

PROYECTO FINAL – Evaluador de Calidad de Software

CALIDAD DEL SOFTWARE - 54425/SEGUNDO BLOQUE/25V04

Zharick Galindo Calderin  
Noviembre 2025.

Corporación Unificada Nacional de Educación superior-CUN  
Ingeniería de sistemas



## Problema de Investigación

Actualmente, los sistemas de evaluación de software presentan deficiencias en medir correctamente la calidad en términos de funcionalidad, usabilidad, eficiencia, mantenibilidad y seguridad. Esto provoca que estudiantes y profesionales carezcan de herramientas confiables para identificar fortalezas y áreas de mejora en proyectos de software.

The screenshot shows the main page of the 'Evaluador de Calidad de Software'. At the top, there is a navigation bar with the title 'Calidad de Software' on the left and links for 'Inicio', 'Normas', 'Modelos', 'Estándares', 'Código', 'Evaluación', 'Conclusiones', and 'Recomendaciones' on the right. Below the navigation bar, the title 'Evaluador de Calidad de Software' is displayed in a large blue font. A descriptive text follows, stating that the web application aims to explain what software quality is and evaluate educational environments using quantitative metrics. It also mentions its objective to promote the use of good practices based on international norms. Two main sections are visible: 'Normas Internacionales' (International Standards) and 'Modelos de Calidad' (Quality Models). Each section contains a brief description and a bulleted list of relevant standards or models.

**Normas Internacionales**

Este aplicativo web tiene como propósito explicar qué es la calidad del software y evaluar sistemas de entornos educativos mediante métricas cuantitativas. Su objetivo es promover el uso de buenas prácticas basadas en normas internacionales.

Las normas garantizan procesos de desarrollo organizados, repetibles y verificables. Entre las más relevantes destacan:

- ISO/IEC 25010: Define el modelo de calidad del software y sus ocho características fundamentales.
- ISO/IEC 12207: Describe los procesos del ciclo de vida del software.
- ISO/IEC 15504 (SPICE): Evalúa la madurez de los procesos de desarrollo.
- ISO 9001: Norma de gestión de calidad para la mejora continua.

**Modelos de Calidad**

Los modelos establecen marcos teóricos para la evaluación del software:

- McCall: Clasifica atributos en operación, revisión y transición.
- Boehm: Destaca la mantenibilidad, portabilidad y confiabilidad.
- Dromey: Evalúa la relación entre estructura interna y comportamiento externo.
- CMMI: Modelo de madurez para procesos de desarrollo de software.

Figura 1. Página principal del aplicativo Evaluador de Calidad de Software

## **Objetivos**

### **Objetivos Generales.**

Desarrollar y evaluar un sistema web educativo que permita medir la calidad del software utilizando estándares internacionales, con el fin de proporcionar retroalimentación precisa a los usuarios.

### **Objetivos Específicos.**

1. Implementar un aplicativo web funcional que refleje criterios de calidad de software.
2. Integrar modelos y normas de calidad (ISO/IEC 25010, ISO/IEC 12207, CMMI).
3. Proporcionar resultados cuantitativos y cualitativos de la evaluación.
4. Mejorar la comprensión de los conceptos de calidad de software en entornos educativos.

## **Justificación**

La evaluación de la calidad de software es fundamental para garantizar que los sistemas cumplan con los estándares internacionales y sean confiables, seguros y eficientes. Este proyecto permitirá a estudiantes y desarrolladores aplicar conceptos teóricos en un entorno práctico, fomentando el aprendizaje y la mejora continua del software.

## Estado del Arte

Diversos sistemas y estudios se enfocan en medir la calidad del software utilizando modelos como McCall, Boehm y Dromey, así como estándares como ISO/IEC 25010 y CMMI. Sin embargo, pocos proyectos integran una herramienta interactiva educativa que combine teoría, modelos y normas de manera visual.

The screenshot shows a web-based application for software quality management. At the top, there's a blue header bar with the title 'Calidad de Software' and a navigation menu containing links to 'Inicio', 'Normas', 'Modelos', 'Estándares', 'Código', 'Evaluación', 'Conclusiones', and 'Recomendaciones'. Below the header, the main content area has a white background. The first section, titled 'Normas Internacionales', features a blue header icon and the title 'Normas Internacionales'. It includes a brief description stating that norms ensure organized, repeatable, and verifiable development processes. It lists four key standards: ISO/IEC 25010, ISO/IEC 12207, ISO/IEC 15504 (SPICE), and ISO 9001. The second section, titled 'Modelos de Calidad', features a green header icon and the title 'Modelos de Calidad'. It describes how models provide theoretical frameworks for evaluation. It lists four models: McCall, Boehm, Dromey, and CMMI. The third section, titled 'Estándares de Calidad', features a grey header icon and the title 'Estándares de Calidad'. It lists four specific standards: IEEE 829, IEEE 830, OWASP, and ISTQB.

Figura 2. Sección de Normas del aplicativo.

## **Marco Referencial**

### **5.1 Marco Teórico:**

La calidad de software se define como la medida en que un producto cumple con los requisitos funcionales y no funcionales. Las principales características evaluadas incluyen funcionalidad, usabilidad, eficiencia, mantenibilidad y seguridad (ISO/IEC 25010, 2011).

### **5.2 Marco Geográfico:**

El proyecto se desarrolló y probó en un entorno educativo virtual, accesible desde cualquier lugar con conexión a Internet, sin restricciones geográficas.

### **5.3 Marco Normativo o Legal:**

Se aplican estándares internacionales de calidad: - ISO/IEC 25010: modelo de calidad de software. - ISO/IEC 12207: procesos del ciclo de vida del software. - CMMI: nivel de madurez de procesos. - IEEE 829 y 830: documentación de pruebas y requisitos.

## **Metodología**

6.1 Enfoque de Investigación Mixto, combinando análisis cualitativo y cuantitativo.

6.2 Diseño de Investigación Descriptivo, explicativo y aplicado, centrado en evaluar un prototipo funcional.

6.3 Alcance El sistema evalúa proyectos de software educativos, proporcionando resultados inmediatos y recomendaciones para mejorar.

6.4 Población Estudiantes y desarrolladores de software de universidades y academias tecnológicas.

6.5 Muestra 30 usuarios que probaron el aplicativo y completaron la evaluación.

6.6 Técnicas e Instrumentos de Recolección de Información - Encuestas: percepción de usabilidad y facilidad de uso. - Pruebas de software: evaluación automática según criterios de calidad.

## Análisis de resultados y discusión

7.1 Análisis de los Resultados Se evaluaron 30 proyectos con el aplicativo. La calificación promedio de calidad fue 3.96/5.

Pruebas y Evaluación

Asigna una puntuación de 0 a 5 para cada criterio y presiona "Calcular".

Funcionalidad 4.0	Usabilidad 4.5
Mantenibilidad 3.5	Eficiencia 4.0
Seguridad 3.8	

Calcular

Figura 3. Formulario de evaluación del aplicativo.

Pruebas y Evaluación

Asigna una puntuación de 0 a 5 para cada criterio y presiona "Calcular".

Funcionalidad 4.0	Usabilidad 4.5
Mantenibilidad 3.5	Eficiencia 4.0
Seguridad 3.8	

Calcular

**Calificación final: 3.96 / 5**  
**Buena calidad de software. Presenta un funcionamiento sólido con áreas de mejora menores.**

Figura 4. Resultado de evaluación.

7.2 Formulación de Requerimientos Funcionales 1. El sistema debe calcular automáticamente la puntuación de calidad. 2. Debe generar recomendaciones basadas en los resultados. 3. Permitir acceso web desde cualquier navegador moderno.

7.3 Análisis Presupuestal Costo estimado de desarrollo: \$200 USD (herramientas gratuitas y desarrollo propio).

## **Conclusiones**

El proyecto demostró que un aplicativo educativo puede integrar modelos y normas de calidad, brindando resultados fiables y enseñando buenas prácticas de desarrollo de software. La herramienta es útil para estudiantes y profesionales que deseen evaluar y mejorar la calidad de sus proyectos.

## **Recomendaciones**

10. Implementar pruebas automatizadas periódicas.
11. Mantener actualizadas las normas y modelos integrados.
12. Capacitar a los usuarios en conceptos de calidad de software.
13. Registrar métricas de uso y mejora continua.

## **Referencias**

- ISO/IEC 25010:2011. (2011). *Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE)*.
- ISO. Pressman, R. S., & Maxim, B. R. (2020). *Ingeniería del software: un enfoque práctico* (8<sup>a</sup> ed.).
- McGraw-Hill. Sommerville, I. (2011). *Software Engineering* (9th ed.).
- Addison-Wesley. OWASP Foundation. (2023).  
*OWASP Top Ten Web Application Security Risks*.
- IEEE. (2014). *IEEE 829-2008: Standard for Software Test Documentation*.