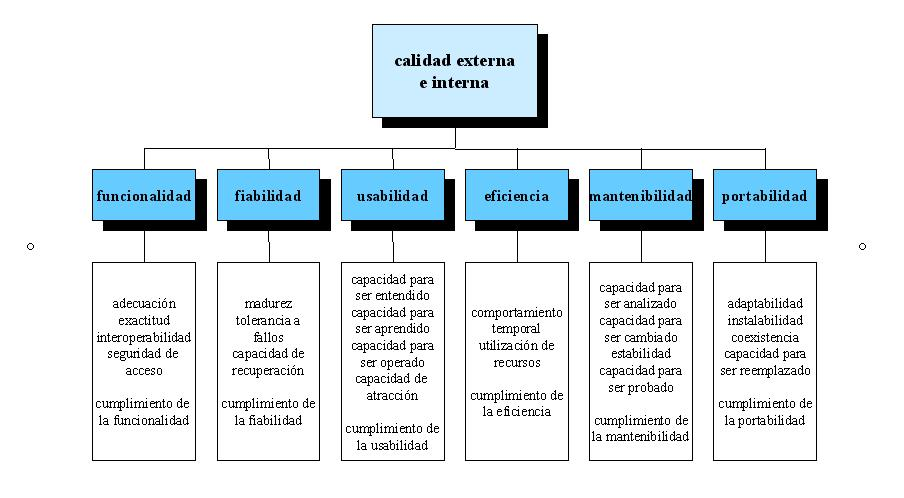
Reusabilidad: Grado en el que un programa (o partes de uno) pueden volverse a utilizar en otras aplicaciones (se relaciona con el empaque y el alcance de las funciones que lleva a cabo el programa).

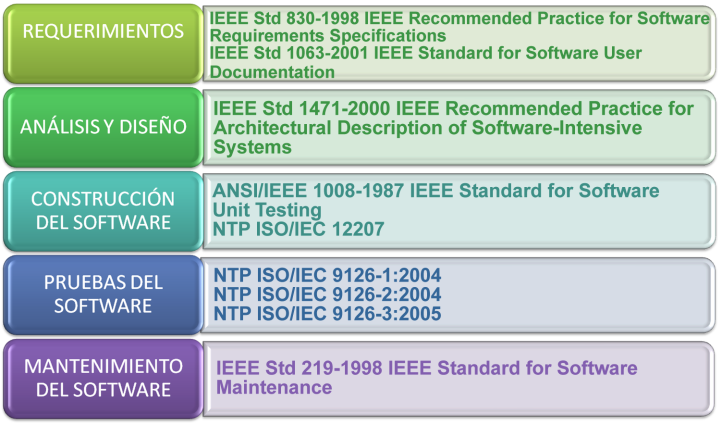
Interoperabilidad: Esfuerzo requerido para acoplar un sistema con otro

ISO ESTANDAR 9126

Y

Atributos





Tecnologías de la Información y Comunicación

Materia: Calidad en el desarrollo de Software

Practica Factores que determinan la calidad del software

Relacione la columna derecha (numero) con el inciso correspondiente de la columna izquierda.

(4 ) ¿Podré usarlo en otra máquina? 1. Facilidad de mantenimiento

( 6 ) ¿Es seguro? 2. Interoperabilidad

( 1 ) ¿Puedo corregirlo? 3. Facilidad de prueba

( 8 ) ¿Puedo cambiarlo? 4. Portabilidad

( 5 ) ¿Hace lo que quiero? 5. Corrección

( 9 ) ¿Está diseñado para ser usado? 6. Integridad

( 10 )¿Se ejecutara en mi hardware lo mejor posible? 7. Reusabilidad

( 3 ) ¿Puedo hacer pruebas? 8. Flexibilidad

( 2 ) ¿Podré hacerlo interactuar con otro sistema? 9. Eficiencia

( 7 ) ¿Podré reutilizar alguna parte del software? 10. Facilidad de uso