

LA CALIDAD DEL SOFTWARE, NUEVAS TÉCNICAS Y TECNOLOGÍAS EN LA ERA DE LA GLOBALIZACIÓN

PROF. ALONSO A. DE LA CRUZ S.
PROF. MERLY, CABALLERO
PROF. MARLENE PINTO

RESUMEN

Objetivo: conocer las bondades de software como medio de calidad en el aprovechamiento de recurso e información.

Introducción: El crecimiento que estamos viviendo actualmente nos dirige a un enfoque competitivo, donde la calidad se convierte en necesidad. La tecnología lleva auestas una participación magistral en el desarrollo global, por lo que la calidad que participa en estos ámbitos se puede apreciar mucho a nivel de software y de hardware.

Nuestro punto principal es dirigiremos a la calidad de software, tomando en cuenta la sociedad de la información de hoy, vemos el auge que ha tenido.

Al hablar de calidad de software, nos orientamos a modelos de calidad, los cuales son una parte fundamental en los procesos de desarrollo y evaluación de un sistema.

Como todo en la vida, los sistemas desde lo más pequeños a los más grandes, ya sea, naturales o hechos por el hombre, se rigen por pasos, normas o guías de cómo y qué hacer, para resolver los problemas de forma sencilla.

Calidad en tecnología de Software

La palabra calidad data sus inicios en el término griego kalos, que significa lo bueno, lo apto, y también en la palabra latina qualitem, que significa cualidad o propiedad. En este sentido, calidad es una palabra de naturaleza subjetiva, una apreciación que cada individuo define según sus expectativas y experiencias, es un adjetivo que califica alguna acción, materia o individuo.

Ahora cuando hablamos de un modelo nos referimos a la representación de un objeto, sistema o idea, de forma diferente al de la entidad misma, cuyo fin es ayudarnos a explicar, entender o mejorar un sistema.

Partiendo de ambas definiciones, entonces, al hablar de Modelos de Calidad se define como: un conjunto herramientas que permiten a las Organizaciones guiarse de manera continua y competitiva al realizar mejoras al sistema obteniendo así productos y servicios de alto nivel.

¿Qué es modelo de calidad de Software?

Un software de calidad surge luego de aplicar el modelo correcto, de acuerdo al tipo de proyecto, que traer consigo los procedimientos y

estándares para cada una de las etapas de proceso, cumpliendo así con principios de calidad como son: confiabilidad, mantenibilidad, facilidad, productividad, tanto para la labor de desarrollo como para el control de la calidad del software.

Modelados de calidad de software según autores:

En las siguientes párrafos veremos algunos modelos de calidad los cuales podremos darnos cuenta que se descomponen jerárquicamente en una serie de características y sub-características, como principios, componentes, categorías, métricas y factores, las cuales son usadas como lista de comprobación de aspectos relacionados con la calidad.

El **modelo de McCall** fue el primero en ser presentado en 1977, busca reducir la brecha entre usuarios y desarrolladores enfocándose a factores de calidad. Es un modelo fijo que consiste en una estructuración de los factores en una jerarquía multinivel, con un conjunto de factores de más alto nivel, unos criterios que descomponen dichos factores, y eventualmente métricas para la medida de cada criterio.

En caso del **Modelo de Boehm**, su prioridad es que un sistema de software debe ser útil. Es el segundo modelo de calidad más conocido, presentado por Barry Boehm. Este modelo introduce características de alto nivel, características de nivel

intermedio y características primitivas, cada una de las cuales contribuye al nivel general de calidad. La Independencia y la autocontención definen la Portabilidad, como unos de sus principios, siguiendo la utilidad, que se refiere a la fiabilidad, eficiencia e ingeniería humana; y por último la mantenibilidad que con lleva a la prueba, entendibilidad y modificabilidad.

A continuación podemos apreciar la tabla de calidad entre los dos más grandes modelos de calidad:

CRITERIO	McCALL	BOEHM
Correctitud	+	+
Integridad	+	+
Eficiencia	+	+
Testeabilidad	+	
Flexibilidad	+	+
Portabilidad	+	+
Modificabilidad		+
Entendibilidad		+
Confiabilidad	+	+
Usabilidad	+	+
Mantenible	+	+
Interoperabilidad	+	
Reusabilidad	+	+
Claridad		+
Documentación		+
Validez		+

Tabla 1. Comparación McCALL-BOEHM

Nota. Fuente: Bautista y Chaico (2012). Modelos de Calidad de Software (p.23).

**Comparación de los dos grandes modelos de calidad de software.*

Procesos en la calidad del software

Existen procesos para la evaluación de sitios web, en este artículo podemos referirnos al modelo WebEQM Luis Olsina el cual **es** una metodología de evaluación de calidad de sitios Web (Web-site Quality Evaluation method), diseñada para la evaluación siguiendo seis fases: planificación y programación de la evaluación de calidad, definición y especificación de requerimientos de calidad, definición e implementación de la evaluación elemental, definición e implementación de la evaluación global, análisis de resultados, conclusión y documentación, validación de métricas (Olsina, 1999). Este modelo se centra más en la evaluación de aplicación de un sistema basándose en requerimientos definidos, bajo principios como Facilidad de Uso, Funcionalidad, Confiabilidad, Eficiencia.

Otro modelo es el modelo ISO/IEC 9126 nuevas técnicas y tecnologías, inclinado en el producto final, el software; su calidad, lo documentado. El objeto de este modelo es “promover un entorno que permita la evaluación de la calidad de software” desde una calidad interna y la calidad externa”.

El estándar 9126 propone un modelo de calidad que se divide en tres vistas: interior, exterior y de uso. Con

sub características, y estas a su vez se componen de atributos.

Factores	Atributos	Acrónimo
Funcionalidad (Functionality)	Características y Capacidad del Programa. Generalidad de las Funciones. Seguridad del Sistema.	F
Usabilidad (Usability)	Factores Humanos. Factores Estéticos. Consistencia de la Interfaz. Documentación.	U
Confiabilidad (Reliability)	Frecuencia y Severidad de las Fallas. Exactitud de las Salidas. Tiempo Medio de Fallos. Capacidad de Recuperación ante Fallas. Capacidad de Predicción.	R
Desempeño (Performance)	Tiempo de Respuesta. Consumo de Recursos. Rendimiento Efectivo Total.	P
Soportabilidad (Supportability)	Extensibilidad. Adaptabilidad. Capacidad de Pruebas. Capacidad de Configuración. Compatibilidad. Requisitos de Instalación	S

Tabla 2. Factores de calidad Modelo FURPS

Se refiere a La calidad de uso en que las instituciones pueden evaluar y comparar su capacidad para desarrollar, desplegar y apoyar la enseñanza virtual, basada en

metodologías de mejora de proceso del ámbito de ingeniería de software,

El Modelo Sistémico de Calidad de Software (MOSCA) LISIUSB enfoca su calidad tanto al proceso de desarrollo como al producto final obtenido, bajo dos dimensiones: Eficiencia y Efectividad. Este modelo integra los otros modelos como submodelos, ya que concluye que la calidad del proceso va de la mano en la calidad del producto y no pueden verse por separados.

eMM Modelo de Madurez de Enseñanza virtual se enfoca en la mejora de los procesos en los sistemas ya maduros, dedicado a la enseñanza virtual: Aprendizaje, Desarrollo, Soporte, Evaluación y organización. Se refiere a La calidad de uso en el que las instituciones pueden evaluar y comparar su capacidad para desarrollar, desplegar y apoyar la enseñanza virtual, basada en metodologías de mejora de proceso en el ámbito de la ingeniería del software, como los modelos de madurez CMM y SPICE.

Evaluación de la calidad de Software

La evaluación de la calidad se compone de dimensiones, Categorías, Características y Métricas.

En la categoría Eficacia y efectividad del producto participan características como Funcionalidad, Fiabilidad, Usabilidad, Eficiencia, Mantenibilidad Y Portabilidad.

En la dimensión proceso, las categorías Eficacia y efectividad, se caracteriza por Cliente proveedor, Ingeniería, Soporte, Gestor y Organizacional

Después de una revisión de modelos de calidad de software se presenta una alternativa en la gestión tradicional de proyectos con el Modelo ágil pueden ser considerados como modelos lógicos, ya sea diseños de sistemas, o modelos como prototipos.

El desarrollo de software ágil es un concepto usado en el desarrollo de software para describir las metodologías de desarrollo incrementales (Cohen, Lindvall & Costa, 2003).

El paradigma Ágil se concentra en la colaboración y tomar decisiones en equipo, además potencia la planificación continua, pruebas permanentes y la integración conjunta del código y los despliegues.

El concepto “Ágil” en este modelado se atribuye a las técnicas ágiles en el desarrollo iterativo, interacción, la comunicación y la reducción de artefactos.

Metodologías en la calidad de la Software

¿Qué es la metodología Ágil Extreme programming (XP)?

Es una Metodología que se centra en el desarrollo, en lugar de aspectos de gestión de proyectos. XP fue diseñada para que las organizaciones tuvieran libertad de adoptar la totalidad o parte de la metodología.

Los proyectos XP comienzan con una fase de planificación de entregables “release plan”, seguido de varias iteraciones, cada una de las cuales termina con pruebas de aceptación del usuario. Cuando el producto tiene suficientes características para satisfacer a los usuarios, el equipo termina la iteración y libera el software.

¿Qué es la metodología Ágil Scrum?

La metodología ágil más usada y extendida en el mundo es Scrum, propuesta por Ken Schwaber y Jeff Sutherland en 1995. Este paradigma para prototipados rápido, con bajo un entorno en que los requisitos se encuentran incompletos en el inicio y son cambiantes durante el desarrollo. Este modelo diferencia de XP, incluye tanto los procesos de gestión como de desarrollo.

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al solicitante del proyecto.

¿Qué es la metodología Lean Development?

La metodología Lean Development se concentra entonces en la reducción de las actividades sin valor agregado, pero también a través de perfeccionar el flujo y eliminado la sobrecarga.

Este paradigma es muy diferente al resto de metodologías, ya que es considerado como un proceso del ciclo de vida de desarrollo de software o a un proceso de administración de proyectos, siempre y cuando se comprueba que cumple los valores y principios establecidos para dicho concepto. Esto último hace que se tenga que ajustar el proceso de desarrollo de software, comprendiendo los principios y adoptando los valores básicos de Lean.

¿Qué es la metodología Kanban?

Kanban es una metodología se enfoca en mejorar la visibilidad del flujo de trabajo, limitar el trabajo en curso de acuerdo con la capacidad disponible y medir el tiempo de ciclo de vida de una actividad. Estas actividades se ubican en bloques de trabajo (tablero Kanban). Sin embargo cada bloque de trabajo tiene sus límites (umbral), para dar secuencia a otras actividades o acciones dentro de un software con características bases para un manejo óptimo y eficaz.

CONCLUSIONES

- Los modelos de Calidad de software permiten definir con mayor nivel de detalle los objetivos de calidad a alcanzar.
- Los objetivos de calidad se a través de una serie de actividades basadas en principios, categorías, características y métricas.
- La globalización de los TI, aumenta la importancia de la gestión de la calidad y de la adopción de técnicas de la calidad para desarrollo en la industria del software.
- La calidad del software es un conjunto complejo que no es directamente comparable con la calidad de la manufactura de productos. En la manufacturación, la calidad viene dada por la similitud entre el producto desarrollado y su especificación.

BIBLIOGRAFÍA

Marco teórico. (2019). Definición de calidad del software. Consultado el 5 de julio, 2019. Recuperado de: <https://www.marcoteorico.com/curso/91/ingenieria-de-software/866/definicion-de-calidad-y-calidad-del-software>

Marco teórico. (2019). Aseguramiento de la calidad del software. Consultado el 5 de julio, 2019. Recuperado de: <https://www.marcoteorico.com/curso/91/ingenieria-de-software/869/aseguramiento-de-la-calidad-del-software>

Marco teórico. (2019). Modelado de la calidad del software. Consultado el 5 de julio, 2019. Recuperado de:

<https://www.marcoteorico.com/curso/91/ingenieria-de-software/861/modelado>

Evaluando Software.com. (2018). ¿Qué es Desarrollo de software ágil?. Consultado el 4 de julio, 2019. Recuperado de:

<https://www.evaluandosoftware.com/desarrollo-software-agil/>

Callejas, Alarcón, Álvarez. (2017). Modelos de Calidad del software, un estado del arte. Consultado el 4 de julio, 2019. Recuperado de:

<https://www.redalyc.org/jatsRepo/2654/265452747018/html/index.html>

Cano, Fernández, Pagés. (2012). Modelo de Madurez de la enseñanza virtual. Consultado el 2 de julio, 2019. Recuperado de:

http://www.esvial.org/wp-content/files/Atica2012_pp101-109.pdf

Peñañiel, Mirian. (2018). Aportaciones para la mejora en la ingeniería virtual. Consultado el 30 de junio, 2019. Recuperado de:

<https://dialnet.unirioja.es/servlet/tesis?codigo=146839>

CORRESPONDENCIA

Alonso Abdiel De La Cruz S. adelacruz763@gmail.com
Julio, 2019.



Merly Caballero. merly_y@hotmail.com
Julio, 2019.



Marlene Pinto. madelis23@hotmail.com
Julio, 2019.

