



Editorial para el número especial de la 14a Conferencia Internacional sobre Investigación e Innovación en Ingeniería de Software (CONISOFT'26)

Editorial for the special issue of the 14th International Conference on Software Engineering Research and Innovation

Juan Carlos Guzmán Preciado

Facultad de Ingeniería Mochis, Universidad Autónoma de Sinaloa, México

dr_guzman@uas.edu.mx

ORCID: 0000-0001-6534-876X

Reyes Juárez-Ramírez

Universidad Autónoma de Baja California, Tijuana, Baja California, México

reyesjua@uabc.edu.mx

ORCID: 0000-0002-5825-2433

Alan Ramírez-Noriega

Facultad de Ingeniería Mochis, Universidad Autónoma de Sinaloa, México

alandramireznoriega@uas.edu.mx

ORCID: 0000-0002-8634-9988

Carlos Alberto Fernández y Fernández

Universidad Tecnológica de la Mixteca, Oaxaca, México

caff@mixteco.utm.mx

ORCID: 0000-0002-1586-8772

Carolina Tripp-Barba

Facultad de Informática Mazatlán, Universidad Autónoma de Sinaloa, México

ctripp@uas.edu.mx

ORCID: 0000-0002-4811-0247

 <https://doi.org/10.36825/RITI.14.35.001>

Resumen: El software está presente hoy en casi todas las actividades humanas y, con ello, la ingeniería de software ha dejado de ser un asunto meramente técnico para convertirse en un factor decisivo del desarrollo social y económico. En una época de cambios tecnológicos continuos, esta disciplina impulsa la transformación digital de la academia, la industria y el sector público. Con ese punto de partida, este número especial reúne los artículos de investigación en español seleccionados para la **14ª Conferencia Internacional sobre Investigación e Innovación en Ingeniería de Software (CONISOFT'26)**. Los trabajos reunidos aquí reflejan tendencias recientes y retos abiertos, y proponen respuestas concretas que, en conjunto, buscan afianzar la comunidad científica, tender puentes entre la academia y la industria y ampliar el conocimiento en ingeniería de software tanto en el ámbito nacional como internacional.

Palabras clave: *Ingeniería de Software, Innovación, Tecnología, Aplicaciones, Colaboración.*

Abstract: Software is part of nearly every human activity today, and Software Engineering has therefore become more than a technical matter: it is now a decisive factor in social and economic development. In a time of continuous technological change, the discipline drives the digital transformation of academia, industry, and the public sector. With that in mind, this special issue gathers the research articles in Spanish selected for the **14th International Conference on Software Engineering Research and Innovation (CONISOFT'26)**. The works collected here reflect recent trends and open challenges, and they offer concrete responses that, taken together, seek to strengthen the scientific community, build bridges between academia and industry, and broaden knowledge in Software Engineering at both the national and international levels.

Keywords: *Software Engineering, Innovation, Technology, Applications, Collaboration.*

1. Editorial

La Ingeniería de Software (IS) vive un momento de cambio permanente. Su tarea ya no se limita a analizar, diseñar, construir y mantener sistemas: hoy también implica imaginar soluciones que sean inteligentes, seguras y sostenibles, y que tengan un efecto real en la economía, la educación, la salud, la industria y la vida diaria de las personas. En un contexto donde la inteligencia artificial, la computación en la nube, el Internet de las Cosas y la ingeniería de datos están presentes en casi cualquier proyecto, el software se ha vuelto un motor de innovación y de cambio social. Las aplicaciones que usamos a diario y las infraestructuras que sostienen a gobiernos y empresas dependen, en gran medida, de decisiones que toma la IS. Por eso se ha convertido en un eje transversal del desarrollo científico y tecnológico.

La **14ª Conferencia Internacional sobre Investigación e Innovación en Ingeniería de Software (CONISOFT'26)** busca ser un punto de encuentro para académicos, investigadores, profesionales de la industria y representantes del sector público. Su propósito es abrir un espacio de diálogo en torno a los retos emergentes, intercambiar experiencias, difundir avances recientes y acercar a sectores que con frecuencia trabajan de manera aislada. En esta edición hemos querido insistir en la necesidad de abordar problemas complejos desde enfoques interdisciplinarios, de promover prácticas de desarrollo sostenible y de llevar el conocimiento hacia la industria. Todo ello con una meta sencilla de enunciar y difícil de lograr: consolidar una comunidad internacional más fuerte, abierta e innovadora en torno a la Ingeniería de Software.

CONISOFT'26 recibió seis artículos para publicación en español: (1) *“GitLab-EffortMiner: Un framework para la extracción de datos de entornos DevOps para la estimación de esfuerzo”*. Calcular cuánto esfuerzo pedirá un proyecto ágil suele depender de la experiencia del equipo y de los pocos datos históricos disponibles. Para atender ese problema, los autores construyen GitLab-EffortMiner, un *framework* que extrae información de repositorios GitLab y la transforma según los factores del método *Generalized Estimation Method*. A partir de datos de 33 proyectos de código abierto, entrenaron modelos de regresión y encontraron que la información obtenida ayuda a estimar los puntos de historia ajustados, en especial cuando se usan modelos basados en árboles. (2) *“Sistema para la gestión y automatización de pruebas de software mediante integración continua en una universidad pública”*. En muchas organizaciones las pruebas aún se realizan de forma manual, lo que complica el seguimiento y aumenta los errores. Este artículo describe una solución desarrollada con *Extreme Programming* en el área de desarrollo de sistemas de una universidad pública, formada por una aplicación web que administra solicitudes, proyectos y resultados de pruebas, y por una plataforma de integración continua que ejecuta de manera automática pruebas unitarias, funcionales y de rendimiento, además de un análisis estático del código. La evaluación con los modelos TAM y SUS, muestra mejoras en la eficiencia del proceso y en la calidad del software. (3) *“Framework B-DataCom: métricas para la comprensibilidad de los datos en entornos de analítica de negocio”*. En las organizaciones que desarrollan software, los datos de negocio sostienen decisiones que van más allá de la construcción del producto, sin que existiera una forma clara de medir qué tan comprensibles son esos datos. Los autores hacen primero un mapeo sistemático de la literatura y extienden la Ontología de la Medición para, después, proponer *B-DataCom*, un marco que combina un modelo conceptual con once métricas basadas en *Goal-Question-Metric*. Las métricas se instanciaron sobre el proceso MPS de COMPETISOFT y el ejercicio sugiere que pueden escalarse en futuros estudios de caso. (4) *“Propuesta de una arquitectura limpia de un sistema basado en crowdsourcing para la búsqueda de personas desaparecidas en México”*. Un sistema de crowdsourcing

para buscar personas desaparecidas debe cuidar al mismo tiempo la mantenibilidad, el rendimiento y la seguridad, sobre todo si los reportes ciudadanos requieren validación humana. Antes de implementarlo, los autores comparan tres alternativas arquitectónicas (monolito modular, microservicios y *Clean Architecture*) con el método ATAM y un árbol de utilidad construido con los atributos de calidad que priorizaron los interesados. Los resultados apuntan a *Clean Architecture* como la opción con mejor aislamiento del dominio y más fácil de probar, aunque con mayor complejidad inicial. El trabajo aporta, además, una metodología de evaluación arquitectónica pensada para sistemas de impacto social con equipos reducidos y validación humana de los reportes. (5) “*Hacia un sistema recomendador de elementos de gamificación basado en el diseño centrado en el usuario*”. La gamificación aplicada de forma genérica rara vez resulta efectiva, ya que cada persona responde de manera distinta a los elementos de juego. Este trabajo propone una prueba de concepto de un sistema recomendador que combina el filtrado basado en reglas con aprendizaje por refuerzo, planteado como un problema de bandido multibrazo contextual. Cada usuario se representa con un perfil híbrido que cambia según su retroalimentación y, en las simulaciones, el sistema converge a partir de la tercera iteración y produce recomendaciones muy personalizadas. (6) “*Análisis comparativo entre reglas de prioridad en la programación de proyectos y técnicas de priorización en contextos ágiles*”. Se trata de una revisión sistemática de la literatura, basada en el protocolo PRISMA 2020, que compara las reglas de prioridad de la programación de proyectos con las técnicas de priorización en contextos ágiles. A partir del análisis de 72 estudios publicados entre 2018 y 2026, seleccionados de 807 registros de Scopus y Web of Science, los autores muestran que ningún estudio compara ambas aproximaciones de manera directa y que solo el 31.9 % integra técnicas formales de gestión de proyectos en contextos ágiles. Se identifican similitudes funcionales, como WSJF con SPT y MoSCoW con EDD, así como diferencias conceptuales: las técnicas ágiles deciden qué construir según el valor percibido por el cliente, mientras que las reglas PM ordenan las actividades bajo restricciones de recursos. El trabajo aporta, además, un marco preliminar para traducir métricas entre ambos enfoques y recomendaciones de formación para los *Product Owners*.

Agradecemos a todas las personas que hicieron posible este evento: conferencistas, talleristas, revisores y quienes integraron el comité organizador. Un reconocimiento especial a la Universidad Autónoma de Zacatecas por las facilidades que brindó para recibir la conferencia. De igual manera, agradecemos al comité editorial de la Revista de Investigación en Tecnologías de la Información y, en particular, a la Dra. Carolina Tripp Barba, por la gestión realizada.

Esperamos que esta lectura le resulte útil y que despierte nuevas preguntas e ideas para seguir trabajando en la Ingeniería de Software.