



TAREA 01

Estudiantes:

Nicolas Alexander Herrera Rincon

Julian Alejandro Archila Caro

Stiven Leonardo Sanchez Leon

Santiago Montealegre Romero

Docente: Oscar Eduardo Alvarez Rodriguez

Fecha de realización:05/09/2025

Fecha de entrega: ?/09/2025

1. ¿Por qué es esencial que los ingenieros de software prioricen el bienestar público?

El Código establece en el Principio 1 (PUBLIC) que la responsabilidad principal del ingeniero de software es actuar en concordancia con el interés público. Esto significa asumir plena responsabilidad sobre el propio trabajo (1.01), equilibrar los intereses personales, del cliente o del empleador con el bien común (1.02) y aprobar software únicamente si es seguro, cumple con las especificaciones y no afecta negativamente la calidad de vida, la privacidad o el medio ambiente (1.03). Asimismo, exige divulgar cualquier peligro potencial vinculado al software (1.04), colaborar en la solución de problemas de gran impacto público (1.05) y evitar declaraciones engañosas (1.06). También se resalta la necesidad de considerar factores sociales y de accesibilidad (1.07) y de contribuir voluntariamente al beneficio colectivo mediante la profesión (1.08).

2. ¿Qué significa mantener la calidad del producto en software?

De acuerdo con el Principio 3 (PRODUCT), mantener la calidad implica

asegurar que los productos y sus modificaciones cumplan los más altos estándares profesionales. Esto abarca el compromiso con entregar software de alta calidad, dentro de costos y plazos razonables, aclarando siempre los compromisos asumidos (3.01). Supone definir objetivos alcanzables (3.02), contemplar factores legales, éticos y ambientales (3.03) y trabajar únicamente en proyectos para los cuales se tenga la preparación adecuada (3.04). También requiere aplicar métodos apropiados (3.05), seguir estándares profesionales pertinentes (3.06) y comprender a fondo las especificaciones (3.07 y 3.08). El ingeniero debe proporcionar estimaciones realistas con sus márgenes de incertidumbre (3.09), garantizar pruebas y revisiones adecuadas (3.10) y mantener documentación completa (3.11). Además, la calidad incluye proteger la privacidad (3.12), usar datos válidos y legales (3.13), velar por la integridad de la información (3.14) y tratar el mantenimiento con el mismo rigor que el desarrollo inicial (3.15).

3. ¿Qué implica el compromiso con el aprendizaje continuo?

El Principio 8 (SELF) señala que el aprendizaje permanente es un deber profesional. Esto implica actualizarse en áreas clave como análisis, diseño, desarrollo, pruebas y gestión de proyectos (8.01), así como mejorar continuamente la capacidad de producir software seguro, confiable y útil dentro de límites de costo y tiempo razonables (8.02). El compromiso también incluye perfeccionar la producción de documentación clara y precisa (8.03), fortalecer la comprensión del software y su entorno de aplicación (8.04) y mantenerse informado sobre estándares y legislación aplicables (8.05). Además, exige conocer en profundidad este mismo Código, su interpretación y aplicación práctica (8.06), evitar cualquier trato injusto motivado por prejuicios (8.07), abstenerse de inducir a otros a violar el Código (8.08) y reconocer que incumplirlo personalmente es incompatible con la práctica profesional (8.09).

4. ¿Qué prácticas promueven la transparencia y la integridad en el desarrollo de software?

La transparencia y la integridad se sustentan en el Principio 4 (JUDGMENT), que requiere mantener independencia y objetividad en las decisiones profesionales.

Esto implica fundamentar los juicios técnicos con criterios que protejan los valores humanos (4.01), aprobar únicamente documentos que estén bajo la propia supervisión o dentro del área de competencia (4.02) y conservar objetividad al evaluar proyectos (4.03). Igualmente, el Código prohíbe explícitamente prácticas financieras indebidas como sobornos o doble facturación (4.04), obliga a revelar conflictos de interés inevitables (4.05) y a rechazar la participación en organismos donde dichos conflictos no sean transparentes (4.06). La integridad también se relaciona con evitar engaños en declaraciones públicas sobre software (1.06), mantener documentación completa sobre problemas y soluciones adoptadas (3.11) y presentar de forma precisa las características del software, evitando afirmaciones falsas o especulativas (6.07).

5. ¿Por qué es importante el respeto y apoyo entre colegas en el campo de la ingeniería de software?

El Principio 7 (COLLEAGUES) establece que la relación entre colegas debe estar marcada por la equidad y el apoyo mutuo. Esto supone alentar a otros profesionales a seguir el Código (7.01) y contribuir a su desarrollo profesional (7.02). También implica reconocer adecuadamente

las aportaciones de otros, evitando apropiarse de méritos ajenos (7.03), y revisar el trabajo de forma objetiva y documentada (7.04). El respeto se refleja en dar un espacio justo a las opiniones y preocupaciones de los colegas (7.05) y en compartir conocimientos sobre prácticas estándar y medidas de seguridad (7.06). A su vez, el apoyo se concreta en no intervenir de forma injusta en la carrera de otro profesional (7.07), salvo cuando el interés del cliente, empleador o público lo justifique, y en reconocer cuándo es necesario acudir a especialistas externos en áreas fuera de la propia competencia (7.08).

6. Referencias

[1] D. Gotterbarn, K. Miller, y S. Rogerson, “Software Engineering Code of Ethics,” *Communications of the ACM*, vol. 40, núm. 11, pp. 110–118, Nov. 1997, doi: 10.1145/265684.265699