

Universidad Mariano Gálvez de Guatemala
Facultad de Ingeniería
Sede Chiquimulilla.

Catedrático:

Carmelo Estuardo Monterroso Mayen.

Curso:

Aseguramiento de la Calidad de Software.

Alumno:

Kevin José Santos Hernández

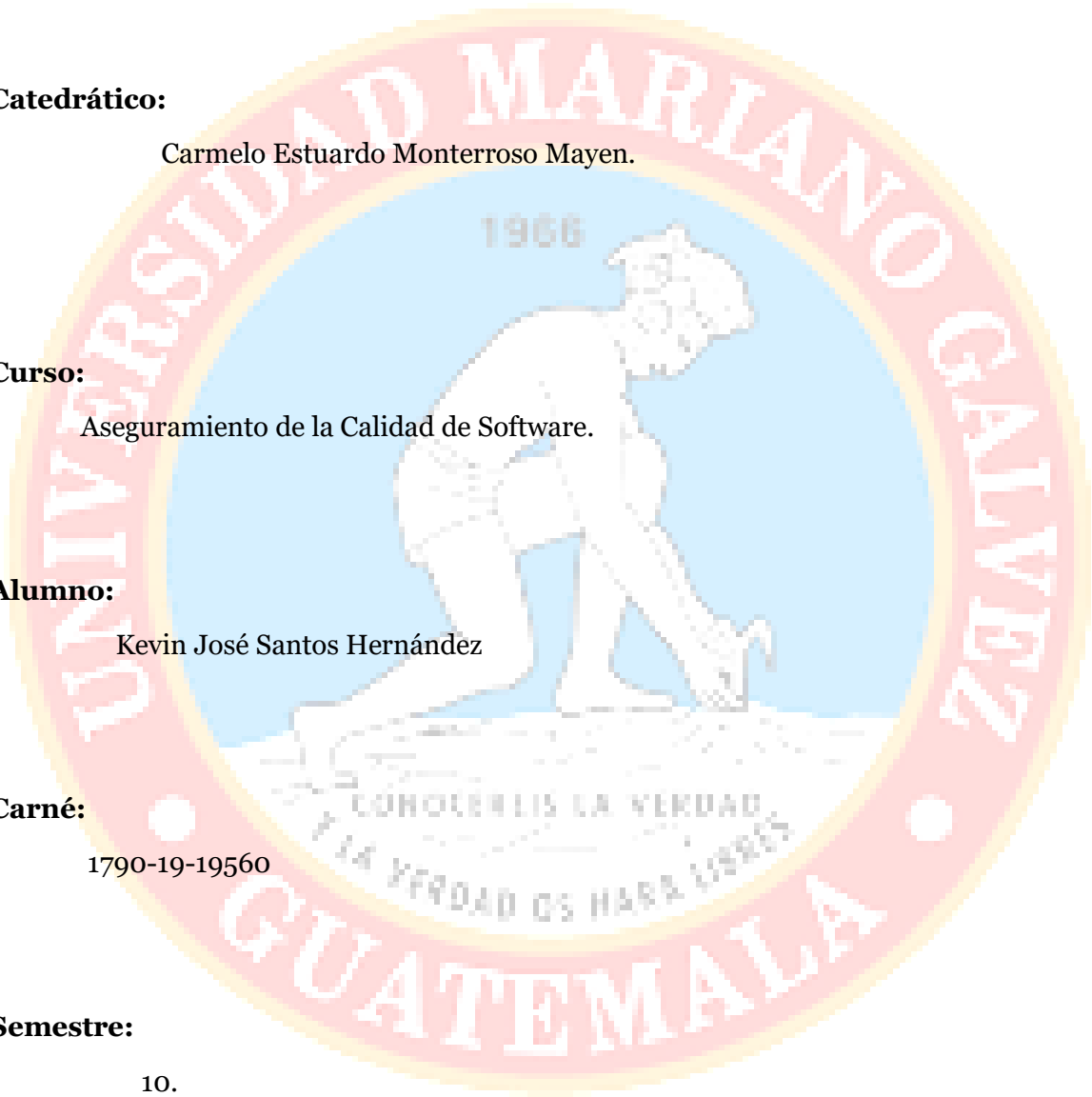
Carné:

1790-19-19560

Semestre:

10.

Chiquimulilla, 25 de julio de 2025.



Aseguramiento de la Calidad de Software Basado en Estándares Internacionales y Buenas Prácticas

1. Estándares internacionales y calidad del software

ISO/IEC 25010: Modelo de calidad del producto software

Propósito:

Este estándar define un modelo de calidad para evaluar productos de software. Su principal objetivo es proporcionar un marco para definir, medir y evaluar la calidad del software a través de ocho características: funcionalidad, rendimiento, compatibilidad, usabilidad, fiabilidad, seguridad, mantenibilidad y portabilidad.

Aplicación en proyectos reales:

En proyectos de desarrollo, ISO/IEC 25010 se utiliza como base para establecer los requisitos de calidad del producto. Por ejemplo, una empresa que desarrolla un sistema bancario puede establecer métricas de seguridad (como prevención de accesos no autorizados) y usabilidad (como facilidad de navegación para el cliente).

Beneficios:

- Mejora la satisfacción del usuario final.
- Permite una medición clara de atributos de calidad.
- Facilita la toma de decisiones basada en métricas.

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

Propósito:

Define un conjunto de estándares internacionales para los procesos de prueba de software, documentación, técnicas y gestión de pruebas. Establece un lenguaje común y prácticas recomendadas en pruebas de software.

Aplicación en proyectos reales:

En una empresa que implementa un ERP, ISO/IEC/IEEE 29119 puede ser usado para definir el plan de pruebas, los casos de prueba, criterios de aceptación y reporte de resultados.

Beneficios:

- Asegura consistencia en las pruebas.
- Facilita la planificación y trazabilidad.
- Reduce errores y retrabajos posteriores.

IEEE 730: Estándar de planes de aseguramiento de la calidad**Propósito:**

Este estándar proporciona los lineamientos para crear planes de aseguramiento de calidad (SQA). Define procedimientos, auditorías, métricas y responsabilidades relacionadas con la calidad del software.

Aplicación en proyectos reales:

En un proyecto de desarrollo de una aplicación de salud, se puede usar IEEE 730 para asegurar que todas las etapas del proyecto cuenten con revisiones de calidad, auditorías internas y control de cambios.

Beneficios:

- Establece un plan formal de aseguramiento de calidad.
- Mejora la gestión del proceso.
- Disminuye la probabilidad de fallas en etapas tardías del desarrollo.

2. Comparativa de modelos de calidad

| Modelo | Atributos evaluados | Enfoque | Época de aparición | Ventajas actuales | Desventajas actuales |
|---------------|--|----------------------------|--------------------|---|--|
| ISO/IEC 25010 | Funcionalidad, rendimiento, usabilidad, seguridad, etc. | Orientado al producto | 2011 | Estándar vigente y ampliamente reconocido | Puede requerir mucho esfuerzo de medición y adaptación |
| McCall | Corrección, fiabilidad, eficiencia, integridad, etc. | Basado en factores | Finales de los 70s | Pionero en estructurar la calidad en dimensiones claras | Enfoque algo rígido y obsoleto en entornos modernos |
| Boehm | Portabilidad, reusabilidad, facilidad de comprensión, etc. | Orientado al mantenimiento | 1978 | Bueno para proyectos que priorizan mantenibilidad | Menos aplicable en contextos ágiles y DevOps actuales |

3. Roles en el aseguramiento de calidad

a. Ingeniero de Pruebas (Test Engineer)

Responsable de diseñar, ejecutar y automatizar pruebas de software. Colabora estrechamente con el equipo de desarrollo para garantizar que las funcionalidades cumplan con los requisitos.

b. Analista de Calidad (QA Analyst)

Se encarga de definir criterios de calidad, elaborar documentación de pruebas, realizar seguimiento a defectos y asegurar el cumplimiento de procesos. Participa en revisiones de requisitos y diseño.

c. Gerente de Calidad (QA Manager)

Lidera al equipo de aseguramiento de calidad, define estrategias de prueba, asegura la alineación con los estándares y representa el área de QA frente a la gerencia y stakeholders.

Interacción con el equipo de desarrollo:

Los roles de calidad trabajan junto al equipo de desarrollo desde etapas tempranas del ciclo de vida, revisando requisitos, proponiendo mejoras, realizando pruebas continuas e impulsando prácticas como integración continua (CI).

4. Análisis crítico

Riesgos de no implementar un proceso formal de aseguramiento de calidad:

- Incremento de defectos en producción.
- Retrasos y sobrecostos por retrabajo.
- Pérdida de confianza del cliente.
- Problemas de seguridad o rendimiento crítico.

Ejemplo 1: Fallo del sistema de salud en el Reino Unido (NHS, 2017)

Un error en el sistema de citación médica causó la pérdida de miles de pruebas de cáncer, afectando diagnósticos críticos. El problema fue causado por la falta de controles de calidad en la programación y monitoreo de procesos.

Ejemplo 2: Lanzamiento fallido de la app de Uber Eats (Hipotético)

Imagina que Uber Eats lanza una nueva versión sin pruebas funcionales adecuadas. Como resultado, los pedidos no se registran correctamente y los usuarios comienzan a abandonar la app. Este fallo impacta directamente en ingresos y reputación.

Reflexión final:

El aseguramiento de la calidad no es un lujo, sino una necesidad crítica para proyectos modernos. A medida que el software se vuelve más complejo los estándares y buenas prácticas permiten mitigar riesgos, ahorrar costos y aumentar la confianza del usuario.

Bibliografía

- ISO/IEC. (2011). *ISO/IEC 25010: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models*.
- ISO/IEC/IEEE. (2013). *ISO/IEC/IEEE 29119: Software and Systems Engineering – Software Testing*.
- IEEE. (2014). *IEEE Std 730-2014: Standard for Software Quality Assurance Plans*.
- Pressman, R. S. (2010). *Ingeniería de software: Un enfoque práctico*. McGraw-Hill.
- Sommerville, I. (2016). *Ingeniería del software*. Pearson Educación.