

MODELO DE CALIDAD
ALGORITMO PARA OBTENCIÓN DEL NIVEL DE CALIDAD ALCANZADO

Características y Subcaracterísticas Utilizadas:

1. Funcionabilidad

- a. Seguridad de Acceso
- b. Exactitud de los resultados

2. Eficiencia

- a. Utilización de recursos
- b. Comportamiento frente al tiempo

3. Fiabilidad

- a. Tolerancia a fallos
- b. Capacidad de recuperación de errores

4. Mantenibilidad

- a. Capacidad del código de ser analizado
- b. Capacidad del código de ser cambiado
- c. Estabilidad

5. Usabilidad

- a. Capacidad de ser entendido
- b. Capacidad de ser operado
- c. Capacidad de ser atractivo para el usuario

6. Portabilidad

- a. Adaptabilidad
- b. Instalabilidad

Niveles de Puntuación

El algoritmo de calidad consiste en evaluar cada subcaracterística como mala, regular o buena, dependiendo de los criterios que se mencionarán más adelante. Las combinaciones de los resultados de la evaluación de las subcaracterísticas determinarán el puntaje que se obtendrá de la característica. El paso siguiente es promediar los puntajes anteriores para obtener un valor general de la característica. Cada una de ellas podría tener un peso que se utilizaría en un promedio general ponderado de las 6 características del modelo de calidad.

El promedio general obtenido determinaría el nivel de calidad obtenido que podría ser SATISFACTORIO o NO SATISFACTORIO.

Criterios de Evaluación de Métricas

1. Funcionabilidad

Seguridad de acceso

Capacidad del producto software para asegurar la integridad de los datos y la confidencialidad de estos.

Características por medir:

Encriptación de datos

Inicio de sesión de usuarios

Evaluación

Mala [0] No cumple con alguna característica.

Regular [1] Cumple con 1 característica.

Buena [2] Cumple con 2 características.

Exactitud de resultados

Es la capacidad del producto software para proporcionar los resultados con el grado necesario de precisión.

Evaluación

Mala [$\geq 10^{-3}$] Los resultados tienen un error del orden de 10^{-3} o superior.

Regular [10^{-4} ; 10^{-6}] Los resultados tienen un error del orden entre 10^{-4} y 10^{-6} .

Buena [$\leq 10^{-7}$] Los resultados tienen un error del orden de 10^{-7} o inferior

2. Eficiencia

Utilización de recursos

Se evaluará la eficiencia del producto software de acuerdo con el porcentaje de uso de procesador que realice.

Evaluación

Mala [41;100] 41% o más de uso de procesador.

Regular [11;40] 11% a 40% de uso de procesador.

Buena [0;10] 10% o menos de uso de procesador.

Comportamiento en el tiempo

Se evaluará el tiempo que está el producto software sin informarle al usuario del estado en que se encuentra la solicitud que realizó.

Evaluación

Mala [≥ 5] El producto está 5 o más segundos sin informar al usuario del estado de la solicitud.

Regular [2;4] El producto está entre 2 y 4 segundos sin informar al usuario del estado de la solicitud.

Buena [0;1] El producto está menos de 1 segundo sin informar al usuario del estado de la solicitud.

3. Fiabilidad**Tolerancia a fallos**

Es la capacidad del producto software de mantener la integridad de los datos cuando se producen fallas del sistema.

Características por medir:

Cuando sucede un error se protegen los datos procesados.

Se realiza un log de actividades que el sistema estaba haciendo.

Evaluación

Mala [0] No cumple con alguna característica.

Regular [1] Cumple con 1 característica.

Buena [2] Cumple con 2 características.

Capacidad de recuperación de errores

Es la capacidad del sistema de reanudar sus actividades cuando se producen errores críticos.

Características por medir:

El sistema reanuda las actividades si se produce una falla crítica.

Reanuda sus actividades y vuelve al estado en que estaba.

Evaluación

Mala [0] No cumple con ninguna característica.

Regular [1] Cumple con 1 característica.

Buena [2] Cumple con 2 características.

4. Mantenibilidad**Capacidad del código para ser analizado**

Para evaluar la capacidad que tiene el código para ser analizado se tiene en cuenta el porcentaje de comentarios que posee el código por cada método y en general.

Evaluación

Mala [0;14] 14% o menos del código comentado.

Regular [15;29] Entre 15 y 29% del código comentado.

Buena [≥ 30] 30% o más del código comentado

Capacidad del código para ser cambiado

Para evaluar la capacidad que tiene el código para ser cambiado se tomará en cuenta la complejidad ciclomática del método.

Evaluación

Mala [21] La complejidad ciclomática es mayor o igual a 21.

Regular [11;20] La complejidad ciclomática es entre 11 y 20.

Buena [1;10] La complejidad ciclomática es menor o igual a 10.

Estabilidad

Para determinar la estabilidad del software se evalúa el promedio de fallas que presenta el producto por prueba.

Evaluación

Mala [5] El software presenta un promedio 5 o más errores por prueba.

Regular [2;4] El software presenta un promedio entre 2 y 4 errores por prueba.

Buena [0;1] El software presenta un promedio entre 0 y 1 error por prueba.

5. Usabilidad**Capacidad de ser entendido**

Capacidad que posee el software, para ayudar a los usuarios ante una determinada situación donde se necesite asistencia.

Características por medir:

Posee ayuda contextual sobre menús y botones de acción.

Manual de usuario incorporado al sistema como un menú dedicado.

Evaluación

Mala [0] No cumple con alguna característica.

Regular [1] Cumple con 1 característica.

Buena [2] Cumple con 2 características.

Capacidad para ser operado

Es la Capacidad del producto software de ser utilizado sin asistencia adicional. Se valúa qué requiere el usuario para operar correctamente el producto.

Evaluación

Mala [1] El usuario requiere consultar a personal especializado para operar el producto software.

Regular [2] El usuario requiere ayuda contextual y manual de uso para operar el producto software.

Buena [3] El usuario opera el producto software sin asistencia.

Capacidad de ser atractivo para el usuario

Es la agrupación correcta de funcionalidad del producto software en su interfaz gráfica, desde su agrupación lógica hasta el número promedio de pasos para alcanzar una función o contenido específico.

Evaluación

Mala [6] 6 o más pasos promedio sin organización de categoría.

Regular [3;5] Entre 3 y 5 pasos promedio y distribuidos en categorías.

Buena [1;2] 1 o 2 pasos promedio y distribuidos en categorías.

6. Portabilidad**Adaptabilidad**

Es la capacidad del producto software de adaptarse a diferentes sistemas operativos sin cambiar su estructura interna.

Evaluación

Mala [1] Compatible con 1 sistema operativo.

Regular [2] Compatible con 2 sistemas operativos.

Buena [≥ 3] Compatible con 3 o más sistemas operativos.

Instalabilidad

El producto software debe poder ser instalado en una cantidad mínima de pasos.

Evaluación

Mala [>7] El producto se instala en 7 o más pasos.

Regular [4;6] El producto se instala entre 4 y 6 pasos.

Buena [1;3] El producto se instala en 3 o menos pasos.

Criterios de aceptación

CARACTERÍSTICA	SUB CARACTERÍSTICA	CALIFICACIÓN
Funcionalidad	Seguridad de acceso	Buena
	Exactitud de los resultados	Buena
Eficiencia	Utilización de los recursos	Buena
	Comportamiento frente al tiempo	Buena
Fiabilidad	Tolerancia a fallos	Regular
	Capacidad de recuperación de errores	Mala
Mantenibilidad	Capacidad del código de ser analizado	Buena
	Capacidad del código de ser cambiado	Buena
	Estabilidad	Buena
Usabilidad	Capacidad de ser entendido	Mala
	Capacidad de ser operado	Buena
	Capacidad de ser atractivo para el usuario	Regular
Portabilidad	Adaptabilidad	Regular
	Instalabilidad	Buena
Total		Buena