



UNIVERSIDAD AUTÓNOMA LATINOAMERICANA

Facultad de Ingenierías

Programa de Ingeniería Informática

**ANÁLISIS DE CASOS SOBRE CONTEXTOS DE ASEGURAMIENTO DE LA
CALIDAD (QA)**

**ACTIVIDAD EVALUATIVA
IF0116 – CALIDAD DE SOFTWARE**

Valor: 15%

26 de febrero del 2026

Medellín, Antioquia.

**PUNTO 1 (Valor 1.0 unidad)**

¿Cuál es el costo de no certificar en procesos de calidad un producto de software? Justifique su respuesta:

Teniendo en cuenta que la no certificación en procesos de calidad de un producto de software implica no someterse voluntariamente a la auditoría de los organismos acreditados que efectúan la norma, justifico que el costo se refleja en riesgos operativos, financieros y competitivos, siendo así una incorrecta implementación del software en el mercado y una preocupante decisión que perjudica la certeza, causando desventaja competitiva. A falta de su auditoría de procesos, no habría una documentación justificada en marcos formales ni mucho menos se daría algún voto de confiabilidad.

El rendimiento, la seguridad y la usabilidad no están garantizados y no sería posible abarcar confianza frente al cliente, consolidando así una mayor probabilidad de incremento de costos en mantenimiento, combinada con una pérdida de oportunidades comerciales, ya que se exigen certificaciones como ISO 9001.

**PUNTO 2 (Valor 0.5 unidad)**

¿Por qué es necesario hacer pruebas constructivas en el proceso de certificación de productos de software? Justifique su respuesta:

Es necesario realizar pruebas constructivas en el proceso de certificación de productos de software porque permiten demostrar de manera objetiva que el sistema ha sido desarrollado conforme a los requisitos funcionales, estándares técnicos y normativas aplicables. Estas pruebas generan evidencia verificable y trazable, requisito indispensable en un proceso formal de certificación, ya que los organismos evaluadores exigen pruebas documentadas que respalden el cumplimiento del producto.

Desde una perspectiva de gobernanza y aseguramiento de la calidad, las pruebas constructivas reducen riesgos, fortalecen la confiabilidad del software y garantizan que las reglas de negocio y controles de seguridad se ejecuten según lo diseñado. Por ejemplo, validar que un usuario sea bloqueado después de tres intentos fallidos de autenticación constituye una prueba constructiva que evidencia el correcto funcionamiento de un control de seguridad definido en los requisitos del sistema.



PUNTO 3 (Valor 0.5 unidad)

¿Por qué es necesario hacer pruebas analíticas en el proceso de certificación de productos de software? Justifique su respuesta.

Porque permiten verificar que el software está correctamente diseñado, que cumple los estándares y mantiene trazabilidad entre requisitos, arquitectura y código, cooperando desde la raíz para lograr la certificación. Es necesario realizar pruebas analíticas en el proceso de certificación de productos de software porque permiten evaluar el cumplimiento normativo del sistema antes y más allá de su ejecución.

Por ejemplo, el analizar el código fuente para comprobar que los controles de autenticación implementan el algoritmo criptográfico por poner un ejemplo, es una prueba analítica confirmando la correcta implementación técnica.



3. Caso Práctico

Partiendo del siguiente enunciado responder los puntos 4 y 5:

Se creó un software para los puntos de venta de un supermercado, el cual permitirá la automatización del pago en las cajas registradoras, y dar soporte a servicios más rápidos, baratos y mejores. Concretamente, el proceso incluye: Pago de los clientes, análisis de las ventas y control automático del inventario.

Una vez el software se comenzó a utilizar en el supermercado se encontró la siguiente situación:

“No calcula en forma correcta el total de la venta actual; no incluye el impuesto”.

PUNTO 4 (Valor 1.5 unidad)

¿Esta incidencia debió ser detectada en pruebas de qué tipo y por qué?:

Requiero que esta incidencia debió ser detectada en algún tipo de validación que corresponde a pruebas de caja negra orientadas al comportamiento esperado del sistema como pruebas constructivas, específicamente en pruebas funcionales del módulo de facturación por que el cálculo del total de la venta, incluyendo cargos adicionales, es un requisito funcional del sistema. Las pruebas constructivas tienen como objetivo validar que las funcionalidades implementadas cumplen exactamente con este requisito funcional el cual debería calcular que dado un producto con precio X y un impuesto Y%, el sistema reconozca correctamente el total incluyendo el impuesto.



$$Total = X + (X \times Y\%)$$

PUNTO 5 (Valor 1.5 unidad)

¿Esta incidencia es un error, un defecto o una falla y por qué?:

Esta incidencia corresponde a una falla → manifestación visible del defecto, porque el problema se evidenció durante el uso del sistema, técnicamente se clasifica como falla y a su vez también fue un “descuido”, por eso observemos:

Usuario final / Cajero → Detecta la falla (porque la ve en ejecución).

Desarrollador → Puede haber cometido un error de código (olvido o mala implementación).

Equipo de QA / Tester → Debió detectar el defecto antes de producción.

Analista de Requerimientos → Si el impuesto no estaba especificado, la omisión puede originarse aquí.

En este caso, el sistema ya estaba en uso y el problema se manifestó en operación real (“no calcula en forma correcta el total”). Por lo tanto, probablemente hubo un error de programación.



Conclusión

Concluyo que, a partir del análisis y los casos prácticos desarrollados, es posible evidenciar cómo las fallas en los procesos de aseguramiento de la calidad afectan directamente la confianza del cliente y generan costos adicionales para la organización. En consecuencia, la calidad del software debe gestionarse de manera estandarizada y estructurada, bajo enfoques preventivos y alineados con estándares reconocidos, garantizando un respaldo técnico y documental que fortalezca la confiabilidad y competitividad del producto en el mercado.

Gracias.

Referencias

International Organization for Standardization. (2015). ISO 9001:2015 Quality management systems — Requirements. ISO.