

Universidad Mariano Gálvez de Guatemala

Ingeniería en Sistemas

**Curso: Aseguramiento De La Calidad De
Software**

Ing. Carmelo Estuardo Mayén Monterroso



Contenido:

**Análisis crítico del aseguramiento de la
calidad de software**

Estudiante:

Jeremy Saul Rodriguez Garcia

Carnet:

1790-20-20725

24 de Julio de 2025

Estándares internacionales y Calidad del Software

1. ISO/IEC 25010: Modelo de calidad del producto del Software

El modelo de calidad representa la piedra angular en torno a la cual se establece el sistema para la evaluación de la calidad del producto. En este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado.

La calidad del producto software se puede interpretar como el grado en que dicho producto satisface los requisitos de sus usuarios aportando de esta manera un valor. Son precisamente estos requisitos (funcionalidad, rendimiento, seguridad, mantenibilidad, etc.) los que se encuentran representados en el modelo de calidad, el cual categoriza la calidad del producto en características y subcaracterísticas.

El modelo de calidad del producto definido por la ISO/IEC 25010 se encuentra compuesto por las nueve características de calidad que se muestran a continuación

Adecuacion Funcinonal:	Eficencia de Desempeño	Compatibilidad
<ul style="list-style-type: none">• Completitud Funcional• Correcion Funcional• Pertinencia Funcional	<ul style="list-style-type: none">• Comportamiento Temporal• Utilizacion de Recursos• Capacidad	<ul style="list-style-type: none">• Coexistencia• Interoperabilidad
Capacidad de Interaccion	Fiabilidad	Seguridad
<ul style="list-style-type: none">• Reconocibilidad de Adecuacion• Aprendizabilidad• Operabilidad	<ul style="list-style-type: none">• Ausencia de fallos• Disponibilidad• Tolerancia a fallos	<ul style="list-style-type: none">• Confidencialidad• Integridad• No-Repudio
Mantenibilidad	Flexibilidad	Proteccion
<ul style="list-style-type: none">• Modularidad• Reusabilidad• Analizabilidad	<ul style="list-style-type: none">• Adaptabilidad• Escalabilidad• Instalabilidad	<ul style="list-style-type: none">• Restriccion operative• Identificacion de riesgos• Proteccion ante fallos

2. ISO/IEC/IEEE 29119: Estándares de pruebas de software

ISO/IEC/IEEE 29119 Software Testing es un conjunto de estándares acordados internacionalmente para pruebas de software que se pueden utilizar dentro de cualquier ciclo de vida u organización de desarrollo de software. Al implementar estos estándares, adoptará los únicos estándares acordados y reconocidos internacionalmente para pruebas de software, que brindarán a su organización un enfoque de pruebas de alta calidad que podrá comunicarse en todo el mundo.

Actualmente hay cinco estándares en la serie:

- ISO/IEC 29119-1: Conceptos y definiciones
- ISO/IEC 29119-2: Procesos de prueba
- ISO/IEC 29119-3: Documentación de prueba
- ISO/IEC 29119-4: Técnicas de prueba
- ISO/IEC 29119-5: Pruebas basadas en palabras clave

- ¿Qué son los estándares?

Conjunto de requisitos obligatorios establecidos por consenso y mantenidos por un organismo reconocido para prescribir un enfoque uniforme y disciplinado o especificar un producto, es decir, convenciones y prácticas obligatorias

Las normas proporcionan un conjunto de conocimientos que sienta las bases de una disciplina profesional.

Base para:

- Comunicación: terminología común
- Cualificaciones profesionales
- Esquemas de certificación/cumplimiento
- Referencia de buenas prácticas del sector
- Contratos

¿Para qué un estándar para pruebas de software?

Cuerpo de conocimiento como base para esta disciplina profesional

- Alto grado de acuerdo internacional
- Adaptable (Tayloring)

Cliente:

- Confianza en el proveedor
- Benchmark para “buena” práctica industrial
- Vinculación contractual

Proveedor:

- Comunicación – terminología común
- Certificación/Acreditación, marketing
- Cualificación profesional, formación
- Mejora continua, Interoperabilidad, consistencia...

- ¿Qué estándares para Pruebas de Software podríamos usar?

Estándares BSI

- BS 7925-1, SW Testing: Part 1-Vocabulary
- BS 7925-2, SW Testing: Part 2-Software Component Testing

Estándares IEEE testing:

- IEEE Std 829, Software Test Documentation
- IEEE Std 1008, Software Unit Testing

¿Qué problemas plantean algunos de estos estándares?

IEEE & BSI Standards

- IEEE 829: documentación (proceso implícito)
- IEEE 1008: pruebas unitarias
- BS 7925-1 & 2: pruebas unitarias

Aspectos clave sin cubrir para un estándar de propósito

general:

- Pruebas no unitarias (integración, sistema, aceptación)
- Modelo de procesos explícito
- Aspectos organizativos y gestión del proyecto. Riesgos
- Visión más completa de técnicas de prueba

3. IEEE 730: Estándar de planes de aseguramiento de la calidad

- Definición del Estándar

El estándar IEEE 730 define el proceso de aseguramiento de calidad del software proponiendo un marco para la creación y mantenimiento de los planes de aseguramiento de Calidad (SQAP) que deben seguir los proyectos de software.

- Importancia en el Software

Implementar el Estándar IEE 730 es crucial para garantizar la calidad del software, minimizando riesgos y asegurando que los productos cumplan con las expectativas y requisitos del cliente. Esto resulta en una mayor satisfacción y confianza del usuario.

- Objetivo del Estándar

El Estándar IEE 730 busca definir y facilitar un enfoque sistemático para el aseguramiento de calidad del software, promoviendo practicas que incrementen la efectividad y eficiencia en el desarrollo de software

- Componentes del SQAP

Los componentes clave del SQAP incluyen la definición de requisitos, planes de prueba, metodologías de revisión y gestión de riesgos. Cada uno de estos elementos contribuye a prevenir defectos y asegurar la calidad del producto final

- **Proceso de Elaboración**

El proceso de elaboración del SQAP comienza con la identificación de las necesidades del cliente y la definición de los objetos de calidad. Luego, se desarrollan estrategias, se asignan responsabilidades y se establecen métricas para medir el éxito del plan

- **Instrucciones de Aplicación**

Las instrucciones de aplicación del SQAP deben ser claras y comprensibles orientando a los equipos sobre como implementar los procesos definidos incluye pautas para la documentación, la capacitación del personal y la evaluación continua

Implementación del proceso de aseguramiento de calidad.

En esta categoría del plan de trabajo se establecen los procesos de aseguramiento de la calidad que estarán en operación durante la ejecución del proyecto de software. Además, estos procesos de aseguramiento de la calidad se coordinan con los procesos de revisión, verificación y validación, auditorías, y otros procesos del ISO 12207 (ISO, 2008) que sean relevantes para los objetivos del proyecto. Por otra parte, se documentan los riesgos del proyecto, herramientas de soporte de aseguramiento a la calidad, estándares y prácticas aplicables, esfuerzo, recursos y cronograma relacionados con la gestión de la calidad.

Actividades de aseguramiento del producto.

En esta categoría se incluyen las actividades que evalúan la adherencia a los requisitos. Entre ellas se evalúa la conformidad de los planes del proyecto con los contratos, estándares y regulaciones. Se evalúan la conformidad del producto con los requisitos establecidos, con su aceptabilidad, con la conformidad de las actividades de soporte. Además, se consideran las actividades de medición del producto.

Actividades de aseguramiento del proceso.

Estas actividades verifican que el proceso de software empleado en el proyecto se adhiera a los estándares acordado. Incluye la evaluación de conformidad de los procesos y planes, la evaluación de los ambientes de desarrollo para conformidad, así como los procesos de los subcontratistas. Además, se incluye la medición del proceso y la valoración de habilidades y conocimientos del personal.

- **Para cada estándar describe su propósito, como se aplica en un proyecto real y que beneficios aporta al proceso de aseguramiento de la calidad**

ISO/IEC/25010 - ¿Cuál es su propósito?

El propósito de este estándar es definir qué aspectos hacen que un software tenga calidad. Nos da una lista de características como funcionalidad, seguridad, rendimiento, usabilidad, entre otras, que sirven como guía para evaluar si un software cumple o no con lo que los usuarios esperan.

¿Cómo se aplica en un proyecto real?

En un proyecto real, por ejemplo, una aplicación para comprar boletos de cine, este estándar te ayuda a asegurarte de que la app cumpla con todo lo necesario para ser útil y confiable.

¿Qué beneficios aporta al proceso de aseguramiento de la calidad?

- Te permite medir la calidad de forma clara y ordenada.
- Mejora la comunicación entre los desarrolladores, testers y el cliente.
- Reduce el riesgo de errores o malas decisiones en el desarrollo.

ISO/IEC/IEEE 29119 - ¿Cuál es su Proposito?

Este conjunto de estándares tiene como objetivo definir cómo deben hacerse las pruebas de software de forma profesional. Incluye procesos, técnicas, documentación y guías para hacer pruebas en cualquier tipo de proyecto, lo que ayuda a asegurar que el software funcione correctamente y cumpla lo que promete.

¿Cómo se aplica en un proyecto real?

Por ejemplo, si estás trabajando en un sistema para registrar notas en una universidad, este estándar te dice cómo organizar el equipo de pruebas, cómo documentar los resultados y qué técnicas usar.

- Se aplican distintos tipos de pruebas:
- Pruebas unitarias (pequeñas partes del software)
- Pruebas de integración (cómo se conectan las partes)

¿Qué beneficios aporta al proceso de aseguramiento de la calidad?

- Mejora el control y la organización del proceso de pruebas.

- Aumenta la confiabilidad del software al detectar errores antes de que lleguen al usuario.
- Permite que los equipos hablen el mismo idioma (terminología común).

IEEE 730 - ¿Cuál es su Proposito?

El IEEE 730 tiene como propósito definir cómo se debe planificar el aseguramiento de calidad en un proyecto de software. Es decir, cómo prevenir problemas, organizar las actividades de calidad y asegurarse de que todo el equipo esté alineado con los objetivos del proyecto.

¿Cómo se aplica en un proyecto real?

Si estamos desarrollando un sistema para gestionar citas médicas, este estándar nos podría ayudar a:

- Crear un plan claro (SQAP) con los pasos y responsables de cada tarea relacionada con la calidad.
- Establecer qué se va a revisar, cómo se van a hacer las pruebas y cómo se gestionarán los riesgos.

¿Qué beneficios aporta al proceso de aseguramiento de la calidad?

- Nos da una estructura ordenada para trabajar con calidad desde el inicio.

2. Comparativa de modelos de calidad

Criterio	ISO/IEC 25010	Modelo de McCall	Modelo de Boehm (Behm)
Época de aparición	2011 (sustituye al ISO 9126)	Finales de los años 70 (1977)	Inicios de los años 80 (1981)
Enfoque	Enfocado en las características internas y externas del producto de software y su calidad en uso.	Se enfoca en factores técnicos y operacionales del software.	Enfocado en satisfacción del usuario y cómo se percibe la calidad del software.

Criterio	ISO/IEC 25010	Modelo de McCall	Modelo de Boehm (Behm)
Atributos evaluados	8 características principales: funcionalidad, confiabilidad, usabilidad, rendimiento, seguridad, compatibilidad, mantenibilidad, portabilidad. Cada una con subcaracterísticas.	3 grandes grupos: Operación del producto, Revisión del producto, Transición del producto. Evalúa factores como corrección, eficiencia, mantenibilidad, etc.	3 niveles: características de alto nivel (como utilidad), características intermedias (fiabilidad, eficiencia, etc.), y métricas primitivas.
Ventajas	- Actual y completo. - Internacionalmente aceptado. - Incluye aspectos modernos como seguridad y experiencia del usuario.	- Fue uno de los primeros modelos que permitió medir la calidad. - Buena base para entender la calidad desde el punto de vista técnico.	- Más cercano a la perspectiva del usuario. - Enlace claro entre calidad técnica y satisfacción del cliente.
Desventajas	- Puede ser complejo de aplicar sin experiencia. - Requiere adaptación según el proyecto.	- Enfoque técnico algo limitado para entornos modernos. - No contempla aspectos actuales como seguridad o experiencia de usuario.	- Estructura compleja. - No contempla directamente nuevas tendencias como agilidad o DevOps.
Uso actual	Muy usado en empresas modernas, especialmente en proyectos que buscan certificación de calidad o alineación con normas internacionales.	Usado principalmente con fines educativos o históricos. Algunas empresas lo toman como base inicial.	Poco usado actualmente, pero influyó en modelos más modernos. Se usa para análisis comparativos y académicos.

3. Roles en el aseguramiento de calidad

1. Project Manager

¿Qué hacen?

El Project manager está a cargo del desarrollo, la organización y la entrega de un proyecto. Ellos dividen el proyecto en partes modulares y diseñan el flujo de trabajo para distribuir las tareas entre el equipo, y, además, son responsables de los contratos, los presupuestos y los tiempos. Asimismo, la mitigación de riesgos y la gestión de circunstancias o demoras inesperadas son responsabilidades de los Project managers. Sin ellos, no habría un plan o estructura para entregar el trabajo.

¿Qué habilidades aportan al equipo?

Los Project managers necesitan fuertes habilidades de liderazgo y organización, ya que son los que coordinan a los distintos miembros del equipo. También deben ser comunicadores eficaces, porque conectan a todas las partes del proyecto y se relacionan con el cliente.

¿Con quienes interactúan más en el equipo?

Como los Project managers son los que supervisan todas las partes del proyecto de principio a fin, interactúan con todos los miembros del equipo, así como también con el cliente y los accionistas. Con eso dicho, el analista comercial interactúa más con el cliente, mientras que el Project manager se enfoca más en el equipo.

2. Scrum Master

¿Qué hacen?

El rol del scrum master es asegurarse de que el equipo siga metodologías y estructuras ágiles (o “Agiles”, como se las conoce en inglés). Estas metodologías son un concepto que puede describirse como una mentalidad de gestión de proyectos que se enfoca en la colaboración y la creación de valor. Según el Manifiesto Ágil, sus valores principales incluyen:

- Individuos e interacciones por sobre procesos y herramientas
- Software funcional por sobre documentación comprensible
- Colaboración con el cliente por sobre negociación de contratos

- Responder a los cambios por sobre seguir un plan

El scrum master es el responsable de cómo se siguen las estructuras en el equipo. Su rol incluye eliminar obstáculos, crear un ambiente productivo y ayudar al equipo a trabajar bien en conjunto. Aunque técnicamente es un líder, no tienen autoridad por sobre otros miembros del equipo.

¿Qué habilidades aportan al equipo?

Los scrum masters deben contar con habilidades fuertes de liderazgo, coaching y organización. También deben tener experiencia en Project management y un entendimiento profundo del desarrollo de software y de metodologías ágiles.

¿Con quienes interactúan más en el equipo?

Como son los responsables de garantizar el uso de las metodologías ágiles en el equipo, los scrum masters interactúan con todos los miembros.

3. Analista comercial

¿Qué hacen?

El analista comercial es el responsable de cumplir con las necesidades comerciales del cliente. Esto significa que tienen en cuenta los objetivos comerciales del cliente y lo ayudan a definir metas claras. Luego, ellos toman estos objetivos y los convierten en soluciones de software viables. Están involucrados en el proyecto desde el inicio, y conectan las necesidades del cliente con las realidades del proyecto. Además, llevan a cabo análisis del mercado, evalúan a la competencia y definen a la audiencia meta. También crean soluciones de documentación y prueban soluciones.

¿Qué habilidades aportan al equipo?

Los analistas comerciales deben entender de números, ya que interpretan información para defender sus decisiones. También necesitan ser pensadores críticos y ser creativos ya que “traducen” lo que el cliente necesita en una solución entregable.

¿Con quienes interactúan más en el equipo?

Los analistas comerciales trabajan de cerca con el dueño del producto y con el project manager.

- **Riesgos de no implementar un proceso formal de aseguramiento de la calidad en el desarrollo del software**

Dentro de los riesgos más comunes que se presentan en un software tenemos:

- El software no hace lo que se espera.
- Tiempos de respuesta inadecuados.
- Cálculos incorrectos.
- Mala experiencia de usuario.
- Falta de accesibilidad.
- Entre otros.

Entre los impactos que podrían generar tenemos:

- Pérdida de confianza en el producto.
- Pérdida de clientes.
- Pérdida de dinero.
- Pérdida de la vida
- Entre otros.

En cualquier proyecto existe la probabilidad que ocurra cualquiera de estos escenarios adversos, pero si excluimos las pruebas de calidad nos arriesgamos que todo esto pueda ocurrir en un entorno operativo. Las pruebas reducen la probabilidad que ocurran ya que se utilizan para mitigar los riesgos identificados en el proyecto antes de ser entregado al usuario final.

El proceso de pruebas tiene un gran impacto sobre los riesgos en un proyecto de desarrollo de software, pero cabe aclarar que incluir el proceso de calidad no asegura que el software no contenga errores, por las mismas razones que el no incluirlas no asegura que no hay errores, por que aun los mismos probadores son seres humanos falibles, pero lo que sí se puede asegurar es que bajo las condiciones probadas el software funcionará correctamente. Las pruebas ayudan a reducir el riesgo que el software falle en el entorno de operación. En Consulting Group la calidad es fundamental y aprovechamos los

beneficios del proceso de pruebas para poder entregar proyectos exitosos que ayuden a los clientes a sacar el mayor provecho a su inversión

- **Los Enemigos de la Calidad: ¿Cómo puede dañar a tu proyecto de desarrollo de software, la falta de buenas prácticas QA?**

En el dinámico mundo del desarrollo de software, la calidad es un factor crucial que puede determinar el éxito o el fracaso de un proyecto. Un proceso robusto de Aseguramiento de la Calidad (QA) no solo garantiza que el producto final cumpla con las expectativas del cliente, sino que también mejora la eficiencia del equipo de desarrollo y reduce los costos a largo plazo.

- **Las amenazas ocultas del Desarrollo de Software**

Los problemas que surgen de un proceso de QA deficiente no solo afectan la funcionalidad del producto final, sino que también pueden generar costos adicionales, retrasos en el proyecto y una experiencia de usuario insatisfactoria. En esta sección, exploraremos cómo los problemas representados por los personajes de nuestra campaña, Vagus, Bugashi y Engorro, pueden impactar negativamente en la calidad del software y cómo Tsoft puede ayudar a tu organización a enfrentarlos y superarlos.

Los problemas que surgen de un proceso de QA deficiente, no sólo afectan la funcionalidad del producto final, sino que también puede generar costos adicionales, retrasos en el proyecto y una experiencia de usuario insatisfactoria.

- **Ejemplos de cómo la falta de Sistemas de Gestión de Calidad puede afectar a las Empresas**

- 1. La Calidad en un Restaurante**

Un grupo de jóvenes reserva cuatro mesas grandes en el restaurante X. El camarero que les sirve es grosero y se produce una discusión. El mesero acusa a una clienta embarazada de llevar alcohol a las instalaciones cuando la clienta embarazada solo ordenó bebidas sin alcohol durante toda la noche. Molesto por las acusaciones y la grosería del camarero, la clienta embarazada le cuenta al resto del grupo. Aunque hay cámaras instaladas en el restaurante, el equipo de administración no puede ver las imágenes para determinar qué ocurrió y asume que el camarero tiene razón. En lugar de disculparse por las acusaciones del camarero, advierten al grupo y les dicen que pueden quedarse, pero que no deben ser

groseros y no pueden consumir alcohol no comprado en el restaurante, ignorando el hecho de que el camarero insultó al cliente.

El grupo se va poco después debido al incidente. Al día siguiente estalla una guerra de llamas en la página de redes sociales del restaurante y en 48 horas, la buena reputación del restaurante en la comunidad cae a una calificación de 2 a 5. En lugar de calmar la situación, el equipo de administración sigue insultando a la clienta embarazada. su página de redes sociales en respuesta a su ventilación pública en la página sobre el incidente. Muy pronto, otros clientes que también estaban en el restaurante cuando ocurrió el incidente se quejan de la falta de servicio y la mala calidad de la comida. Esto lleva a más acusaciones y el restaurante pierde una gran parte de su base de clientes.

El ejemplo anterior muestra lo fácil que es para una empresa perder clientes, ingresos, prestigio público y credibilidad a través de un solo incidente. Si se hubiera implementado un control de calidad adecuado con respecto a cómo dirigirse a los clientes, qué nivel de servicio brindar, cómo resolver los problemas y qué calidad de los alimentos y el servicio brindar, el incidente tal vez nunca hubiera sucedido. Si el equipo de gestión se hubiera acercado a la situación de acuerdo con un conjunto de reglas y procedimientos para hacer frente a tales situaciones, el restaurante, a pesar de que el camarero estaba equivocado, habría sido capaz de apaciguar al cliente y retratar una imagen profesional.

2. Calidad en el Comercio Internacional

Supongamos que una empresa en Sudáfrica ordena una gran cantidad de juguetes inflables a un proveedor en otro país. Los productos llegan tres semanas después y la empresa sudafricana los distribuye sin controles de calidad. Los productos terminan en los estantes de los minoristas, pero cada séptimo artículo vendido se devuelve debido a una falla. La compañía sudafricana reembolsa a los minoristas que reembolsan a sus clientes.

Los minoristas están molestos, los clientes no quieren volver a comprarles, y la compañía sudafricana pierde una gran cantidad de dinero y su buena reputación como proveedor. Tenían razones para esperar una calidad constante, pero aunque el proveedor extranjero tiene un sistema de gestión de calidad, la falta de documentación conduce a inconsistencias

en el informe de los productos devueltos y debido a que la compañía tampoco realiza auditorías de calidad a su proveedor, termina teniendo un impacto La peor parte de los vendedores enojados.

Todo se reduce a que las empresas no pueden permitirse el lujo de operar sin contar con sistemas de gestión de calidad. También significa que los sistemas de gestión de calidad deben implementarse en toda la organización. Deben realizarse mejoras constantes y debe existir documentación para garantizar la coherencia. También se hace evidente que la idea de una empresa de un sistema de gestión de calidad puede diferir de la de otra. Con la estandarización de los sistemas de gestión de calidad, es posible tener la seguridad de que un proveedor extranjero aplicará controles de calidad específicos en la fabricación y entrega de sus productos. Aquí es donde la Organización Internacional de Normalización se vuelve importante. Con el establecimiento de normas ISO, es posible crear puntos de referencia con los que se pueden medir los sistemas de gestión de calidad de las empresas.

Referencias

ISO/IEC 25010: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>

ISO/IEC/IEEE 29119: <https://softwaretestingstandard.org> &
<https://in2test.lsi.uniovi.es/gt26/presentations/ISO29119-Presentacion-GT26-20151215.SanSebastian.pdf>

IEEE 730: <https://www.mat.uson.mx/~mireles/gestionCalidad/GCgestion.html> &
<https://prezi.com/p/b1glwreovdbc/investigacion-sobre-el-estandar-ieee-730>

Roles en el aseguramiento de la calidad

<https://projectcor.com/es/blog/roles-fundamentales-en-un-equipo-de-desarrollo-de-software>

Riesgos de no implementar un proceso formal de aseguramiento de calidad en el desarrollo del software

<https://www.cgclatam.com/Blog/Las-consecuencias-de-no-incluir-pruebas-de-calidad-en-un-proyecto-de-software> & <https://www.tsoftglobal.com/los-enemigos-de-la-calidad-como-puede-danar-a-tu-proyecto-de-desarrollo-de-software-la-falta-de-buenas-practicas-qa>