

DEPÓSITO LEGAL ZU2020000153

ISSN 0041-8811

E-ISSN 2665-0428

Revista de la Universidad del Zulia

Fundada en 1947
por el Dr. Jesús Enrique Lossada



Ciencias
Exactas,
Naturales
y de la Salud

79
ANIVERSARIO

Año 17 N° 49
Mayo - Agosto 2026
Tercera Época
Maracaibo-Venezuela

Impacto académico del uso de inteligencia artificial mediante equipos ágiles de desarrollo de software

Wilian Richart Delgado Muentes *

José Jacinto Reyes Cárdenas **

Walter Daniel Párraga Andrade ***

RESUMEN

El objetivo general de la presente investigación fue de determinar el impacto de la inteligencia artificial generativa en la productividad y calidad del código en equipos ágiles de desarrollo de software en la especialidad de ingeniería de software de la ULEAM, núcleo Manta, Ecuador. La metodología adoptada para el desarrollo de la investigación que se presenta estuvo enmarcada en el paradigma cuantitativo, con un diseño de campo, se seleccionó una muestra de cuatrocientos cincuenta y seis (456) estudiantes cursantes del tercer semestre de la especialidad de ingeniería de software, el instrumento empleado para el diagnóstico estuvo diseñado con la escala de likert, se aplicó vía correo electrónico a los estudiantes que manifestaron voluntariamente participar en la investigación. Los resultados obtenidos permitieron constatar que los sujetos encuestados prefieren el uso de herramientas de desarrollo con metodologías ágiles y la inteligencia artificial generativa, esto por la versatilidad y precisión que proporciona el desarrollo de códigos de programación en los cuales se minimizan los errores humanos, los cuales de otra forma son posibles. Se concluye que la mayoría de los estudiantes son conscientes y están debidamente orientados desde fuentes verificadas de cuáles son los beneficios y potencialidad que sugiere el uso de aplicaciones especializadas derivadas de IA generativa y las metodologías ágiles.

PALABRAS CLAVE: Metodologías ágiles, Educación, Ingeniería de software, Desarrollo de aplicaciones, Inteligencia artificial generativa.

* Docente. Universidad Laica Eloy Alfaro de Manabí. Manta, Ecuador. ORCID: <https://orcid.org/0000-0002-5136-0677>. E-mail: wilian.delgado@uleam.edu.ec

** Docente. Universidad Laica Eloy Alfaro de Manabí, Ecuador. ORCID: <https://orcid.org/0000-0001-5838-2011>. E-mail: jacinto.reyes@uleam.edu.ec

*** Docente. Universidad Laica Eloy Alfaro de Manabí, Ecuador. ORCID: <https://orcid.org/0000-0003-4719-5872>. E-mail: walterd.parraga@uleam.edu.ec

Recibido: 11/03/2026

Aceptado: 29/04/2026

Academic Impact of the Use of Artificial Intelligence Through Agile Software Development Teams

ABSTRACT

The general objective of this research was to determine the impact of generative artificial intelligence on the productivity and quality of the code in agile software development teams in the software engineering specialty of ULEAM, Manta core, Ecuador. The methodology adopted for the development of the research presented was framed in the quantitative paradigm, with a field design, a sample of four hundred and fifty-six (456) students in the third semester of the software engineering specialty was selected, the instrument used for the diagnosis was designed with the likert scale, it was applied via email to the students who voluntarily expressed participation in the research. The results obtained allowed us to confirm that the subjects surveyed prefer the use of development tools with agile methodologies and generative artificial intelligence, this due to the versatility and precision provided by the development of programming codes in which human errors, which are otherwise possible, are minimized. It is concluded that the majority of students are aware and are properly guided from verified sources about the benefits and potential suggested by the use of specialized applications derived from generative AI and agile methodologies.

KEYWORDS: Agile methodologies, Education, Software engineering, Application development, Generative artificial intelligence

Introducción

La necesidad presente de incrementar la productividad y la eficiencia en los desarrollos que involucran las prácticas empresariales, ha originado que los programadores de software empleen elementos concernientes a la Ingeniería Artificial (IA) generativa y los estándares que están presentes en las gestiones de construcción ágil.

Según Openexpo (2023), la IA generativa está cambiando el juego en cada una de estas fases. En la fase de planificación, puede sugerir arquitecturas óptimas y gestionar la estimación de tiempos. Durante el diseño, herramientas como GitHub Copilot ofrecen soluciones automáticas a problemas complejos, facilitando el trabajo de los desarrolladores. Además, en la implementación, la IA ayuda a generar código automáticamente, lo que reduce errores y acelera el proceso. Por último, en las pruebas, se están integrando sistemas de IA que permiten una cobertura más amplia y detallada en menos tiempo. Todo esto

afecta directamente los tiempos de entrega, la calidad del producto final y los recursos necesarios para llevar a cabo un proyecto de software.

De manera que, para la construcción de un producto final de software es necesario según refiere Armijos et al. (2024) un conjunto de técnicas, prácticas y herramientas empleadas por equipos dedicados al desarrollo de software que permiten llevar a cabo un proyecto con altas posibilidades de éxito. Su propósito radica en planificar, diseñar, construir, probar y entregar software de excelencia de manera eficaz y eficiente. Estas metodologías proporcionan una estructura para el desarrollo del software, abarcando desde la captura de requisitos hasta la fase de mantenimiento. Además, asignan funciones y responsabilidades a los integrantes del equipo, establecen procesos para la gestión del proyecto, la comunicación y la supervisión del avance del mismo.

En este sentido, es importante señalar que como refiere Molina (2018) existen metodologías utilizadas para el desarrollo de software, las cuales se dividen en dos categorías: las tradicionales y las ágiles. Las primeras se basan en las prácticas establecidas en la ingeniería de software, siguiendo un marco disciplinado y un proceso riguroso de implementación. Según Fowler (2005), las metodologías tradicionales buscan imponer disciplina al proceso de desarrollo de software y de esa forma volverlo predecible y eficiente. Para conseguirlo se soportan en un proceso detallado con énfasis en planeación.

De acuerdo con Maida et al. (2015) durante el periodo de los años 90 surgieron nuevas metodologías llamadas “ágiles”, caracterizadas por su enfoque adaptable y sencillo, facilitando entregas frecuentes y rápidas en estrecha colaboración entre desarrolladores y clientes. Es así como, las metodologías ágiles surgen como respuesta a los desafíos que demandan una adaptación rápida en entornos flexibles y sujetos a cambios constantes, ignorando la documentación rigurosa y los procedimientos formales. Es por eso que, en este caso nos centramos en el estudio de las metodologías ágiles con mayor implementación, las cuales son: Extreme Programming (XP), Scrum y Kanban.

Es así como se tiene que, según UNIR (2025), las metodologías ágiles son enfoques de gestión de proyectos que priorizan la colaboración y la flexibilidad, y se basan en ciclos cortos de trabajo para adaptarse a cambios, entregar rápidamente y mejorar de forma continua.

Por tanto, las metodologías ágiles aportan estrategias que proporcionan la mejora mediante la adaptabilidad y colaboración para lograr ese objetivo. Es por esto que, la metodología Agile facilita el desarrollo de proyectos que requieren rapidez y flexibilidad para adaptarse a las necesidades del cliente; siempre orientada a mejorar los resultados. A diferencia de la gestión tradicional de proyectos, las metodologías ágiles no requieren definir completamente el alcance al inicio de los mismos, de hecho, consideramos las metodologías ágiles como una forma innovadora de trabajar y organizarse que "fragmenta" los proyectos en partes que pueden adaptarse sobre la marcha, complementarse y resolverse en poco tiempo. Es decir, no se planifica ni se diseña el proyecto de antemano, sino que a medida que avanza, se va definiendo el proyecto gracias a un feedback constante.

Así mismo, se trabaja en períodos de tiempo durante los cuales cada miembro del equipo debe realizar una serie de tareas. Después de completar estas tareas, se entregan los avances, se reciben comentarios y se reinicia el proceso, permitiendo implementar los cambios necesarios.

De acuerdo con IEBS Biztech School (2026), las metodologías ágiles son aquellas que permiten adaptar la forma de trabajo a las condiciones del proyecto, consiguiendo flexibilidad e inmediatez en la respuesta para amoldar el proyecto y su desarrollo a las circunstancias específicas del entorno.

En esencia, las empresas que apuestan por esta metodología consiguen gestionar sus proyectos de forma flexible, autónoma y eficaz reduciendo los costes e incrementando su productividad.

1. Ventajas del Agile Project Management

Los métodos ágiles tienen muchas ventajas sobre los sistemas tradicionales, porque te permiten empezar un proyecto sin, necesariamente, saber cómo va a acabar, basta con que siempre sepas que quieres hacer en el próximo sprint.

Mejora de la calidad del producto: Estas metodologías fomentan el enfoque proactivo de los miembros del equipo en la búsqueda de la excelencia del producto. Además, la integración, comprobación y mejora continua de las propiedades del producto repercute considerablemente en el resultado final.

Mayor satisfacción del cliente: El cliente está más satisfecho al verse involucrado y comprometido a lo largo de todo el proceso de desarrollo. Mediante varias demostraciones y entregas, el cliente vive a tiempo real las mejoras introducidas en el proceso.

Mayor motivación de los trabajadores: Los equipos de trabajo autogestionados, facilitan el desarrollo de la capacidad creativa y de innovación entre sus miembros.

Trabajo colaborativo: La división del trabajo por distintos equipos y roles junto al desarrollo de reuniones frecuentes, permite una mejor organización del trabajo.

Uso de métricas más relevantes: Las métricas utilizadas para estimar parámetros como tiempo, coste, rendimiento, etc. son normalmente más reales en proyectos ágiles que en los tradicionales. Gracias a la división en pequeños equipos y fases podemos ser más conscientes de lo que está sucediendo.

Mayor control y capacidad de predicción: La oportunidad de revisar y adaptar el producto a lo largo del proceso ágil, permite a todos los miembros del proyecto ejercer un mayor control sobre su trabajo, cosa que permite mejorar la capacidad de predicción en tiempo y costes.

Por lo tanto, es posible constatar lo dicho por Patel (2012), quien afirma que las metodologías ágiles son adaptativas. Este hecho es de gran importancia ya que contrasta con la predictibilidad buscada por las metodologías tradicionales. Con el enfoque de las metodologías ágiles los cambios son eventos esperados que generan valor para el cliente. Para Hoda (2010) las metodologías ágiles funcionan bien dentro de un contexto específico caracterizado por equipos pequeños de desarrollo, ubicados en el mismo sitio, con clientes que pueden tomar decisiones acerca de los requerimientos y su evolución, con requerimientos que cambian con frecuencia (semanal, mensual), con alcance del proyecto o presupuesto variable, con pocas restricciones legales y con pocas restricciones en el proceso de desarrollo

Para Auraquantic (2025), la Inteligencia Artificial generativa, conocida en inglés como Generative AI o GEN AI, abre un vasto horizonte de oportunidades en la creación de contenido novedoso y original. Esta área de la IA tiene la capacidad de producir texto, código de software, imágenes, vídeos, sonidos y diseños de productos y estructuras. Todo esto se logra mediante el uso de algoritmos y modelos de aprendizaje automático, como las Redes Generativas Antagónicas (GANs) o las arquitecturas Transformer, que pueden aprender a partir de un extenso conjunto de datos.

De manera que, la IA generativa emplea modelos extensos formados en lenguajes de programación y en lenguajes naturales para interpretar las peticiones en lenguaje natural (requerimientos) y producir el código adecuado. Es capaz de proponer código, optimizar funciones, desarrollar pruebas, redactar documentación e inclusive transformar bocetos o requisitos de interfaz de usuario en componentes operativos. No obstante, numerosos entornos de montura o estructuración de códigos permiten a los programadores conservar el control: la IA sugiere y los humanos aprueban, modifican o desestiman.

2. IA Generativa y Tecnologías de Gestión de Desarrollo Ágil

Respecto a la combinación de la IA generativa con las tecnologías de gestión de desarrollo ágil, es posible decir que hay modelos verificados que pueden ser empleados para este fin; dentro de las principales herramientas desarrolladas como parte de la gestión ágil de proyectos se encuentran las siguientes, según Neracode (2025):

2.1. Metodología Scrum

Scrum es probablemente la metodología ágil más conocida. Se basa en ciclos cortos llamados sprints que suelen durar entre dos y cuatro semanas. Durante cada sprint, el equipo trabaja en un conjunto específico de tareas, revisa avances y ajusta prioridades. Este enfoque permite: Entregas parciales, pero funcionales en poco tiempo, mejor comunicación entre equipo y cliente, detección temprana de problemas, otorga flexibilidad y cambios. En proyectos de software, Scrum es ideal para productos que requieren mejoras constantes y retroalimentación continua. Asimismo, es perfecto para proyectos que se van definiendo a la par que se van desarrollando y en los que se debe entregar y probar en tiempo récord.

2.2. Metodología Kanban

Kanban es otro tipo de metodología ágil que se centra en la visualización del flujo de trabajo a través de un tablero con columnas que representan cada etapa del proceso. Cada tarea avanza de izquierda a derecha hasta completarse. Sus beneficios incluyen:

1. Mejor claridad en el estado de cada tarea.
2. Reducción de cuellos de botella.
3. Adaptación rápida a cambios de prioridades.

4. Kanban es muy útil cuando trabajamos en proyectos con requerimientos variables o tareas de soporte y mantenimiento. En Neracode la utilizamos para eficientar costos con nuestros equipos de staff y soporte.

2.3. Extreme Programming (XP)

Extreme Programming o XP es una metodología ágil enfocada en mejorar la calidad del software y la capacidad de respuesta ante cambios. Pone énfasis en prácticas como la programación en pareja (pair programming), revisiones de código frecuentes y pruebas automatizadas. Con XP se logra: Mayor calidad del código desde las primeras fases, menos errores en producción y mejora continua del producto.

2.4. Lean Software Development

Inspirada en la filosofía lean de manufactura, este tipo de metodología ágil elimina desperdicios, optimiza recursos y entrega valor lo antes posible. Se centra en priorizar sólo lo que aporta beneficios reales al cliente. En proyectos de software, ayuda a mantener la simplicidad y evitar funcionalidades innecesarias. Utiliza indicadores que permiten medir, mejorar y manejar eficientemente los recursos.

Una vez que estos modelos han sido entrenados, son capaces de generar contenido nuevo a partir de una entrada inicial o simplemente creando muestras aleatorias. Esta habilidad para producir contenido inédito convierte a la IA Generativa en una tecnología poderosa con numerosas aplicaciones prácticas. Sin embargo, también presenta ciertos desafíos, ya que el contenido generado puede tener implicaciones sociales y culturales significativas.

Por lo antes dicho, se plantea el desarrollo de la presente investigación con el objetivo de determinar el impacto de la inteligencia artificial generativa en la productividad y calidad del código en equipos ágiles de desarrollo de software en la especialidad de ingeniería de software de la ULEAM, núcleo Manta, Ecuador.

3. Metodología

La investigación se fundamentó dentro del enfoque cuantitativo, se ha definido dentro del marco de un diseño de campo, que incluyó la aplicación de un instrumento de recolección de datos de tipo cuestionario para un total de cuatrocientos cincuenta y seis

(456) personas, los cuales son estudiantes de la especialidad de Ingeniería de Software de la ULEAM, núcleo Manta, Ecuador.

De manera que, la población estuvo constituida por el total de los estudiantes del tercer semestre de la especialidad referida, siendo la muestra 456 estudiantes de las secciones en las cuales imparten clases de programación de sistemas. Los criterios de inclusión utilizados fueron los de aquellos estudiantes que desearon participar en el trabajo investigativo, incluidos ambos géneros de edades superiores a 18 años hasta los 35 años.

La metodología de recolección de datos fue mediante un cuestionario diseñado con la escala de Likert, debidamente validado el cual fue evaluado con el estadístico Alfa de Cronbach, obteniéndose un valor considerable que permitió la aceptación de los evaluadores y su posterior aplicación de forma online, mediante la aplicación Google Form y enviados por medio del correo electrónico a cada participante.

4. Resultados

Seguidamente se exponen los resultados del proceso de diagnóstico de la investigación:

Opciones de respuesta	Desde	Hasta	Nivel
1 Totalmente en desacuerdo	1,00	1,80	Muy bajo
2 En desacuerdo	1,81	2,60	Bajo
3 Ni de acuerdo ni en desacuerdo	2,61	3,40	Medio
4 De acuerdo	3,41	4,20	Alto
5 Totalmente de acuerdo	4,21	5,00	Muy alto

Ítem 1. ¿El uso de aplicaciones de IA generativa ha potenciado el desarrollo de mejores aplicaciones en el proceso formativo?

Tabla 1. Uso de aplicaciones de IA generativo para potenciar el desarrollo de aplicaciones

Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
1	0	109	171	175

Nota. Fuente: El instrumento de recolección de datos (2026)

Ítem 2. ¿Se logra con la IA generativa minimizar errores de codificación en equipos ágiles de desarrollo de software?

Tabla 2. Minimizan los errores la codificación con equipos agiles de desarrollo de software

Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
0	1	112	162	181

Nota. Fuente: El instrumento de recolección de datos (2026)

Ítem 3. ¿Es posible que la evaluación de los criterios de eficiencia y productividad sean mejorados con la IA generativa?

Tabla 3. Existen criterios de eficiencia y productividad con el uso de IA generativa

Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
1	1	72	178	204

Nota. Fuente: El instrumento de recolección de datos (2026)

Ítem 4. ¿La integración controlada de la IA dentro del proceso ágil ha sido pilar de los desarrollos que ha realizado en el contexto académico?

Tabla 4. El uso de IA ha sido pilar de los desarrollos que ha realizado en el contexto académico

Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
0	0	99	168	189

Nota. Fuente: El instrumento de recolección de datos (2026)

Ítem 5. ¿Es conveniente la gestión automatizada de backlog?

Tabla 5. La gestión automatizada de Backlog es conveniente

Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
0	0	85	175	196

Nota. Fuente: El instrumento de recolección de datos (2026)

Ítem 6. ¿Ha sido posible analizar datos de productividad y comentarios previos para identificar patrones, ofreciendo información valiosa sobre cómo mejorar la colaboración y el rendimiento del equipo con el uso de IA generativa?

Tabla 6. Identificación de mejores patrones de colaboración y rendimientos con el equipo de IA generativa

Totalmente en desacuerdo	En desacuerdo	Ni de acuerdo ni en desacuerdo	De acuerdo	Totalmente de acuerdo
0	1	70	171	214

Nota. Fuente: El instrumento de recolección de datos (2026)

De los resultados obtenidos se resumen los siguientes valores:

Tabla 7. Estadísticos descriptivos de los valores resultantes del instrumento de recolección de datos

N válido	Media	Mediana	D.estandar	Cuartil1 Q1	Cuartil3 Q3	IQR	Min	Max	IC 95% (Inf)	IC 95% (Sup)	Nivel
456	4,22	4,00	0,77	4,00	5,00	1,00	1,00	5,00	4,15	4,29	Muy alto

Nota. Fuente: El instrumento de recolección de datos (2026)

Tabla 8. Cálculo de confiabilidad de los ítems (Alfa de Cronbach) en la aplicación Microsoft Excel

Items	Varianza		Items	6
P1	0,62		Σ Var(ítems)	3,50
P2	0,63		Var(Total)	10,53
P3	0,56		Alfa (α)	0,80
P4	0,59		Criterio	Alta (0.80–0.89)
P5	0,56		Válido (S/N)	SI
P6	0,54			
Tot. varianza	10,53			

Nota. Fuente: El instrumento de recolección de datos (2026)

5. Discusión y conclusiones

De acuerdo con Bautista (2022), el desarrollo de software ha experimentado una creciente demanda y una acelerada evolución tecnológica, lo que ha impulsado la necesidad de metodologías ágiles para garantizar la eficiencia y la adaptabilidad en los proyectos de software. En este sentido, Molina et al. (2019) refiere que las metodologías ágiles se crearon para mejorar el proceso de desarrollo de software y los problemas asociados con el uso de metodologías tradicionales, que presentan rigidez y un extenso proceso de documentación, así como también surgen de ideas de mejores prácticas de la comunidad orientada a objetos. Para Iriondo (2022), estas metodologías destacan por su flexibilidad, adaptabilidad y enfoque centrado en la entrega de valor al cliente de manera iterativa e incremental. A diferencia de los enfoques tradicionales, las metodologías ágiles se fundamentan en principios colaborativos, interacción constante con los clientes y equipos multidisciplinarios autoorganizados.

En su mayoría los resultados de la recolección de datos como parte de la presente investigación permitió obtener los valores sobre la percepción de las metodologías ágiles y el uso de Inteligencia Artificial (IA) generativa, los cuales indican la flexibilidad y adaptabilidad que los desarrolladores participantes consideran otorga a los proyectos. De tal manera que, la mayoría de los encuestados afirma que el uso de aplicaciones de este tipo

ha potenciado la construcción de software, lo que coincide con el trabajo desarrollado por Armijos et al. (2024), donde el 93.33% de las empresas destacaron la mejora en la colaboración y comunicación entre los miembros del equipo como uno de los principales beneficios. Así mismo, el 73.33% mencionó la entrega de software de mayor calidad en ciclos más cortos como otro beneficio significativo. Además, el 53.33% de las empresas señalaron una mayor satisfacción del cliente como un beneficio relevante, mientras que el 26.67% destacó la mayor flexibilidad para adaptarse a cambios en los requisitos del cliente como otro aspecto beneficioso de la adopción de metodologías ágiles. Estos resultados resaltan la percepción positiva de las empresas sobre las ventajas que ofrecen las metodologías ágiles en el desarrollo de software.

En esta investigación se constató en un porcentaje superior al 80% que los participantes concuerdan en la minimización de errores y la reducción de gastos de recursos como parte de la construcción basada en el modelo ágil de desarrollo de software, lo que se acerca a lo dicho por Correa (2025) con respecto al uso de aplicaciones de IA generativa más las tecnologías de gestión de desarrollo ágil, en la mejora de la traducción de código y la migración de lenguajes legacy con PHP y mejor performance, escalabilidad, conocimiento del equipo, otro lenguaje de programación o framework, incluso pasar de una versión antigua de PHP a una actual.

Al respecto, refieren Armijos et al. (2024) que el 93.33% de las empresas destacaron la mejora en la colaboración y comunicación entre los miembros del equipo como uno de los principales beneficios. Así mismo, el 73.33% mencionó la entrega de software de mayor calidad en ciclos más cortos como otro beneficio significativo. Además, el 53.33% de las empresas señalaron una mayor satisfacción del cliente como una ventaja relevante, mientras que el 26.67% destacó la mayor flexibilidad para adaptarse a cambios en los requisitos del cliente como otro aspecto beneficioso de la adopción de metodologías ágiles.

En el presente estudio surgen estadísticas generales positivas sobre el uso de las metodologías ágiles y la inteligencia artificial generativa, de esta forma se puede afirmar que los estudiantes hacen uso de las mismas para el logro con mayor precisión de los objetivos del desarrollo de aplicaciones determinados, es así como, la IA generativa en el ámbito educativo lleva a cabo un examen exhaustivo de una diversidad de documentos en este campo, este estudio brinda una comprensión global del estado actual de la investigación en este ámbito. Al sintetizar y resumir los hallazgos clave, las metodologías y las

recomendaciones de múltiples documentos, este estudio se convierte en un recurso para investigadores, educadores y formuladores de políticas que estén genuinamente interesados en explorar la intersección transformadora de la GAI y la educación.

En definitiva, lo que se persigue es un desarrollo ágil debido a la precisión, como afirmaban los participantes encuestados, donde se consigan los estándares de calidad de desarrollo y la satisfacción del cliente. Según Navarro (2013), entre los principios que derivan de las metodologías ágiles, destacan: la satisfacción del cliente mediante entregas tempranas y continuas de software que funcione, requerimientos cambiantes en cualquier etapa del proyecto, participación activa del cliente, simplicidad, equipos de desarrollo motivados y auto-organizados, comunicación efectiva, auto inspecciones y adaptación.

En lo que respecta a la investigación presentada, se puede argumentar que la mayoría de los estudiantes son conscientes y están documentados de los beneficios y potencialidad que sugiere el uso de aplicaciones especializadas derivadas de IA generativa y las metodologías ágiles.

Referencias

Armijos Ortega, Lady M; Velez Macas, Carlos A; Lojan Cueva, Edison L. (2024). Estudio de la adopción de metodologías ágiles en proyectos de desarrollo de software en la región 7 del Ecuador. *Espacios*, Vol. 45, Núm. 4, agosto 2024, p. 73-84. Doi: <https://doi.org/10.48082/espacios-a24v45n04p06>.

Auraquantic (2025). IA Generativa: Qué es y Tipos De Aplicaciones. <https://www.auraquantic.com/es/blog/ia-generativa/>

Bautista Villegas, E. (2022). Metodologías ágiles XP y Scrum, empleadas para el desarrollo de páginas web, bajo MVC, con lenguaje PHP y framework Laravel. *Revista Amazonía Digital*, Vol. 1, Núm. 1. DOI:[10.55873/rad.v1i1.168](https://doi.org/10.55873/rad.v1i1.168)

Correa, J. (2025). AI for developers: Guía completa de AI para desarrolladores de software [2025]. <https://develpero.io/blog/ai-for-developers>

Fowler, M. (2005). The new methodology. <http://martinfowler.com/articles/newMethodology.html>

Hoda, R., Kruchten, P., Noble, J., & Marshal, S. Agility (2010). in Context. Proceedings of the ACM International Conference on Object Oriented Programming Systems Languages and Applications (OOPSLA '10). New York, ACM.

IEBS Biztech School (2026). Metodologías ágiles: ¿Qué son y cuáles son más utilizadas? <https://www.iebschool.com/hub/que-son-metodologias-agiles-agile-scrum/>

Iriondo Alzola, I. (2022). Uso de metodologías ágiles en grandes organizaciones. Euskal Herriko Unibertsitatea.

Maida, EG, Pacienza, J. (2015). Metodologías de desarrollo de software [en línea]. Tesis de Licenciatura en Sistemas y Computación. Facultad de Química e Ingeniería “Fray Rogelio Bacon”. Universidad Católica Argentina, 2015. Disponible en: <http://bibliotecadigital.uca.edu.ar/repositorio/tesis/metodologias-desarrollo-software.pdf>

Molina Ríos, J., Pedreira-Souto, N. (2019). Enfoque de Metodologías Ágiles en el Desarrollo de Software Basado en Web. *Información*, 10 (10), 314. <https://doi.org/10.3390/info10100314>

Molina Montero, B., Vite Cevallos, H., & Dávila Cuesta, J. (2018). Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software. *Espirales Revista Multidisciplinaria De investigación*, 2(17). <https://doi.org/10.31876/re.v2i17.269>

Navarro Cadavid, Andrés; Fernández Martínez, Juan Daniel; Morales Vélez, J. (2013). Revisión de metodologías ágiles para el desarrollo de software. *Prospectiva*, Vol. 11, Núm. 2, julio-diciembre, 2013, pp. 30-39 Universidad Autónoma del Caribe. doi:10.15665/rp.v11i2.36

Neracode (2025). Metodologías ágiles más usadas en desarrollo de software. <https://neracode.com/blog/metodologias-agiles-mas-usadas-desarrollo-software>

Openexpo (2023). Impacto de la IA generativa en el ciclo de vida del desarrollo de software (SDLC). <https://openexpo.europa.com/es/impacto-de-la-ia-generativa-en-el-ciclo-de-vida-del-desarrollo-de-software-sdlc/>

UNIR (2025). Metodologías ágiles para la gestión de proyectos tecnológicos <https://mexico.unir.net/noticias/ingenieria/metodologias-agiles/>

Conflicto de interés

Los autores de este manuscrito declaran no tener ningún conflicto de interés.

Declaración ética

Los autores declaran que el proceso de investigación que dio lugar al presente manuscrito se desarrolló siguiendo criterios éticos, por lo que fueron empleadas en forma racional y profesional las herramientas tecnológicas asociadas a la generación del conocimiento.

Copyright

La *Revista de la Universidad del Zulia* declara que reconoce los derechos de los autores de los trabajos originales que en ella se publican; dichos trabajos son propiedad intelectual de sus autores. Los autores preservan sus derechos de autoría y comparten sin propósitos comerciales, según la licencia adoptada por la revista

Licencia Creative Commons

Esta obra está bajo una Licencia Creative Commons Atribución-NoComercial-Compartir Igual 4.0 Internacional



REVISTA DE LA UNIVERSIDAD DEL ZULIA, Fundada el 31 de mayo de 1947

UNIVERSIDAD DEL ZULIA, Fundada el 11 de septiembre de 1891