

Contents

Módulo 3, Clase 1: Introducción a los Requisitos de Calidad del Software	1
--	---

Módulo 3, Clase 1: Introducción a los Requisitos de Calidad del Software

Objetivos de la Clase:

- Definir el concepto de calidad del software y su importancia en el desarrollo de software exitoso.
- Comprender por qué los requisitos de calidad son cruciales para la planificación y ejecución de proyectos.
- Distinguir entre diferentes perspectivas sobre la calidad (usuario, desarrollador, empresa).
- Identificar las consecuencias de ignorar los requisitos de calidad.

Contenido Teórico Detallado:

1. ¿Qué es la Calidad del Software?

La calidad del software es un concepto multidimensional que abarca varios aspectos. No existe una única definición universalmente aceptada, ya que la calidad puede significar diferentes cosas para diferentes personas involucradas en el proyecto.

- **Desde la perspectiva del usuario:** La calidad se relaciona con la satisfacción de sus necesidades y expectativas. Un software de calidad es aquel que es fácil de usar, confiable y proporciona el valor esperado.
- **Desde la perspectiva del desarrollador:** La calidad se centra en la adherencia a estándares de codificación, la arquitectura del software y la facilidad de mantenimiento. Un software de calidad es aquel que es bien diseñado, documentado y fácil de modificar.
- **Desde la perspectiva de la empresa:** La calidad se relaciona con la rentabilidad, la reducción de costos de mantenimiento y el cumplimiento de los plazos de entrega. Un software de calidad es aquel que se desarrolla dentro del presupuesto, se entrega a tiempo y genera un retorno de la inversión aceptable.

En resumen, la calidad del software puede definirse como el grado en que un producto de software cumple con los requisitos especificados y satisface las necesidades implícitas del usuario. Implica atributos como funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad.

2. La Importancia de los Requisitos de Calidad:

Los requisitos de calidad son declaraciones que especifican las características y atributos deseados del software en términos de calidad. Estos requisitos guían el proceso de desarrollo y aseguran que el producto final cumpla con las expectativas de las partes interesadas.

- **Alineación con las necesidades del usuario:** Los requisitos de calidad ayudan a asegurar que el software se ajuste a las necesidades reales de los usuarios, lo que lleva a una mayor satisfacción y adopción.
- **Reducción de riesgos:** Al definir claramente los criterios de calidad, se pueden identificar y mitigar los riesgos asociados con el desarrollo de software, como defectos, retrasos y sobrecostos.
- **Mejora de la comunicación:** Los requisitos de calidad sirven como un lenguaje común entre los diferentes equipos involucrados en el proyecto, facilitando la comunicación y la colaboración.
- **Facilitación de las pruebas:** Los requisitos de calidad proporcionan una base para la planificación y ejecución de las pruebas, asegurando que el software se evalúe exhaustivamente.
- **Reducción de costos a largo plazo:** Al abordar la calidad desde el principio, se pueden evitar costosas correcciones y retrabajos en etapas posteriores del ciclo de vida del software.

3. Consecuencias de Ignorar los Requisitos de Calidad:

No prestar atención a los requisitos de calidad puede tener graves consecuencias para el proyecto y la organización.

- **Software defectuoso:** La falta de requisitos de calidad claros puede resultar en software con errores, fallas y un rendimiento deficiente.
- **Insatisfacción del cliente:** Un software que no cumple con las expectativas de calidad puede generar frustración, quejas y pérdida de clientes.
- **Sobrecostos:** Los defectos y errores detectados en etapas tardías del desarrollo pueden requerir correcciones costosas y retrasos en el cronograma.
- **Daño a la reputación:** Un software de mala calidad puede dañar la reputación de la organización y erosionar la confianza de los clientes.
- **Problemas legales:** En algunos casos, un software defectuoso puede generar problemas legales y responsabilidad por daños y perjuicios.

Ejemplos y Casos de Estudio:

- **Caso de Estudio: El fracaso del Therac-25.** Este acelerador lineal, utilizado en radioterapia, sufrió fallas de software que llevaron a sobredosis de radiación en pacientes, resultando en lesiones graves y muertes. Una de las causas principales fue la falta de requisitos de calidad claros y la ausencia de pruebas adecuadas.
- **Ejemplo:** Consideremos una aplicación de banca móvil. Un requisito de calidad podría ser: "La aplicación debe autenticar al usuario en menos de 3 segundos el 99.9% de las veces." Si este requisito no se especifica y no se cumple, los usuarios podrían experimentar retrasos frustrantes al acceder a sus cuentas, lo que afectaría negativamente su experiencia.

Problemas Prácticos y Ejercicios:

1. **Ejercicio:** Enumera cinco posibles atributos de calidad que serían importantes para una aplicación de comercio electrónico desde la perspectiva del usuario.
2. **Ejercicio:** Describe un escenario en el que la falta de un requisito de calidad específico podría tener un impacto negativo en una empresa de desarrollo de software.
3. **Problema:** Estás desarrollando una aplicación para un hospital. ¿Cuáles son tres atributos de calidad cruciales que deberías considerar, dado el contexto crítico de la aplicación? Justifica tu respuesta.

Soluciones:

1.
 - Facilidad de uso (intuitiva).
 - Seguridad (protección de datos personales y bancarios).
 - Rendimiento (carga rápida de páginas).
 - Confiabilidad (funcionamiento sin errores).
 - Disponibilidad (accesible 24/7).
2.
 - Escenario: No especificar requisitos de seguridad para una aplicación que almacena datos sensibles de usuarios. Impacto: Brecha de seguridad, robo de datos, daño a la reputación de la empresa, posibles demandas legales.
3.
 - **Confiabilidad:** La aplicación debe funcionar correctamente sin errores críticos, ya que las fallas podrían afectar la atención al paciente.
 - **Seguridad:** La aplicación debe proteger la información confidencial del paciente (historiales médicos, etc.) contra accesos no autorizados.
 - **Disponibilidad:** La aplicación debe estar disponible en todo momento, ya que el personal médico debe poder acceder a la información del paciente de manera oportuna.

Materiales Complementarios Recomendados:

- **Libro:** "Software Quality Engineering: Testing, Quality Assurance, and Quantifiable Improvement" por Jeff Tian.
- **Artículo:** "The Importance of Software Quality" - [Buscar artículos relevantes en IEEE Xplore o ACM Digital Library].
- **Video:** Buscar videos introductorios sobre calidad del software en plataformas como YouTube (por ejemplo, "What is Software Quality Assurance?").

Próxima Clase: Tipos de Requisitos de Calidad (Funcionales, No Funcionales, de Rendimiento, de Seguridad, de Usabilidad).