



Calidad de software

Calidad de software

Contenidos

- ¿Qué es Calidad?
- ¿Qué no es Calidad?
- Terminología (ISO 8402):
- La calidad del software:
- Factores que determinan la calidad del software
- Medir la calidad del software
- Un proceso "confiable"
- El Comité de Calidad != Calidad del producto
- El uso de estándares != Calidad del producto
- La industria de software

Calidad de software

Contenidos

- ¿Cómo se puede llegar a la Calidad?
- Sistema de Calidad
- Enfoques de calidad
- Software Quality Assurance (SQA)
- Atributos de Calidad
- Funciones y Actividades de SQA
- Habilidades y Capacidades requeridas en SQA
- ISO 25000
- Referencias

¿Qué es Calidad?

La Calidad puede definirse como el conjunto de características de una entidad, que le confieren la aptitud para satisfacer las necesidades establecidas y las implícitas [ISO 8402].

- El concepto de entidad se extiende tanto a **procesos**, como a **productos** y **servicios**, una organización, o combinación de ellos.
- Las **necesidades establecidas** son las que están especificadas, ya sea por un reglamento (necesidades para un proceso, producto, etc.), por un cliente (características para un producto o servicio), etc.
- Las **necesidades implícitas** son las que no están especificadas, pero que conviene identificar y definir.

¿Qué no es Calidad?

- Unos lo confunden con un ***producto de unas cualidades inmejorables***. Sin embargo la calidad va más allá de las características de un producto o servicio.
- Otros la asocian con una ***acumulación de papeles*** que no sirven sino para ralentizar el trabajo, y el desarrollo de las actividades. Sin embargo, la calidad, o en este caso el ***SISTEMA DE CALIDAD*** en una organización, es algo más que una serie de documentos y papeles para rellenar.
- Y muchos otros identifican calidad con ***CONTROL DE CALIDAD***, siendo este último sólo una parte que constituyen un Sistema de Calidad.

Terminología (ISO 8402):

Calidad:

“Conjunto de propiedades y características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades explícitas o implícitas”.

Control de calidad:

“Conjunto de técnicas y actividades de carácter operativo, utilizadas para verificar los requerimientos relativos a la calidad del producto o servicio”.

Terminología (ISO 8402):

Garantía de calidad:

“Conjunto de acciones planificadas y sistemáticas necesarias para proporcionar la confianza adecuada de que un producto o servicio satisfará los requerimientos dados sobre calidad”.

Terminología (ISO 8402):

Gestión de la calidad:

“Aspecto de la función de gestión que determina y aplica la política de la calidad, los objetivos y las responsabilidades y que lo realiza con medios tales como la planificación de la calidad, el control de la calidad, la garantía de calidad y la mejora de la calidad”.

- La gestión de la calidad es responsabilidad de todos los niveles ejecutivos, pero debe estar guiada por la alta dirección. Su realización involucra a todos los miembros de la organización.
- En la gestión de la calidad, se tienen en cuenta también criterios de rentabilidad.

Terminología (ISO 8402):

Sistema de gestión de la calidad:

“Conjunto de la estructura de la organización, de responsabilidades, procedimientos, procesos y recursos que se establecen para llevar a término la gestión de calidad”.

- El QS debe tener el volumen y alcance suficiente para conseguir los objetivos de calidad.
- El QS de una organización está fundamentalmente previsto para satisfacer las necesidades internas de la organización. Es más amplio que los requerimientos de un cliente concreto que únicamente valor el QS que le interesa (directamente).
- Para finalidades contractuales o vinculantes en la valoración de la calidad, se puede exigir que se ponga de manifiesto la realización de ciertos elementos del QS.

La calidad del software:

“La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario”. (IEEE, Std. 610-1990).

*“Concordancia del software producido con los requerimientos explícitamente establecidos, con los estándares de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente, que desea el usuario”
(Pressman, 1998).*

Factores que determinan la calidad del software:

Se pueden clasificar en dos grandes grupos:

- Factores que pueden ser medidos directamente.
- Factores que solo pueden ser medidos indirectamente.

Se centran en tres aspectos importantes de un producto software:

- Características operativas.
- Capacidad de soportar los cambios.
- Adaptabilidad a nuevos entornos.

Factores que determinan la calidad del software:

Características operativas:

- ***Corrección:*** ¿Hace lo que quiero?
- ***Fiabilidad:*** ¿Lo hace de forma fiable todo el tiempo?
- ***Eficiencia:*** ¿Se ejecutará en mi hardware lo mejor que pueda?
- ***Seguridad (Integridad):*** ¿Es seguro?
- ***Facilidad de uso:*** ¿Está diseñado para ser usado?

Capacidad de soportar los cambios:

- ***Facilidad de mantenimiento:*** ¿Puedo corregirlo?
- ***Flexibilidad:*** ¿Puedo cambiarlo?
- ***Facilidad de prueba:*** ¿Puedo probarlo?

Factores que determinan la calidad del software:

Adaptabilidad a nuevos entornos:

- **Portabilidad:** ¿Podré usarlo en otra máquina?
- **Reusabilidad:** ¿Podré reutilizar alguna parte del software?
- **Interoperabilidad:** ¿Podré hacerlo interactuar con otro sistema?

Medir la calidad del software:

Dogmas, leyendas y supersticiones sobre la creación de software de calidad que:

- no son congruentes con los hechos
- más y más gentes cuestionan cada vez menos.

y se convierten en:

- verdades cotidianas
- estándares a exigir.

Medir la calidad del software:

Es posible medir subjetivamente (solicitando opiniones) la calidad del software

- **Confiable:** Pocos errores.
- **Flexible:** Adaptable a nuevas situaciones.
- **Robustez:** No falla.
- **Comprensión:** ¿Es entendible el código?
- **Fácil de usar:** Ergonómico.
- **Reusable:** Se pueden usar porciones en otro software.
- **Rápido:** (medición objetiva)
- **Mantenible:** Fácil de hacerle cambios.

Medir la calidad del software:

- Pocas mediciones son objetivas
- Se mide altura en vez de volumen
- Muchas mediciones son estáticas
- Medimos lo que podemos, no lo que deberíamos

“busco la llave donde hay luz”

- Para cada atributo hay una medición confiable (objetiva)

“Si no puede medir la liebre, mida al gato. Da lo mismo.”

Un proceso "confiable"

Un proceso confiable produce un producto confiable

- Una manera de hacer buenos productos es seguir buenas recetas (buenos procesos, buenos algoritmos)
 - *Ejemplo: sopa de arroz*
 - *Ejemplo: curtido de cuero*
- Esto es cierto cuando la humanidad ha tenido mucha experiencia O cuando hay ciencias (Física, Química...) antiguas que apoyan tales procesos.
- Entonces, el proceso se vuelve observable (podemos ver si lo estamos siguiendo bien)
- Y el producto se vuelve controlable:
 - *Dado un defecto en el producto, sabemos qué parte del proceso modificar*

Un proceso "confiable"

Pero esto no sucede con el software...

- Esto es cierto en la fabricación de bisagras o varillas de acero (leyes de la Física) .
- Pero no en la creación del software Más cercano a creación artística: sinfonías, novelas, pinturas, obras de teatro...
- Es creación, no fabricación!
- La calidad del software producido no está relacionada con el proceso seguido.
- Es posible diseñar un proceso confiable para producir buen software y si cumplimos con ese proceso, habremos hecho software de buena calidad.

Un buen proceso implica producir software de buena calidad
“Seguir el proceso para pintar de Diego Rivera asegura obras de alta calidad”

Un proceso "confiable"

Ejemplo de un proceso confiable, observable

- Reuniones cada lunes a las 9am; minutas
- Conformar un comité de calidad
- Que los integrantes del grupo de desarrollo conozcan las normas ISO 9000, 9000-3...
- Firme convicción de poder crear software de calidad
- Revisiones periódicas de código (walk-through)
- Llevar un Control de Cambios y un Control de Versiones

Todo esto no garantiza ***un producto de calidad***, el proceso es controlable; pero ¿el producto es controlable? ...

Un proceso "confiable"

- El proceso es observable (ver si se siguió)
- El proceso es controlable (fácil corregir: llegar más temprano a las juntas)
- El producto es observable (fácil ver si es de calidad)
 - *Está fallando la comunicación asíncrona entre procesos que van en distintos procesadores*
- Si se detecta alguno de los siguientes errores, ¿qué parte del proceso de creación de software modificar?
 - *La comunicación entre módulos en distintos procesadores a veces falla.*
 - *El software producido no es reusable.*
 - *Varias especificaciones no se cumplen.*
- No se sabe
- El producto no es controlable vía el proceso

El producto (software) no es controlable directamente desde el proceso!! - no es corregible mediante correcciones al proceso -

El Comité de Calidad != Calidad del producto:

- El comité de calidad elabora normas sobre cómo conducir el proceso y vigila el apego a ellas del grupo desarrollador
- Exige apego a las normas.
- Regaños y sanciones
- El problema original (software poco reusable) seguirá sin corregirse o se corregirá “por casualidad”.

"El Comité de Calidad de Software es algo bueno y su actuación corrige los problemas de calidad"

El uso de estándares != Calidad del producto

- “Hay que usar UML”
- “Hay que usar CMM del Software Engineering Institute”
- “Hay que usar ISO 9001”
- No se sabe cómo hacer software de calidad, copiar a alguien con éxito (en producción de software, no en venta de estándares) puede ser útil, ¡o puede no ser!

"Seguir estándares es bueno, si sigo un estándar internacional, no me puede ir mal."

La industria de software:

- No ha acabado de salir de la fase artesanal
- Padecemos de ***“prisa patológica”***, que es consecuencia directa de:
 - *Desorganización*
 - *Falta de planificación*
 - *Alta dependencia de los “héroes”*
- Dedicamos nuestros esfuerzos de hoy a arreglar lo que se hizo mal ayer.
- Carencia de un corpus de conocimiento aceptado mayoritariamente que sirva como fundamentos.
- La mayoría de las organizaciones son inmaduras en sus procesos.
- No somos concientes de la necesidad de formalizar los conocimientos, procesos y herramientas base.
- El software es mucho más que programas de computador.
- Construir software implica más tareas que el proceso de programar.
- Sin modelos de procesos y metodologías para gestionarlo el resultado es impredecible.

La industria de software:

Crisis, Software sin calidad.

¿Cómo se puede llegar a la Calidad?



Sistema de Calidad

- Un Sistema de Calidad identifica, documenta, coordina y mantiene las actividades necesarias para que los productos/servicios cumplan con los requisitos de calidad establecidos, sin tener en cuenta dónde estas actividades se producen.
- Un Sistema de Calidad pone requisitos a las actividades y procesos que se realizan en la empresa, y documenta cómo se realizan estas actividades.
- El objetivo de un Sistema de Calidad es satisfacer las necesidades internas de la gestión de la organización. Por tanto va más allá de satisfacer los requisitos que pone el cliente.
- Un Sistema de Calidad debe abarcar todas las actividades que se realizan en la empresa y que puedan afectar (directa o indirectamente) a la calidad del producto/servicio que suministra.
- Un Sistema de Calidad ayuda a evitar problemas en la ejecución de estas actividades, ya que la filtración de errores a través de las actividades de la empresa puede ocasionar importantes pérdidas.

Sistema de Calidad

- El costo de corregir un error entre proveedor y cliente antes de firmar el contrato, es mucho menor que si el error se detecta en la entrega al cliente del producto/servicio terminado.
- El espíritu de los Sistemas de Calidad es prevenir errores para evitar estas filtraciones y pérdidas económicas.

¿Qué alcance tiene un sistema de calidad?

- actividades de compra
- control del diseño
- control de la documentación
- realización de ofertas
- identificación de los productos
- control de los procesos
- inspección de los productos
- tratamiento de productos no conformes
- almacenamiento
- formación del personal
- etc.

Enfoques de calidad

El enfoque de calidad para la organización se puede afrontar en función de los siguientes factores:

- Tamaño de la empresa.
- Disponibilidad de recursos (entre ellos el tiempo).
- Motivo por el que necesita un Sistema de Calidad.

Los posibles enfoques son los siguientes:

- Enfoque global
- Enfoque práctico

Software Quality Assurance (SQA)



El Aseguramiento de la Calidad del Software es un conjunto de actividades sistemáticas que proveen capacidad al proceso de software para producir un producto adecuado para el uso.

Software Quality Assurance (SQA)

- SQA forma parte fundamental de los procesos de la Ingeniería de Software.
- El proceso de aseguramiento de calidad se debe de dar en cada uno de los procesos de la Ingeniería de Software y no sólo hasta el final.
- El SQA es el principal control de calidad y consiste en la realización de pruebas a todos niveles.
- El Software Quality Assurance tiene como objetivo el lograr la calidad del software.
- SQA se define por un conjunto de mejores prácticas que llevan la verificación y validación del software desarrollado.
- El proceso de SQA aunque se enfoca más en la medición y control del código fuente aunque tiene que ver con las demás áreas del proceso.
- El SQA logra resolver la mayoría de los errores que se podrían presentar en el software, asegurando que este sea confiable, usable y costo-efectivo.

Atributos de Calidad

Calidad del producto:

- usabilidad
- mantenibilidad
- correctitud
- rendimiento
- disponibilidad
- confiabilidad
- performance
- amigabilidad
- robustez
- portabilidad
- reusabilidad
- etc.

Calidad del proceso:

- El proceso debe estar definido, documentado y debe ser practicado y medido

Funciones y Actividades de SQA

Cada desarrollo de software es en alguna medida único. Aún cuando se usen los mismos métodos y técnicas, existen diferencias por factores que tienen impacto en la calidad, por ejemplo:

- Requerimientos de plazo
- Presupuesto disponible
- Complejidad técnica
- Tamaño esperado del producto
- Experiencia relativa del grupo humano
- Recursos disponibles
- Requerimientos contractuales

Funciones y Actividades de SQA

- Mejorar la calidad de los procesos de desarrollo y mantenimiento del software, monitoreando, durante el transcurso del ciclo de vida de los diferentes proyectos, el cumplimiento de los estándares y procesos establecidos, antes de su puesta en producción.
- Establecer planes, estándares y procesos que satisfagan las políticas de la organización y se ajusten a las necesidades de cada proyecto en particular.
- Revisar y auditar los productos y actividades desarrolladas para verificar que ellos satisfacen los procesos y estándares definidos;
- Proveer las herramientas necesarias que den soporte al proceso definido, facilitando su ejecución, visualización, administración y seguimiento;
- Proveer al equipo de proyecto y a otros interesados, los resultados sobre las revisiones, auditorías y actividades;
- Escalar problemas no resueltos dentro del equipo de un proyecto hacia un nivel apropiado de administración para su resolución.

Habilidades y Capacidades requeridas en SQA

- Conocimiento de todo el proceso de desarrollo de software.
- Experiencia en varios roles dentro de la organización.
- Ser proactivo.
- Capacidad de seguimiento de actividades.
- Capacidad de trato con la gente.

Calida de software:

“Grado en que el producto software satisface las necesidades expresadas o implícitas, cuando se usa bajo condiciones determinadas”. ISO 25000.

Aspectos de la calidad de software:

- ***Interna:*** medible a partir de las características intrínsecas del propio producto software (como el código fuente).
- ***Externa:*** medible a partir del comportamiento del producto software (como durante una prueba)
- ***En uso:*** medible durante la utilización efectiva por parte del usuario (en un entorno de prueba o producción).

ISO 25000

Software Product Quality Requirements and Evaluation (SQuaRE)

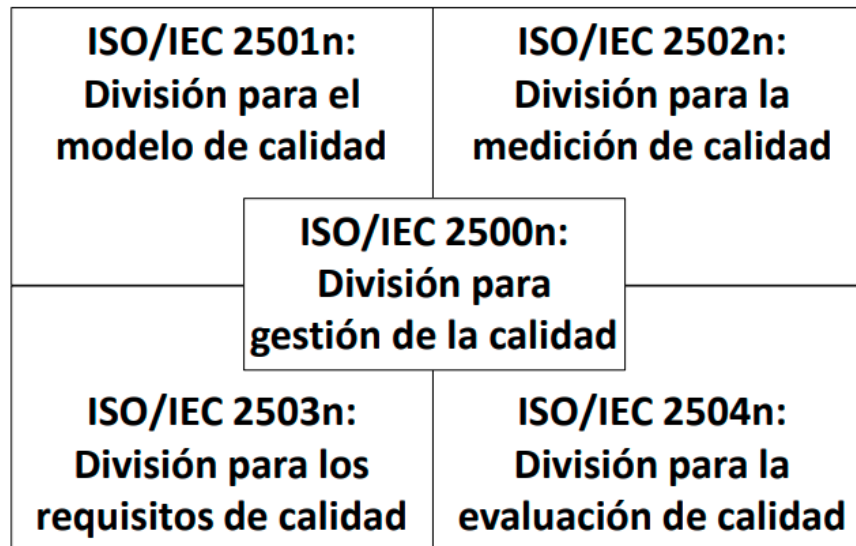
- Nace por las inconsistencias entre ISO 9126 e ISO 14598.
- El objetivo es aglutinar bajo una misma familia el modelo de calidad y el proceso de evaluación.
- La versión de la primera parte es de 2005.
- Estado actual: 90.93 International Standard confirmed (ver estados en: http://www.iso.org/iso/stages_table.htm)



ISO 25000

Software Product Quality Requirements and Evaluation (SQuaRE)

Formada por 5 partes (divisiones).



Referencias

- Pressman, R. (2010). Ingeniería del software (7ª ed.). México: McGrawHill.
- ISO/IEC 25000:2014 Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Guide to SQuaRE <https://www.iso.org/standard/64764.html>



Gracias!
