



Universidad  
del Cauca®

**CALIDAD DE SOFTWARE**  
**Mag. Carlos Alberto Ardila Albarracín**

**CAPÍTULO 5**  
**MEDICIÓN EN EL PROCESO DE DESARROLLO DE SOFTWARE**  
**5.3. Medida de factores de Calidad**

Los factores que afectan a la calidad del software se pueden categorizar en dos amplios grupos:

- (1) Factores que se pueden medir directamente (defectos por punto de función).
- (2) Factores que se pueden medir sólo indirectamente (por ejemplo, facilidad de uso o de mantenimiento).

En todos los casos debe aparecer la medición.

Debemos comparar el software (documentos, programas, datos) con una referencia y llegar a una conclusión sobre la calidad.

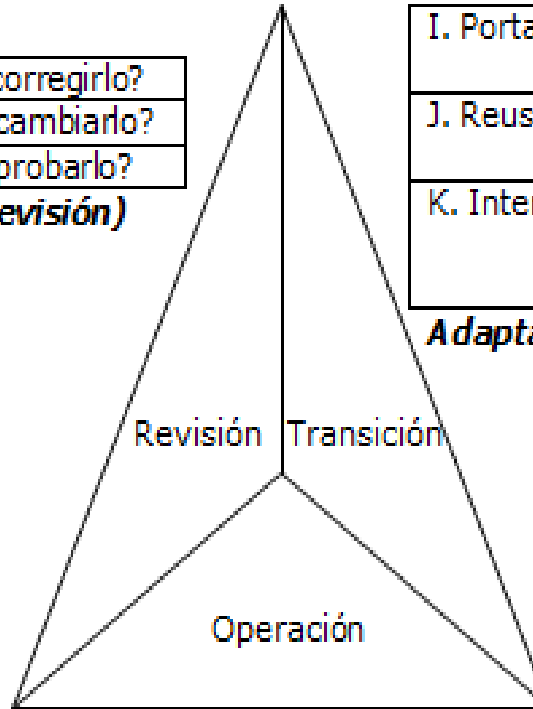
### Ejemplo. Modelo de McCall.

F. Facilidad de mantenimiento.	¿Puedo corregirlo?
G. Flexibilidad.	¿Puedo cambiarlo?
H. Facilidad de prueba.	¿Puedo probarlo?

**Capacidad de soportar los cambios (Revisión)**

I. Portabilidad.	¿Podré usarlo en otra máquina?
J. Reusabilidad.	¿Podré reutilizar alguna parte del software?
K. Interoperabilidad.	¿Podré hacerlo interactuar con otro sistema?

**Adaptabilidad a nuevos entornos (Transición)**



A. Corrección.	¿Hace lo que quiero?
B. Fiabilidad.	¿Lo hace de forma fiable todo el tiempo?
C. Eficiencia.	¿Se ejecutará en mi hardware lo mejor que pueda?
D. Seguridad (Integridad).	¿Es seguro?
E. Usabilidad (Facilidad de uso).	¿Está diseñado para ser usado?

**Características operativas (Operación)**

### Ejemplo. Modelo de McCall.

Es difícil, y en algunos casos imposible, desarrollar medidas directas de los factores de calidad anteriores.

Por tanto, se definen y emplean un conjunto de criterios para desarrollar expresiones para todos los factores, de acuerdo con la siguiente relación:

$$\mathbf{QF} = (c_1 * m_1) + (c_2 * m_2) + (c_3 * m_3) + \dots + (c_n * m_n)$$

Donde:

**QF** es un factor de calidad del software,

**c<sub>n</sub>** son coeficientes de regresión (  $c_1 + c_2 + \dots + c_n = 1.0$  )

**m**, son los valores de los criterios que afectan al factor de calidad.

### **Ejemplo. Modelo de McCall.**

El problema es que muchas de los criterios definidas por McCall pueden medirse solamente de manera subjetiva. Las métricas pueden ir en forma de lista de comprobación que se emplea para "puntuar" atributos específicos del software.

El esquema de puntuación propuesto por McCall es una escala del 0 (bajo) al 10 (alto).

Se emplean las siguientes criterios en el esquema de puntuación:

### Ejemplo. Modelo de McCall.

**Facilidad de auditoría.** La facilidad con la que se puede comprobar el cumplimiento de los estándares.

**Exactitud.** La exactitud de los cálculos y del control.

**Estandarización de comunicaciones.** El grado de empleo de estándares de interfaces y protocolos.

**Compleción.** El grado con que se ha logrado la implementación total de una función.

**Complejidad.** Valoración del grado de complejidad del componente / función.

**Concisión.** Lo compacto que es el programa en términos de líneas de código.

**Consistencia.** El empleo de un diseño uniforme y de técnicas de documentación a lo largo del proyecto de desarrollo.

**Estandarización de datos.** El empleo de estructuras y tipos de datos estándares a lo largo del programa.

**Tolerancia al error.** El daño causado cuando un programa encuentra un error.

**Eficiencia de ejecución.** El rendimiento del funcionamiento de un programa.

**Capacidad de expansión.** El grado con que se pueden ampliar el diseño arquitectónico, de datos o procedimental.

### Ejemplo. Modelo de McCall.

**Generalidad.** La amplitud de aplicación potencial de los componentes del programa.

**Independencia del hardware.** El grado con que se desacopla el software del hardware donde opera.

**Instrumentación.** El grado con que el programa vigila su propio funcionamiento e identifica los errores que ocurren.

**Modularidad.** La independencia funcional de componentes de programa.

**Operatividad.** La facilidad de operación de un programa.

**Seguridad.** La disponibilidad de mecanismos que controlan o protegen los programas y los datos.

**Autodocumentación.** El grado en que el código fuente proporciona documentación significativa.

**Simplicidad.** El grado de facilidad con que se puede entender un programa.

**Independencia del sistema software.** El grado de independencia de programa respecto a las características del lenguaje de programación no estándar, características del sistema operativo y otras restricciones del entorno.

**Trazabilidad.** Capacidad de seguir una representación del diseño o un componente real del programa hasta los requisitos.

**Formación (Entrenamiento).** El grado en que ayuda el software a manejar el sistema a los nuevos usuarios.

### 5.3. Medida de factores de Calidad

Factor de calidad \ Criterio	A. Corrección	B. Fiabilidad	C. Eficiencia	D. Seguridad (Integridad)	E. Usabilidad	F. Facilidad de Mantenimiento	G. Flexibilidad	H. Facilidad de Pruebas	I. Portabilidad	J. Reusabilidad	K. Interoperatividad
01. Facilidad de auditoría (FA)				X				X			
02. Exactitud (EX)		X									
03. Estandarización de comunicaciones (EC)											X
04. Compleción (CM)	X										
05. Complejidad (CX)		X					X	X			
06. Concisión (CN)			X			X	X				
07. Consistencia (CS)	X	X				X	X				
08. Estandarización de datos (ED)											X
09. Tolerancia al error (TE)		X									
10. Eficiencia de ejecución (EE)			X								
11. Capacidad de expansión (CE)							X				
12. Generalidad (GE)							X		X	X	X
13. Independencia del hardware (IH)									X	X	
14. Instrumentación (IN)				X		X		X			
15. Modularidad (MD)		X				X	X	X	X	X	X
16. Operatividad (OP)			X		X						
17. Seguridad (SG)				X							
18. Autodocumentación (AD)						X	X	X	X	X	
19. Simplicidad (SM)		X				X	X	X	X		
20. Independencia del sistema software (IS)										X	
21. Trazabilidad (TZ)	X										
22. Formación (Entrenamiento) (FM)					X						

Relación entre Criterios y Factores de calidad según McCall



### Ejemplo. Modelo de McCall.

Las relaciones entre los factores de calidad y los criterios son las siguientes:

**A. Corrección** =  $(C1 * CM) + (C2 * CS) + (C3 * TZ)$

**B. Fiabilidad** =  $(C1 * EX) + (C2 * CX) + (C3 * CS) + (C4 * TE) + (C5 * MD) + (C6 * SM)$

**C. Eficiencia** =  $(C1 * CN) + (C2 * EE) + (C3 * OP)$

**D. Seguridad (Integridad)** =  $(C1 * FA) + (C2 * IN) + (C3 * SG)$

**E. Usabilidad (Facilidad de uso)** =  $(C1 * OP) + (C2 * FM)$

**F. Facilidad de mantenimiento** =  $(C1 * CN) + (C2 * CS) + (C3 * IN) + (C4 * MD) + (C5 * AD) + (C6 * SM)$

**G. Flexibilidad** =  $(C1 * CX) + (C2 * CN) + (C3 * CS) + (C4 * CE) + (C5 * GE) + (C6 * MD) + (C7 * AD) + (C8 * SM)$

**H. Facilidad de Pruebas** =  $(C1 * FA) + (C2 * CX) + (C3 * IN) + (C4 * MD) + (C5 * AD) + (C6 * SM)$

**I. Portabilidad** =  $(C1 * GE) + (C2 * IH) + (C3 * MD) + (C4 * AD) + (C5 * SM)$

**J. Reusabilidad** =  $(C1 * GE) + (C2 * IH) + (C3 * MD) + (C4 * AD) + (C5 * IS)$

**K. Interoperatividad** =  $(C1 * EC) + (C2 * ED) + (C3 * GE) + (C4 * MD)$

-----FIN DEL DOCUMENTO