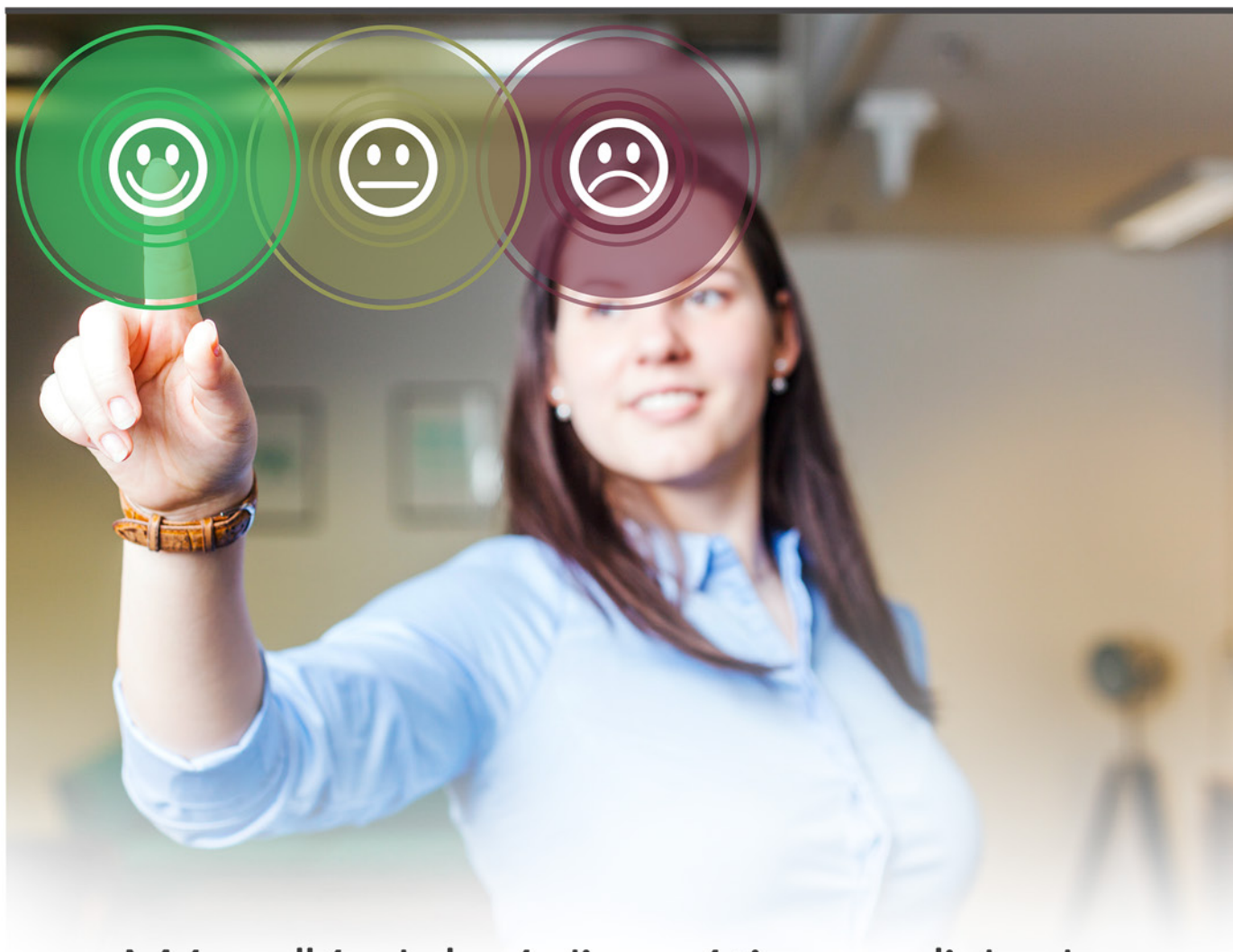




Más allá del código: ética, calidad y satisfacción del cliente

XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

Autores:

Yeison David Chavarro Restrepo
Karina Audoro Vasquez
Valery Botello Artunduaga
Daniela Trujillo Correa

Resumen

Este artículo explora cómo la ética y la calidad se relacionan dentro del proceso de desarrollo de software, a partir de una revisión integrativa de diferentes fuentes. Para ello, se consultaron y analizaron diversas publicaciones digitales y artículos recientes que reflexionan sobre este tema desde perspectivas técnicas y humanas.

Los resultados muestran que en el desarrollo de software la responsabilidad de la tarea, la toma de decisiones de calidad en este proceso, así como la formación en las habilidades técnicas necesarias para que cualquier programa funcione correctamente son importantes.

La ética no se reduce a proteger datos o a garantizar la seguridad de la información, sino que influye sobre cómo se construye la calidad del producto. La usabilidad, fiable de sistema, trabajo en equipo, innovación responsable, son aspectos éticos en el desarrollo de software que afectan a la experiencia del usuario y nivel de satisfacción en el producto final.

La ética no es un añadido a la calidad del software, sino una base fundamental para crecer soluciones tecnológicas que sean confiables, útiles y centradas en las necesidades reales de ser humano. Un desarrollo ético se convierte en el camino para lograr no sólo productos eficientes, sino también software que genera confianza, valor y bienestar social.

Palabras clave:

Ética; Calidad del software; Confianza; Seguridad de la información; Experiencia del usuario; Trabajo en equipo; Innovación responsable.

Abstract

This article explores how ethics and quality are connected within the software development process, based on an integrative review of different sources. For this purpose, various digital publications and recent articles were consulted and analyzed, approaching the topic from both technical and human perspectives.

The results show that in software development, responsibility, decision-making, and proper technical training are essential to ensure that any program works effectively. Ethics is not limited to protecting data or guaranteeing information security; it also influences how product quality is built. Usability, system reliability, teamwork, and responsible innovation are ethical aspects that directly affect the user experience and the level of satisfaction with the final product.

Ethics is not an addition to software quality but rather a fundamental basis for creating technological solutions that are trustworthy, useful, and centered on real human needs. Ethical development thus becomes the path toward achieving not only efficient products but also software that generates trust, value, and social well-being.

Keywords:

Ethics; Software quality; Trust; Information security; User experience; Teamwork; Responsible innovation.

Introducción

En la actualidad vivimos rodeados de tecnologías, usamos aplicaciones, redes sociales y plataformas creadas por ingenieros de software. Aunque deben ser buenos desarrollando, es muy importante que comprendan lo que está bien y lo que está mal en su profesión; por eso la ética es un factor muy valioso en el desarrollo de software. (Spin ello, 2001).

Aristóteles define la ética como el camino hacia la felicidad mediante el cultivo de virtudes, practicando acciones correctas y buscando el justo medio para beneficiar tanto al individuo como a la comunidad (Soto, 2008). En este sentido, Johnson (1992) afirma que los ingenieros de software, por su conocimiento especializado y su impacto social, deben orientar su trabajo al bien común, protegiendo la seguridad y el bienestar de las personas frente a posibles fallas, brechas de seguridad o impactos sociales graves.

El trabajo de los ingenieros de software impacta significativamente la vida de las personas, ya que deben proteger datos sensibles y evitar prácticas como copiar código ajeno o crear sistemas dañinos, lo que forma parte de la ética profesional (Andorno & Biller-Andorno, 2017). Esta ética también influye en la calidad del software, fomentando la creación de código seguros, confiables, bien documentados, y respetuosos de la privacidad de los usuarios (Rincón et al., 2019).

Don Gotterbarn, del código de ética ACM/IEEE, sostiene que la ética es fundamental para la calidad del software que debe ser no solo funcional y eficiente, sino también seguro, confiable y socialmente responsable (ACM/IEEE Code of Ethics, 2020). La ética guía a los ingenieros frente a presiones como reducir pruebas o usar código ilícitado, ya que muchos errores comienzan por decisiones poco éticas como ignorar advertencias o no documentar adecuadamente (Research Gate, 2022).

En este trabajo destacamos la importancia que tiene la responsabilidad en el desarrollo de software, argumentando que programar correctamente no es suficiente sin la toma de decisiones éticas por parte del ingeniero a la hora de proteger a los usuarios y a la sociedad. Solo de esta forma, estos profesionales de la informática podrán apoyar en la misión de crear tecnologías confiables, justas y responsables (Camacho, 2023).

Metodologías

Este estudio adopta un enfoque de revisión integrativa, buscando desmenuzar, inspeccionar y fusionar los datos actuales sobre cómo la ética se entrelaza con la administración de la calidad durante la creación de software. Para lograrlo, se rastreó material bibliográfico relevante, como trabajos académicos y publicaciones periódicas, además de recursos en línea, utilizando plataformas de datos científicos y portales web especializados.

Esta forma de encarar la investigación ayuda a impulsar una perspectiva analítica y detallada de las nociones clave, las discusiones y los conflictos que emergen en relación con la calidad y la ética dentro del entorno del desarrollo de software.

Resultados y Discusiones

La ética como base para un software confiable

Debatir sobre la ética en la creación de software no es solo para los cerebritos de la tecnología; en realidad, impacta el día a día de muchísimas personas. Las apps de salud, la banca online y las redes sociales se han transformado en herramientas indispensables, y cada vez que las usamos, damos por sentado que esos sistemas se han diseñado con responsabilidad y un sólido criterio ético.

Crear software de forma ética implica priorizar la protección de los datos, resguardar la privacidad y ser claros sobre cómo se gestiona la información. Si estos principios se dejan de

lado, no solo se pone en entredicho la confianza en un producto en particular, sino que además se socava

la reputación de las compañías y de todo el sector tecnológico en su conjunto (Morley et al., 2020; Greene et al., 2021).

Diversas investigaciones señalan que la ética debería entenderse no como una mera lista de normas impuestas, sino como una dedicación continua en la labor de los creadores de software. Integrar principios como la integridad, la equidad y la consideración en cada elección convierte el software en un medio para impulsar el progreso social y prevenir efectos dañinos, como la marginación digital o el abuso de datos (Floridi & Cows, 2020; Mittelstadt, 2021).

De este modo, la ética se vuelve un elemento esencial del proceso de creación tecnológica, arraigándose en la cultura laboral y trascendiendo un mero formalismo.

En este sentido, hablar de ética en la creación de software no es una cuestión teórica, sino una exigencia tangible. Garantizar que los sistemas digitales se diseñen con responsabilidad y respeto hacia los usuarios genera un clima de seguridad que consolida el vínculo entre la tecnología y la sociedad (Jobin et al., 2019; Zwitter & Gstrein, 2020).

La calidad de software desde el punto de vista del usuario

Cuando una persona en la evaluación de un software hace una valoración, lo relevante es cómo ha sido capaz de vivir la experiencia de utilizarlo en estos primeros momentos. Más allá de las características técnicas, como el mantenimiento o la eficiencia, es importante que el usuario perciba la facilidad de su uso, la fiabilidad o la utilidad real que tiene (ISO/IEC 25010, 2020).

El concepto de calidad está muy por encima de la usabilidad. Un software que se comporta bien técnicamente hablando puede fallar si la aplicación es muy complicada o poco

intuitiva y no crea una experiencia de usuario positiva. Por esta razón, su usabilidad y su adaptación a la tecnología son necesarias. Estudios recientes en ingeniería de software han mostrado que la calidad percibida por los usuarios es la respuesta tanto a la capacidad de cumplir expectativas, como a la satisfacción al utilizar el software (Bevan, 2022).

Adicionalmente, otro aspecto importante es la confianza. Los usuarios consideran importante saber que el software protege su información, que no genera interrupciones y que cumple sus promesas. La calidad percibida, mejorada mediante seguridad y transparencia, tiene cada vez más importancia, ya que los errores o fallos pueden minar la confianza del cliente a la hora de usar el software (Alonso et al., 2021). Por lo tanto, la calidad no se limita a que un software funcione bien, sino que también es importante que ofrezca confianza, satisfacción e higiene a la hora de interactuar con él.

La experiencia del cliente como guía de mejora

Hoy en día el verdadero valor de un software no está limitado a su funcionalidad, sino que está dado por la vivencia que ofrece a quienes lo utilizan. La experiencia del cliente se ha erigido como principal baluarte para la continua mejora de cualquier producto digital. Esto nos permitirá distinguir tanto los aspectos exitosos como aquellos en los que se hace hincapié ante las necesidades de los usuarios (Kujala et al., 2020). Escuchar al cliente no se limita a recoger

las impresiones de este, sino que es una forma de orientarlo hacia el desarrollo de un software más amigable, usable y confiable.

Cuando una empresa pone al cliente en el centro se convierte en capaz de convertir información poco estructurada en decisiones eficaces de diseño y modernización. Un claro ejemplo son los recientes estudios que permiten sostener que aquellas empresas que apuestan por la experiencia del usuario pueden aumentar considerablemente la fidelización y reducir los

niveles de abandono en entornos de alta competencia (Komssi et al., 2021). Este enfoque también favorece la innovación ya que muchas de las mejoras que más valoran los usuarios son directamente las que les son expresadas por las personas que utilizan el software en su día a día.

Por otro lado, la experiencia del cliente permite exteriorizar aquello que se ha denominado como "barómetro social" con referencia a la visión general del uso del producto más allá del aspecto técnico. No se trata únicamente de que el sistema ejecute sus funciones, sino de cómo las personas lo adoptan en su día a día, si lo encuentran comprensible o intrincado, si inspira confianza o, por el contrario, exaspera. Esa perspectiva humana es esencial para identificar aspectos que rara vez se descubren en pruebas internas. Incorporar este tipo de retroalimentación continua se traduce en un proceso de mejora incesante, acorde tanto a las expectativas del mercado como a los valores de calidad que fomenta la ética en el desarrollo de software (Lallemant et al., 2021).

En resumen, un software que presta atención aprende y se adapta según la experiencia de sus clientes no solo optimiza su rendimiento, sino que establece lazos fuertes y duraderos. La experiencia del cliente, por ende, no debe concebirse como un mero añadido, sino como la guía que señala el camino del crecimiento y la credibilidad en cada nueva entrega del producto.

Trabajo en equipo y cultura de mejora continua

La calidad en el desarrollo de software no depende únicamente de la tecnología que utilices, sino de cómo trabajan en equipo y de lo dispuesto que estén a mejorar. Cuando un equipo se comunica, trabaja en conjunto para resolver problemas y comparte conocimientos, hace su trabajo mejor y con menos errores. Cuando desarrolladores, testers, diseñadores y managers trabajan en equipo, la calidad del producto final se verá mejorada (Gonçalves et al., 2021).

El practicar una cultura de mejora continua significa revisar y actualizar procesos en pequeños ciclos de tiempo. Métodos ágiles como Scrum o Kanban favorecen revisiones cortas y frecuentes durante el proceso, lo que acaba ayudando a detectar defectos y oportunidades de mejora de manera temprana (Rico et al., 2022). La mejora continua no es únicamente una técnica, sino también una mentalidad que ayuda a los equipos a aprender de la experiencia mientras continúa dedicando lo mejor de sí mismos aprendiendo nuevas maneras de desarrollar productos software.

Diversas investigaciones han afirmado que grupos de personas en un entorno colaborativo y con una cultura de aprendizaje son aquellos que logran más satisfacción para

las personas que lo forman, además de producir productos con mayor fiabilidad y conforme a las exigencias del cliente. La sinergia entre el trabajo en equipo y la mejora continua no solamente disfruta de una implicación sobre la calidad del software, sino que también contribuye a fomentar el desarrollo profesional de las personas implicadas y de la organización misma (Martínez & Aguilar, 2023).

Normas y guías como apoyo, no como carga

Las normas y guías relacionados con el desarrollo de software suelen ser percibidas simplemente como un conjunto de restricciones que limitan la capacidad de los equipos de trabajo para ser creativos. No obstante, en su empeño, estas herramientas se comportan como un diseño de referencia que ayuda a la organización, la seguridad y el tener éxito en el producto final.

A modo de ejemplo, las normas ISO/IEC 25010 ayudan a decidir unos criterios de calidad como fiabilidad, usabilidad, mantenibilidad, lo que hace que los equipos no trabajen sin rumbo, sino que lo hagan con unos parámetros que permiten medir su avance de manera

objetiva (ISO, 2022). Adicionalmente, el uso de estas normas ayuda a generar la confianza del cliente al conocer que el software diseñado sigue unas directrices que son reconocidas a nivel internacional.

Las guías más que un obstáculo forma parte de las posibilidades de pasar a ser aliadas de los procesos de mejora. Estudios recientes muestran que las empresas que se han decidido por el uso de normas de comportamiento menos rígidas y adaptadas a la persona alcanzan mayores niveles de satisfacción de los usuarios y de los equipos de trabajo (García & Torres, 2023).

De esta manera, si se dispone de guías, éstas dejan de ser una carga para ser un instrumento de mejora, que contribuye al desarrollo técnico y a la experiencia de uso. Así lo entienden Ramesh y Rao (2023), para quienes el reto no consiste en adecuarse a la letra de las normas sino en darles vida, es decir, otorgarles un marco de aprendizaje y de orientación, donde cada uno de los requisitos pueda adaptarse a las necesidades del proyecto. En este sentido, el verdadero valor de las normas es cuando sustentan la toma de decisiones y la gestión de riesgos.

Innovación responsable

La innovación es uno de los pilares más relevantes del desarrollo tecnológico actual. Sin embargo, cuando se habla de innovación en el software no es suficiente pensar en nuevas funciones o en el uso de nuevas herramientas avanzadas, sino que es preciso hacerlo desde la responsabilidad.

Es decir, para cada avance no puede ignorarse un análisis de su viabilidad técnica o la manera en que se va a impactar, sino que debe incluirse también el análisis de cómo afecta a la sociedad o a la privacidad de los usuarios, así como su sostenibilidad. La discusión en torno

a la ética tecnológica ha sido en este sentido un aporte esencial para la manera en que se requiere innovar. Y al mismo tiempo, se intenta evitar recurrir a soluciones que puedan dar lugar a desigualdades o asumir riesgos innecesarios (Morales & Silva, 2022).

La innovación responsable sugiere aplicar ahí una mirada crítica a los productos que se van a desarrollar: ¿qué implicaciones tendrán para los usuarios? ¿favorecerá la inclusión o limitará? ¿está alineada con principios de transparencia y de seguridad? A partir de estas preguntas se puede trabajar de modo que el intercambio permita desarrollar capacidades de los usuarios y proponer resultados que aporten valor real.

Un estudio de Martínez y López (2023) afirma que aquellas organizaciones que introducen prácticas de innovación responsable tienden a elevar la confianza de las personas y fortificar la reputación que tienen en el mercado. Esto es, que la innovación tiene que ocupar la atención no solo del "ser los primeros" o de desarrollar pronto productos o los servicios, sino de hacer de las propuestas de innovar una oportunidad de creación de soluciones que signifiquen una mejora en las condiciones de vida de las personas y el conjunto de la sociedad.

Uno de los ejemplos cabría ser el creciente interés por las tecnologías verdes y accesibles, que persiguen el objetivo de disminuir la huella planetaria y abrir el acceso a las personas con discapacidades. Estas prácticas, muestran que es posible incorporar la tecnología sin tener que renunciar a la ética y a la sostenibilidad (UNESCO, 2023).

Por lo tanto, la innovación responsable mostraría una búsqueda del equilibrio entre la creatividad, la ética y la responsabilidad social. De esta manera, se conseguiría un software innovador, justo, seguro y orientado hacia las personas.

Conclusiones

La ética y la calidad del software se manifiestan en que el artículo técnico no solo debe funcionar, sino que el concepto de confianza y seguridad con cualidad a los destinatarios tiene que ir aparejado. La calidad tiene sentido real si se apoya en un conjunto de principios éticos que generen confianza en la información, garanticen la protección de la información personal y sean fáciles de utilizar para los usuarios. Los resultados revisados demuestran que escuchar al usuario, colaborar en equipo y mantener una cultura de mejora con los usuarios son prácticas que fortalecen tanto la confianza social del software como su calidad técnica. También las normas y guías de calidad son herramientas que pueden ser útiles para guiar procesos y consolidar e impulsar la credibilidad de las soluciones digitales. Por último, la innovación responsable se convierte en un camino que nos permite equilibrar creatividad y sostenibilidad, de modo que el software no solo alcance altos estándares técnicos, sino que también aporte, en definitiva, valor real e inclusivo a la sociedad.

Referencias

- Jobin, A., Ienca, M., & Vayena, E. (2019). The global landscape of AI ethics guidelines. *Nature Machine Intelligence*, 1(9), 389–399. <https://doi.org/10.1038/s42256-0190088-2>
- Komssi, M., Kiljander, J., Raatikainen, M., & Parviainen, P. (2021). Customer experience-driven development in digital services. *Journal of Systems and Software*, 176, 110941.
- Kujala, S., Roto, V., Väänänen-Vainio-Mattila, K., & Karapanos, E. (2020). UX and customer experience: How they relate and differ. *Behaviour & Information Technology*, 39(1), 1–17. <https://doi.org/10.1080/0144929X.2019.1667937>
- Lallemand, C., Gronier, G., & Koenig, V. (2021). User experience: A key concept for digital product improvement. *International Journal of Human–Computer Studies*, 150, 102605. <https://doi.org/10.1016/j.ijhcs.2021.102605>

- Martínez, F., & Aguilar, D. (2023). Organizational culture and teamwork in software quality enhancement. *Software Quality Journal*, 31(4), 899–915.
- Martínez, H., & López, C. (2023). Responsible innovation practices in software development companies. *Journal of Responsible Innovation*, 10(2), 215–229.
- Mittelstadt, B. D. (2021). Principles alone cannot guarantee ethical AI. *Nature Machine Intelligence*, 3(10), 854–855. <https://doi.org/10.1038/s42256-021-00359-7>
- Morales, J., & Silva, R. (2022). Ethical approaches to responsible innovation in the digital age. *Ethics and Information Technology*, 24(2), 157–170.
- Morley, J., Floridi, L., Kinsey, L., & Elhalal, A. (2020). From what to how: An initial review of publicly available AI ethics tools, methods and research to translate principles into practices. *Science and Engineering Ethics*, 26(4), 2141–2168.
- Ramesh, K., & Rao, S. (2023). Adapting ISO/IEC standards for dynamic software environments. *International Journal of Information Systems*, 18(2), 85–99.
- Rico, J., Hernández, P., & Torres, A. (2022). Continuous improvement practices in agile environments: A case study. *International Journal of Information Systems*, 18(2), 56–
- UNESCO. (2023). Recommendation on the ethics of artificial intelligence: A global framework for responsible innovation. UNESCO Publishing.
- Zwitter, A., & Gstrein, O. J. (2020). Big data, privacy and COVID-19 – learning from humanitarian expertise in data protection. *Journal of International Humanitarian Action*, 5(4), 1–17. <https://doi.org/10.1186/s41018-020-00072-6>
- Alonso, J., Pérez, M., & Rodríguez, L. (2021). User trust and software reliability in digital environments.

Journal of Software Systems, 45(2), 133–148.

<https://www.acm.org/binaries/content/assets/code-of-ethics/se-code-spn.pdf>

Bevan, N. (2022). Usability and user experience as quality components in software engineering.

International Journal of Human–Computer Interaction, 38(5), 421–433.

https://www.researchgate.net/publication/365232668_Ethics_in_Science_and_Technology_Policy-Making_A_Proposed_Normative_Framework

Floridi, L., & Cowls, J. (2020). A unified framework of five principles for AI in society.

http://www.scielo.org.co/scielo.php?pid=S190038032017000100236&script=sci_arttext

García, A., & Torres, P. (2023). Flexible adoption of ISO standards in software companies: Benefits and challenges. Software Quality Journal, 31(3), 541–558.

https://www.researchgate.net/publication/319344598_Computer_Ethics

Gonçalves, L., Souza, R., & Almeida, T. (2021). Collaboration and communication in agile software teams: Impacts on quality. Journal of Software Engineering, 27(3), 145– 159.

Harvard Data Science Review, 2(1). <https://doi.org/10.1162/99608f92.8cd550d1>

International Organization for Standardization.

<https://mail.cienciamatriarevista.org.ve/index.php/cm/article/view/1495/2498>

ISO/IEC 25010. (2020). Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models.

<https://www.proquest.com/openview/fdd917c9e0dbb6018e73d2e11d53229f/1?pqorigsite=gscholar&cbl=1820941>