**INSTITUTO TECNOLÓGICO SUPERIOR YAVIRAC**

****

**INTEGRANTES:**

* **Cajilema Armando**
* **Cruz Kenny**
* **López Christian**
* **Ostaiza José**

**ASIGNATURA:** [**Fundamentos De Software Engineering**](http://eva.yavirac.edu.ec/mod/forum/view.php?id=18480)

**TEMA: Modelo de Calidad McCall**

**FECHA:24/06/2020**

**INTRODUCCIÓN**

La finalidad de todo software es satisfacer las necesidades del usuario. El uso de aplicaciones de software dentro de las organizaciones se ha convertido en una necesidad para el buen funcionamiento y desarrollo de las mismas. Frecuentemente estas organizaciones se enfrentan a la liberación de software que no cumplen totalmente con sus necesidades. Lo cual puede ir desde el incumplimiento de los requerimientos establecidos hasta la facilidad de uso. Debido a esto surge la necesidad de que el software pase por una evaluación detallada con el fin de mejorar la calidad del producto final y satisfacer las necesidades del usuario. El proceso de medición de la calidad del software (CS) debe implementarse en todo el ciclo de vida del mismo, pero es importante destacar la evaluación del producto terminado, ya que el tener un alto nivel de calidad en el proceso de desarrollo, no garantiza productos finales de calidad.

Actualmente, existen diferentes modelos de calidad que indican las cualidades deseables para determinar la calidad de un producto de software. Sin embargo, tales modelos no establecen mecanismos definidos para su evaluación, por lo que la presente investigación propone integrar y definir, en base a estos modelos, los principales factores que se pueden analizar y la forma de evaluarlos, considerando principalmente la integración de los usuarios finales en dicha evaluación, todo esto para contribuir a lograr aplicaciones de software de alta calidad.

En este sentido se han realizado muchos trabajos que establecen propuestas de los factores cualitativos que influyen en la calidad del software, entre ellos el modelo de McCall.

McCall se focaliza en el producto final, identificando atributos claves desde el punto de vista del usuario, estos atributos se denominan factores de calidad y son normalmente atributos externos, pero también se incluye algunos atributos posiblemente internos.

**DESARROLLO:**

Es uno de los modelos de calidad más conocidos que definen factores cualitativos que afectan la calidad del software, este modelo define factores, criterios, y métricas con estructuras jerárquicas, donde los elementos de nivel superior son mucho más abstractos que los del nivel inferior que son más específicos y pueden ser medidos con facilidad.

El modelo de McCall fue el primero en ser presentado en el 1977 y se originó motivado por Air Forcé y Dod. Este modelo se focaliza en el producto final identificando atributo claves desde el punto de vista del Cliente. Esto atributos se denominan factores de calidad y son normalmente atributos externos, pero también se incluyen algunos atributos internos.

Cada atributo externo atributo se dominan factores de calidad los cuales son abstractos para ser medidos directamente por lo cual se introduce un atributo de bajo nivel denominado criterios de calidad. Según McCall algunos criterios de calidad son atributos internos que tienen efectos directos en atributos externos.

El modelo de McCall organiza los factores en tres ejes o puntos de vista desde los cuales el usuario puede contemplar la calidad de un producto, basándose en once factores de calidad organizados en torno a los tres ejes y a su vez cada factor se desglosa en criterios de calidad.

El modelo de McCall se centra en tres aspectos importantes de un producto de software:

* Sus características operativas/Operación del Producto
* Su capacidad para soportar los cambios/Revisión del Producto
* Su adaptabilidad a nuevos entornos/Transición del producto

**Lista de factores:**

**Operación del Producto**

* Corrección: mide el grado en que un programa satisface sus especificaciones y consigue los objetivos del usuario.
* Fiabilidad: mide el grado en que se puede esperar que un programa lleve a cabo sus funciones esperada con la precisión requerida.
* Eficiencia: mide la cantidad de recursos de computadora y de código requerido por un programa para que lleve a cabo las funciones especificadas.
* Integridad: es el grado en que puede controlarse el acceso al software o a los datos por personal no autorizado.
* Facilidad de Uso: es el esfuerzo requerido para aprender un programa e interpretar la información de entrada y de salida.

**Revisión del Producto**

* + Facilidad de Mantenimiento: es el esfuerzo requerido para localizar y arreglar programas.
  + Facilidad de Prueba: es el esfuerzo requerido para probar un programa.
  + Flexibilidad: es el esfuerzo requerido para modificar un sistema operativo.

**Transición del Producto**

* + Portabilidad: es el esfuerzo requerido para transferir un software de un hardware o un entorno de sistemas a otro.
  + Reusabilidad: es el grado en que un programa (o partes de un programa) se puede reutilizar en otro.
  + Facilidad de Interoperación: es el esfuerzo requerido para asociar un programa a otro.

**Ventajas del Modelo de Calidad McCall**

* Se enfoca en el modelo final identificando los atributos claves desde el punto de vista del usuario.
* Identifica una serie de criterios, tales como rastreabilidad, simplicidad, capacidad de expansión, etc.
* El factor de calidad es estándar (el mismo).
* Se puede reutilizar para realizar la evaluación de otros productos.

**Desventajas del Modelo de Calidad McCall**

* No siempre existe una relación perfectamente lineal entre los valores métricos y las características que se deben estimar
* Al ser fijo da a entender que todos los criterios de evaluación serán idénticos y suficientes para evaluar todo tipo producto.

**CONCLUSIONES:**

Existen varios modelos para medir la calidad de un software y en el modelo de McCall se ha basado principalmente en la operación transición y revisión que abarca de una manera general los estándares de calidad con la diferencia que en este modelo lo clasifica de la manera que el usuario se le pueda facilitar la compresión y respuesta ante sus preguntas de la siguiente manera:

**OPERACIÓN**

Corrección: ¿El software hace lo que necesito?

Fiabilidad: ¿Lo Hace de forma exacta siempre?

Eficiencia: ¿Aprovecha el hardware lo mejor posible?

Integridad: ¿Es seguro?

Facilidad de Uso: ¿Lo puedo usar con facilidad?

**REVISIÓN**

Facilidad de prueba: ¿Se puede probar?

Flexibilidad: ¿Se puede modificar?

Facilidad de mantenimiento: ¿Que tan difícil es arreglarlo?

**TRANSICIÓN**

Portabilidad: ¿Se puede usar en otra maquina?

Reusabilidad: ¿Se puede reutilizar parte del Software?

Interoperabilidad: ¿Se puede comunicar con otros sistemas?

Por tal motivo podemos decir que todos los modelos pueden servir para establecer una idea clara de un software que cumpla con los parámetros fundamentales para ser de calidad, pero la diferencia seria que cada uno cambiaria el orden la clasificación y medición de la misma además cada modelo se adaptaría de acuerdo a Alos requerimientos del usuario, en el caso del modelo McCall se apegaría más a las interrogantes del cliente.

BILIOGRAFIA:

Yeimmy Roa. (17 de Julio de 2018). Modelo de McCall (Archivo de vídeo). Recuperado de: <https://www.youtube.com/watch?v=5qX4j6ox5ro>

Salvador J. (2014). Modelo McCall. Recuperado de: <https://prezi.com/cndj3aiz8koa/modelo-mccall/>

Bahamon J. (2010). Control de calidad en el software. Recuperado de: <https://repository.icesi.edu.co/biblioteca_digital/bitstream/item/4008/1/Control_calidad_software.pdf>