

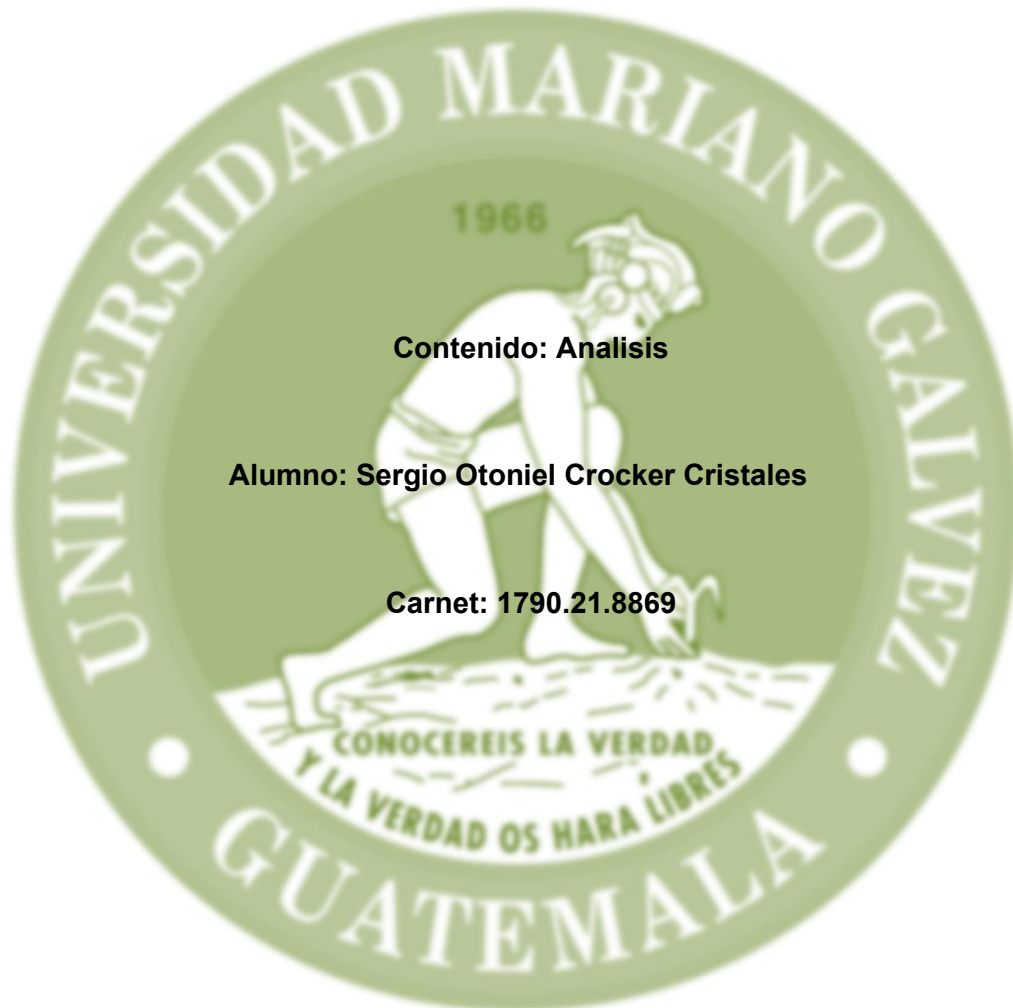
Universidad Mariano Gálvez de Guatemala

Facultad de Ingeniería en Sistemas

Curso: Aseguramiento de Calidad de Software

Catedrático: Ing. Carmelo Estuardo Mayen Monterroso

Semestre: Decimo



Contenido: Analisis

Alumno: Sergio Otoniel Crocker Cristales

Carnet: 1790.21.8869

Chiquimulilla Santa Rosa 2025

1. Estándares internacionales y calidad del software

El aseguramiento de la calidad del software (SQA) tiene como objetivo principal garantizar que el producto final cumpla con los requisitos establecidos, tanto funcionales como no funcionales. Para ello, es fundamental apoyarse en estándares internacionales ampliamente aceptados. Estos estándares definen buenas prácticas, estructuras y procesos que ayudan a las organizaciones a desarrollar software con altos niveles de calidad, seguridad, mantenibilidad y eficiencia.

- **ISO/IEC 25010:** Este modelo define ocho características de calidad del software: funcionalidad, eficiencia de desempeño, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad. Se aplica desde las etapas de diseño, ayudando a definir requisitos no funcionales. En un proyecto de desarrollo de una aplicación bancaria, por ejemplo, se puede usar para establecer criterios claros de seguridad y usabilidad. El principal beneficio de este estándar es su estructura moderna y adaptada a las necesidades actuales de los usuarios.
- **ISO/IEC/IEEE 29119:** Este conjunto de normas ofrece un marco detallado para la gestión de pruebas de software. Desde la planificación hasta la ejecución y documentación, sus cinco partes permiten estandarizar el proceso de pruebas. Se utiliza en empresas que desarrollan sistemas críticos, como aeronáutica o salud, donde la trazabilidad y rigurosidad en las pruebas es clave. Los beneficios incluyen reducción de fallos en producción y mejor documentación de resultados.
- **IEEE 730:** Esta norma define el contenido mínimo de un plan de aseguramiento de la calidad, incluyendo políticas, métricas, auditorías, procedimientos y responsables. Es útil desde el inicio del proyecto para establecer cómo se controlará la calidad. Aplicado, por ejemplo, en sistemas gubernamentales, permite garantizar conformidad normativa. Aporta claridad organizacional y mejora continua del proceso.

2. Comparativa de modelos de calidad

La evolución de la ingeniería del software ha dado lugar a diferentes modelos de calidad, cada uno con enfoques particulares sobre cómo definir y medir la calidad. A continuación se comparan tres de los más conocidos: ISO/IEC 25010, McCall y Boehm.

Modelo	Atributos Evaluados	Enfoque	Época	Ventajas / Desventajas
ISO/IEC 25010	8 características: funcionalidad, fiabilidad, seguridad, etc.	Orientado a usuario, calidad en uso y del producto.	2011	Actual, completo, adaptado a sistemas modernos. Puede ser complejo.
McCall	Corrección, eficiencia, flexibilidad, testabilidad, etc.	Divide calidad en tres perspectivas: operación, transición y revisión.	1977	Pionero, claro. Limitado en tecnologías modernas.
Boehm	Mantenibilidad, confiabilidad, eficiencia, etc.	Jerárquico, basado en objetivos del software.	1978	Estructurado. No contempla características modernas como la seguridad.

3. Roles en el aseguramiento de calidad

- Ingeniero de Pruebas: Su rol es validar que el software cumpla los requisitos mediante casos de prueba. Interactúa con desarrolladores para corregir defectos. También puede usar herramientas de automatización y pruebas de regresión. Aporta seguridad antes del lanzamiento del producto. (Pressman, 2014)•
- Analista QA: Se enfoca en la planificación de pruebas, definición de métricas, revisión de procesos y aseguramiento de que se sigan los estándares. Se comunica con stakeholders para validar que la calidad esperada esté alineada con los objetivos del negocio.
- Auditor de Calidad: Evalúa de forma independiente que se cumplan procedimientos de calidad, realiza auditorías internas y documenta hallazgos. Es clave para la mejora continua del proceso y para cumplir regulaciones en industrias críticas.

4. Análisis crítico

Ejemplo 1: En 2012, Knight Capital Group sufrió una pérdida de 440 millones USD debido a un software desplegado sin pruebas completas. El sistema ejecutó operaciones financieras no deseadas por un error de configuración que pudo haberse detectado con QA formal. (Group., 2012)

Ejemplo 2: En 2018, un sistema médico de registro en línea provocó fallos en el diagnóstico y asignación de medicamentos debido a errores no detectados. La falta de pruebas de integración y validación con usuarios reales generó demandas legales y problemas éticos.

Estos casos muestran que el aseguramiento de calidad no es una opción, sino una obligación ética y profesional en el desarrollo de software confiable, eficiente y seguro.

Bibliografía

Group., K. C. (2012). *Análisis crítico*. Alfa.

Pressman, R. S. (2014). *Roles en el aseguramiento de calidad*. Alfa .

<https://www.hiberus.com/crecemos-contigo/los-estandares-de-calidad-del-software-mas-importantes/>

<https://www.redalyc.org/journal/2654/265452747018/html/>