

**REPÚBLICA BOLIVARIANA DE VENEZUELA
MINISTERIO DEL PODER POPULAR PARA LA EDUCACIÓN
UNIVERSITARIA, CIENCIA Y TECNOLOGÍA
UNIVERSIDAD POLITÉCNICA TERRITORIAL
DEL ESTADO BOLÍVAR
PROGRAMA NACIONAL DE FORMACIÓN
EN INFORMÁTICA
T4-INF-1.
GESTIÓN DE PROYECTOS INFORMÁTICOS**



ATRIBUTOS DE CALIDAD DE PRODUCTO INFORMÁTICO

Profesora:

Prof. Yosmely Pérez

Alumno:

Oscar Cedeño 20.137.156

Ciudad Bolívar, noviembre del 2022

Aseguramiento de la Calidad

La naturaleza cambiante en los requerimientos del software complica la precisión del equipo de desarrollo para entregar un producto funcional a tiempo, dentro del presupuesto aprobado y con el alcance solicitado por el cliente. Para mantener los objetivos de calidad determinados en un proyecto, es importante tener en cuenta que cualquier cambio en una de esas tres variables implica una modificación a las otras dos.

Las organizaciones que no aprenden a manejar la naturaleza cambiante en los requerimientos del software, no pueden establecer mecanismos eficientes para asegurar la calidad de sus proyectos.

El aseguramiento de la calidad del software es un área importante de proceso, que persigue evaluar la calidad del producto de software, en gran medida, mediante el establecimiento de un plan de aseguramiento de la calidad.

Este plan de aseguramiento de la calidad, dentro de su marco de revisión formal, establece llevar a cabo técnicas de evaluación estática de los componentes del software, y técnicas de evaluación dinámicas, las cuales consisten en una secuencia de pruebas que miden cada escenario potencial donde existe un mínimo riesgo de que el producto no cumpla las expectativas del cliente.

Los individuos altamente capacitados de un equipo de desarrollo, muchas veces tienen suficiente dominio, documentación, y además cuentan con las herramientas correctas para realizar las inspecciones del código fuente, hacer las pruebas de usabilidad, las pruebas unitarias, las pruebas de integración, las pruebas de aceptación, las pruebas de regresión, entre otras.

La realidad es que, aun así, los fracasos en la calidad de los proyectos conforme a las estadísticas que año tras año recopila The Standish Group Report Chaos (group, 2019), ascienden a un promedio del 20%. Esto implica que, de cada 100 proyectos registrados, 20 proyectos fallan en cumplir sus objetivos de calidad.

Una propuesta para que las organizaciones impacten positivamente sus proyectos, en miras de lograr sus objetivos de calidad determinados, es adoptar agilidad en su cultura organizacional.

La agilidad es un mindset de continua exploración, adaptación, aprendizaje y mejora, que a partir del desarrollo iterativo e incremental busca obtener el producto correcto con el mayor valor, realizado de la mejor forma posible, basado en la colaboración, confianza y motivación de las personas involucradas.

Al tratarse de un mindset organizacional, no existe una fórmula exacta para incorporarlo con éxito, sino que es el resultado de la aplicación del método empírico, donde solo a través de la exploración de las prácticas de metodologías ágiles, y la adaptación de dichas prácticas a su realidad organizacional, puede encontrarse la alineación entre la excelencia técnica y la entrega de valor a un ritmo sostenible.

Este ritmo sostenible en la dinámica de trabajo del equipo, requiere un ambiente laboral de mucha motivación, colaboración y empoderamiento del equipo, para construir la capacidad de auto organización de los equipos, que es un ingrediente indispensable para el éxito de la institucionalización de la transparencia y la mejora continua.

De manera, que definitivamente podemos concluir destacando que el aseguramiento de la calidad en los proyectos de software, desde una perspectiva donde únicamente se utilicen modernas herramientas de prueba, y la elaboración de planes que generen artefactos de documentación propios de un modelo de calidad, no son suficientes para obtener resultados exitosos.

Es necesario dar un paso más allá, mediante la adopción de un proceso adaptativo que impulse la agilidad dentro del código genético de la organización. En adición, es importante que el liderazgo responsable de promover la agilidad, reconozca que se trata de una ciencia empírica, que requiere una dinámica constante de exploración, tiempo para descubrir las prácticas que funcionan en la organización y recibir el apoyo incondicional desde las instancias más altas. Si tienen éxito en el trayecto, empezarán a revelarse beneficios significativos en torno a la calidad de los procesos, los proyectos y las personas.

Evaluación de la Calidad del Producto

Según Juran (1992), la calidad, para poder ser entendida de una mejor manera y posteriormente ser medida con eficacia, debe ser expresada por medio de otros términos que tengan más sentido para el usuario. En el caso del software. Estos factores son el medio por el cual se traduce el término “calidad” al lenguaje de las personas que manejan la tecnología.

Los factores de calidad que afectan a la calidad del software se dividen en dos grandes grupos:

Los que miden directamente (defectos descubiertos en las pruebas).

Los que se miden indirectamente (facilidad de uso o de mantenimiento).

En cada caso debe presentarse una medición. Se debe comparar el software con algún conjunto de datos y obtener así algún indicio sobre la calidad. McCall, Richards & Walters (1977), propusieron una clasificación de los factores que afectan directamente a la calidad del software. Estos factores se muestran en la figura 2.30 En ella se concentran tres aspectos importantes de un software:

1. Características operativas.
2. Capacidad para experimentar cambios.
3. Capacidad para adaptarse a nuevos entornos.

A continuación se describen los factores que propone McCall, Richards & Walters.

- Corrección.

El grado en que el programa cumple con su especificación y satisfacer los objetivos que propuso el cliente.

- Confiabilidad.

El grado en que se esperaría que un programa desempeña su función con la precisión requerida.

- Eficiencia.

La cantidad de código y de recursos de cómputo necesarios para que un programa realice su función.

- Integridad.

El grado de control sobre el acceso al software o los datos por parte de las personas no autorizadas.

- Facilidad de uso.

El esfuerzo necesario para aprender, operar y preparar los datos de entrada de un programa interpretan la salida.

- Facilidad de mantenimiento.

El esfuerzo necesario para localizar y corregir un error en un programa.

- Flexibilidad.

El esfuerzo que demanda probar un programa con el fin de asegurar que realiza su función.

- Portabilidad.

El esfuerzo necesario para transferir el programa de un entorno de hardware o software a otro.

- Facilidad de reutilización.

El grado en que un programa o partes de él pueden reutilizarse en otras aplicaciones (en relación con el empaquetamiento y el alcance de las funciones que realiza el programa).

- Interoperabilidad.

El esfuerzo necesario para acoplar un sistema con otro.

Es difícil y en algunos casos imposibles, desarrollar medidas directas 1 de estos factores de la calidad. En realidad, muchas de las métricas que definen McCall et al. Sólo se miden de forma subjetiva. Ya que es común que las métricas adquieran la forma de una lista de comprobación que se emplea para “asignar una graduación” a atributos específicos del software. Vega et al. (2008), proponen un modelo con métricas distintas al propuesto por McCall y que ha sido utilizado y comprobado en distintos proyectos de desarrollo de software. Los factores que conforman al modelo y su descripción, se presentan a continuación.

- Corrección.

El grado en que un producto de software satisface sus especificaciones y consigue los objetivos de la misión encomendada por el usuario.

- Confiabilidad.

El grado en que se puede esperar que un producto de software lleve a cabo sus funciones esperadas con la precisión requerida.

- Eficiencia.

La cantidad de recursos computacionales y de código requeridos por un producto de software para llevar a cabo las funciones encomendadas.

- Integridad.

El grado en que puede controlarse (facilitar y restringir) el uso y acceso al software y a los datos, tanto al personal autorizado como al no autorizado.

- Facilidad de uso.

El esfuerzo requerido para aprender, trabajar, preparar la entrada e interpretar la salida de un producto de software.

- Facilidad de mantenimiento.

El esfuerzo necesario para localizar y corregir los errores en un producto de software.

- Flexibilidad.

El esfuerzo requerido para modificar un producto de software una vez que se encuentra ya liberado o en producción, esto es, una vez que el usuario esté haciendo uso de él.

- Facilidad de prueba.

El esfuerzo requerido para probar un producto de software, de tal forma que se asegure que realiza las funciones especificadas por el usuario.

- Portabilidad.

El esfuerzo requerido para transferir un producto de software de una plataforma (entorno de hardware y software) a otra.

- Reusabilidad.

El grado en que un producto de software (o alguna de sus partes) pueda volver a ser utilizado en otras aplicaciones, aun cuando la funcionalidad de la misma cambie.

- Facilidad de interoperación.

El esfuerzo requerido para lograr que un producto de software trabaje con otro, compartiendo recursos.

Atributos de Calidad de Producto Informático

Un atributo de calidad es una propiedad medible de un sistema, que indica qué tan bien el sistema satisface las necesidades de las partes interesadas.

A los atributos de calidad también se les conoce como:

Requerimientos no funcionales.

Características de arquitectura.

Propiedades de calidad.

Teniendo en cuenta la anterior definición, los atributos de calidad de este proyecto son los siguientes:

1. Este proyecto tendrá visibilidad del estado del sistema porque se tendrá todas las posibilidades que dispone la aplicación a la vista en todo momento gracias a la interfaz gráfica y a sus menús
2. Este proyecto tendrá control y libertad para el usuario porque tendrá control total sobre la aplicación y todas las operaciones que puede hacer, no tendrá distinciones ni permisos especiales
3. Este proyecto será robusta y consistente por ende tendrá prevenciones de errores
4. Este proyecto tendrá diálogos estéticos y diseño minimalista debido a que la interfaz gráfica se ha diseñado para que contenga la información necesaria para su uso y nada más
5. Este proyecto será accesible debido a que tendrá capacidades de adaptación que podrá disponer la aplicación

6. Este proyecto utilizara el lenguaje de los usuarios debido a que como no se conoce exactamente al posible usuario todo se controla con menús intuitivos además de utilizar términos científicos-técnicos apropiados
7. Este proyecto se ira trabajando en lapsos de 2 horas diarias por un aproximado de 7 meses en lo cual es el estimado que se realizara el mismo.
8. Este proyecto tendrá una facilidad de uso moderada debido a que será necesaria una pequeña capacitación para el uso del mismo
9. Este proyecto será del 1 a 10 un 8.5 de confiable lo cual se ira resolviendo sobre la marcha cualquier imprevisto que salgan después de la implementación