



FACULTAD DE INGENIERÍA Y CIENCIAS APLICADAS

ISWZ3208 - Calidad de Software

Aplicación de técnicas de calidad en un proyecto

PLAN DE ACCIÓN

Said Carrera

Kristiany Cerón

David Navarrete

Mateo Cartagena

Universidad de las Américas

06/07/2025

Plan de Acción

Roles y responsabilidades

ROLES	RESPONSABILIDADES
Líder del equipo	Coordina el trabajo y supervisa la integración.
Responsable de análisis de métricas	Define e interpreta métricas de calidad.
Responsable de revisión manual	Aplica principios de Clean Code para mejorar el código.
Responsable de análisis estático	Configura y ejecuta herramientas como Checkstyle, JaCoCo y PMD.

INTEGRANTES	ROLES
Kristiany Cerón	Responsable de análisis de métricas
David Navarrete	Líder del equipo
Said Carrera	Responsable de revisión manual
Mateo Cartagena	Responsable de análisis estático

Problemas de calidad identificados

CATEGORÍA	DESCRIPCIÓN	HALLAZGOS
Nombres confusos	Las variables y métodos tienen nombres poco descriptivos.	Variables como “t” o “tasks” poco claras
Métodos largos	Algunas funciones realizan múltiples tareas y carecen de cohesión.	Las impresiones están en cada método “System.out.println(“Task added.”);”
Falta de validaciones	No se detecta control de errores suficientes en los métodos	No hay validación de tareas vacías en “public void addTask (String t)”
Falta de cobertura de pruebas	No hay pruebas unitarias para todas las funcionalidades.	No hay ningún test que valide comportamiento del sistema.

Falta de análisis estático	No hay detección de errores de estilo o bugs	No se implementan plugins de análisis de código estático
-----------------------------------	--	--

Estrategias y métricas propuestas

PROPUESTAS	APLICACIÓN
Principio SOLID - SRP	Separar responsabilidades para asegurar que cada clase tenga una única tarea.
Refactorización del código	Creación de clases independientes, extracción de responsabilidades, validación de entradas y diseño de interfaces.
Análisis de código estático	Pruebas unitarias, verificación de errores, cobertura y estilo de código y CI/CD.

Métricas para evaluación de calidad

MÉTRICA	HERRAMIENTA	OBJETIVO
Cobertura de código	Jacoco	>= 80% de cobertura
Estilo de código	Checkstyle	0 errores de estilo
Defectos comunes	PMD	0 violaciones críticas del diseño
Funcionalidades correctas	JUnit5	100 % de pruebas implementadas
Automatización	GitHub Actions	Ejecución correcta del pipeline y análisis en cada commit con resultados visibles.

Enfoque PDCA

1. Planificar (Plan)

- a. **Análisis:** Verificación del código base
- b. **Problema detectado:** El sistema base es difícil de mantener, poco legible, sin validaciones ni pruebas. Lo que compromete su escalabilidad y fiabilidad.
- c. **Acciones de mejora:** Refactorización del código, implementación de pruebas y validaciones, integración de herramientas de análisis estático y automatización CI/CD.

2. Hacer (Do)

- a. **Implementación:** División de responsabilidades, creación de interfaces, elaboración de Tests y configuración de plugins.

3. Verificar (Check)

- a. **Indicadores de métricas:**
 - i. $\geq 80\%$ de cobertura
 - ii. 0 errores de estilo
 - iii. 0 violaciones críticas del diseño
 - iv. 100 % de pruebas implementadas
 - v. GitHub Actions funciona sin fallos en cada push.


4. Actuar (Act)

- a. **Ajustes y mejora continua:**
 - i. Si no se alcanzan las métricas de corrección de errores y diseño, refactorizar la estructura del código.
 - ii. Ampliar pruebas para incrementar la cobertura de código.
 - iii. Corroborar que el pipeline se ejecute correctamente, caso contrario validar los errores detectados
 - iv. Documentar y reutilizar interfaces para futuras extensiones.

Evidencias

- a) **Evidencias para el Planteamiento de Roles y Objetivos: Roles asignados a cada miembro del equipo y objetivos asignados con justificación.**

INTEGRANTES	ROLES	JUSTIFICACIÓN
Kristiany Cerón	Responsable de análisis de métricas	Tiene un enfoque detallado y analítico, lo que le permite interpretar indicadores de calidad con precisión. Su responsabilidad es definir métricas clave (cobertura, estilo, errores) y evaluar si se alcanzan los objetivos del proyecto.
David Navarrete	Líder del equipo	Posee habilidades de liderazgo y organización, lo que facilita la planificación y coordinación del equipo. Su rol garantiza una correcta



		integración de tareas, seguimiento del progreso y toma de decisiones efectivas.
Said Carrera	Responsable de revisión manual	Tiene buen criterio para detectar código confuso o innecesariamente complejo. Su responsabilidad es aplicar principios de Clean Code, mejorar la legibilidad del código fuente y asegurar que se cumplan buenas prácticas de programación.
Mateo Cartagena	Responsable de análisis estático	Tiene más conocimiento técnico en herramientas de análisis de código. Su rol es configurar y ejecutar herramientas como Checkstyle, PMD, SpotBugs y JaCoCo para detectar errores, violaciones de estilo y evaluar la cobertura de pruebas.