

Guía práctica sobre herramientas de IA que se utilizan en el proceso de desarrollo de software bajo metodologías ágiles.


Practical guide on AI tools used in the software development process under agile methodologies.

Guerrero-Benalcázar, Rubén Ignacio¹; Lascano-Rivera, Samuel².

¹ Universidad Politécnica Estatal del Carchi; Ecuador, Tulcán; <https://orcid.org/0009-0000-6805-2139>; rubenignacio1522@hotmail.com

² Universidad Politécnica Estatal del Carchi; Ecuador, Tulcán; <https://orcid.org/0000-0001-5967-6441>; samuel.lascano@upec.edu.ec

¹ Autor Correspondencia

 <https://doi.org/10.63618/omd/isj/v3/n4/155>

Cita: Guerrero Benalcázar, R. I., & Lascano-Rivera, S. (2025). Guía práctica sobre herramientas de IA que se utilizan en el proceso de desarrollo de software bajo metodologías ágiles. *Innova Science Journal*, 3(4), 523-539. <https://doi.org/10.63618/omd/isj/v3/n4/155>

Recibido: xx/xx/20xx
Aceptado: xx/xx/20xx
Publicado: xx/xx/20xx



Copyright: © 2025 por los autores. Este artículo es un artículo de acceso abierto distribuido bajo los términos y condiciones de la **Licencia Creative Commons, Atribución-NoComercial 4.0 Internacional. (CC BY-NC)**.

(<https://creativecommons.org/licenses/by-nc/4.0/>)

Resumen: El artículo presenta una guía práctica sobre el uso de herramientas de inteligencia artificial (IA) en el desarrollo de software bajo metodologías ágiles. A través de una revisión sistemática de literatura basada en el protocolo PRISMA, se analizaron 30 artículos publicados entre 2019 y 2024, provenientes de contextos académicos e industriales. Los resultados evidencian que la IA se integra con mayor frecuencia en metodologías como Scrum y XP, optimizando tareas de planificación, codificación, pruebas y documentación. Herramientas como GitHub Copilot, Azure AI, Test.AI y motores NLP destacan por mejorar la productividad, la calidad del software y la reducción del tiempo de entrega. La guía clasifica las herramientas según las fases del ciclo ágil, ofreciendo prompts y recomendaciones prácticas para su aplicación. Se concluye que la IA se ha consolidado como un componente estratégico en la ingeniería de software, aunque su implementación requiere capacitación continua y adaptación al contexto organizacional. La investigación propone validar empíricamente esta guía en entornos reales para medir su impacto en la productividad y la madurez ágil.

Palabras clave: inteligencia artificial; metodologías ágiles; desarrollo de software; automatización; productividad; Scrum.

Abstract: The article presents a practical guide on the use of artificial intelligence (AI) tools in software development under agile methodologies. Through a systematic literature review based on the PRISMA protocol, 30 articles published between 2019 and 2024, from academic and industrial contexts, were analyzed. The results show that AI is most frequently integrated into methodologies such as Scrum and XP, optimizing planning, coding, testing, and documentation tasks. Tools such as GitHub Copilot, Azure AI, Test.AI, and NLP engines stand out for improving productivity, software quality, and reducing delivery time. The guide classifies the tools according to the phases of the agile cycle, offering prompts and practical recommendations for their application. It concludes that AI has established itself as a strategic component in software engineering, although its implementation requires continuous training and adaptation to the organizational context. The research proposes to empirically validate this guide in real environments to measure its impact on productivity and agile maturity.

Keywords: artificial intelligence; agile methodologies; software development; automation; productivity; Scrum.

1. Introducción

La evolución del desarrollo de software se caracteriza actualmente por la adopción generalizada de metodologías ágiles como respuesta a entornos dinámicos y a la necesidad de productos más adaptables y funcionales. Metodologías como Scrum, Kanban y Extreme Programming (XP) se consolidan como enfoques clave para mejorar los ciclos de entrega, fomentar la colaboración con el cliente y fortalecer la capacidad de adaptación frente a los cambios (Martínez et al., 2022; Gaete et al., 2021).

De forma paralela, la inteligencia artificial (IA) se posiciona como una tecnología transformadora en múltiples dominios, incluyendo el desarrollo de software. En los últimos años, su integración en el ciclo de vida del desarrollo gana fuerza (Salazar-Cedeño et al., 2024; Monsalves et al., 2024). En contextos ágiles, la IA permite automatizar tareas repetitivas, optimizar la planificación, mejorar la calidad de los entregables y asistir en la toma de decisiones en tiempo real (Kalava, 2024; Seppälä, 2024; Kumar, 2024).

Diversos estudios evidencian beneficios concretos derivados de esta integración. Seppälä (2024) reporta un aumento en la eficiencia de codificación mediante el uso de GitHub Copilot; Monsalves et al. (2024) destacan la utilidad de servicios como Azure AI y Google Cloud AI (AlaaS) en flujos de trabajo ágiles como Scrum y Kanban; mientras que Upadhyaya (2022) y Salazar-Cedeño et al. (2024) resaltan la automatización de pruebas y la predicción de fallos como elementos claves en la mejora de la calidad del software.

A pesar de estos avances, persiste una brecha tanto teórica como práctica en cuanto a la sistematización del uso de herramientas de IA específicas en contextos ágiles. Aunque existen algunas revisiones centradas en el impacto general de la IA (Rendón De La Torre, 2024; Tupsakhare, 2022), todavía no se consolida una guía que articule estas tecnologías con las distintas fases del desarrollo ágil ni que sintetice sus beneficios reales según la metodología aplicada.

Desde un enfoque teórico, este estudio se apoya en la teoría “Re-Evaluating Media Richness in Software Development Settings” (Mohammed et al., 2014), complementada por enfoques contemporáneos de ingeniería de software asistida por IA, aprendizaje automático, procesamiento de lenguaje natural (NLP) y plataformas cognitivas (Kumar, 2024; Monsalves et al., 2024). Conceptos como automatización inteligente, testing autónomo, estimaciones predictivas y colaboración asistida por IA constituyen pilares de esta línea de innovación (Guerrero et al., 2023; GitLab, 2023).

En el contexto latinoamericano, se observa un crecimiento sostenido en la adopción de metodologías ágiles dentro de industrias tecnológicas y de servicios (Armijos et al., 2024). La presión por entregar software de calidad en menor tiempo impulsa a muchas organizaciones a incorporar IA como soporte en fases críticas como la planificación, el desarrollo y el testing (Cabrerero-Daniel, 2023).

En este marco, la presente investigación establece tres líneas de acción complementarias: primero, analizar el impacto de la inteligencia artificial en las distintas fases del desarrollo ágil; segundo, identificar las metodologías ágiles y herramientas de IA con mayor aplicabilidad en contextos reales; y tercero, proponer una guía

estructurada que oriente su integración efectiva, sustentada en evidencia científica e industrial reciente.

Esta investigación se fundamenta en una revisión sistemática de literatura que reúne aportes científicos y experiencias prácticas, con el objetivo de identificar patrones de uso, beneficios alcanzados y tecnologías emergentes que están transformando el desarrollo de software. A partir de este análisis, se diseña una propuesta práctica que busca ofrecer soporte estratégico, fundamentado y escalable para equipos de desarrollo, gestores de proyectos e investigadores interesados en adoptar soluciones de IA en entornos ágiles.

2. Materiales y Métodos

Este estudio se enmarcó en un enfoque mixto, de naturaleza descriptiva y analítica, orientado a comprender y sistematizar el uso de herramientas de inteligencia artificial (IA) dentro de proyectos de desarrollo de software gestionados mediante metodologías ágiles. Se adoptó como método principal una revisión sistemática de literatura (RSL), guiada por los lineamientos PRISMA 2020 (Page et al., 2021), complementada con análisis cuantitativos de tipo descriptivo enfocados en la frecuencia y distribución de herramientas de IA en cada fase del ciclo ágil. Este enfoque permitió identificar patrones de uso, tendencias tecnológicas dominantes y niveles de adopción según el tipo de metodología, aportando una visión más estructurada del panorama actual.

2.1. Diseño del estudio

El proceso metodológico de revisión se desarrolló en cuatro etapas. En primer lugar, se planificó la revisión mediante la definición de un protocolo de búsqueda, la selección de palabras clave, el uso de operadores booleanos y el establecimiento de criterios de inclusión y exclusión. Posteriormente, se llevó a cabo la búsqueda de literatura en bases de datos académicas reconocidas como IEEE Xplore, Scopus, SpringerLink, Redalyc y Google Scholar, complementada con documentos técnicos proporcionados por el autor. En la tercera fase, se aplicaron filtros temáticos y lingüísticos que permitieron seleccionar los artículos más relevantes para el estudio. Finalmente, se procedió a la extracción y análisis de los datos, organizando la información en matrices de codificación y realizando análisis estadísticos descriptivos para identificar patrones y tendencias significativas.

2.2. Estrategia de búsqueda

La búsqueda se ejecutó entre enero y abril de 2025 utilizando combinaciones de palabras clave:

("artificial intelligence" OR "IA") AND ("agile methodologies" OR "Scrum" OR "XP" OR "Kanban" OR "agile software development") AND ("tools" OR "applications" OR "benefits")

Se utilizó el operador booleano AND para refinar resultados.

2.2.1. Criterios de inclusión y exclusión de textos o artículos

Los criterios de inclusión y exclusión empleados para seleccionar los artículos analizados se detallan en la Tabla 1, los cuales fueron definidos con base en la

relevancia temática, el acceso al texto completo y la pertinencia metodológica dentro del enfoque del estudio.

Tabla 1

Criterios de inclusión y exclusión

Tipo de criterio	Aplicación
Inclusión	Publicaciones entre 2019-2024; estudios en inglés o español; estudios que relacionen IA con metodologías ágiles en desarrollo de software; documentos revisados por pares o tesis académicas.
Exclusión	Artículos duplicados, sin acceso completo, o no relacionados directamente con software ágil e inteligencia artificial.

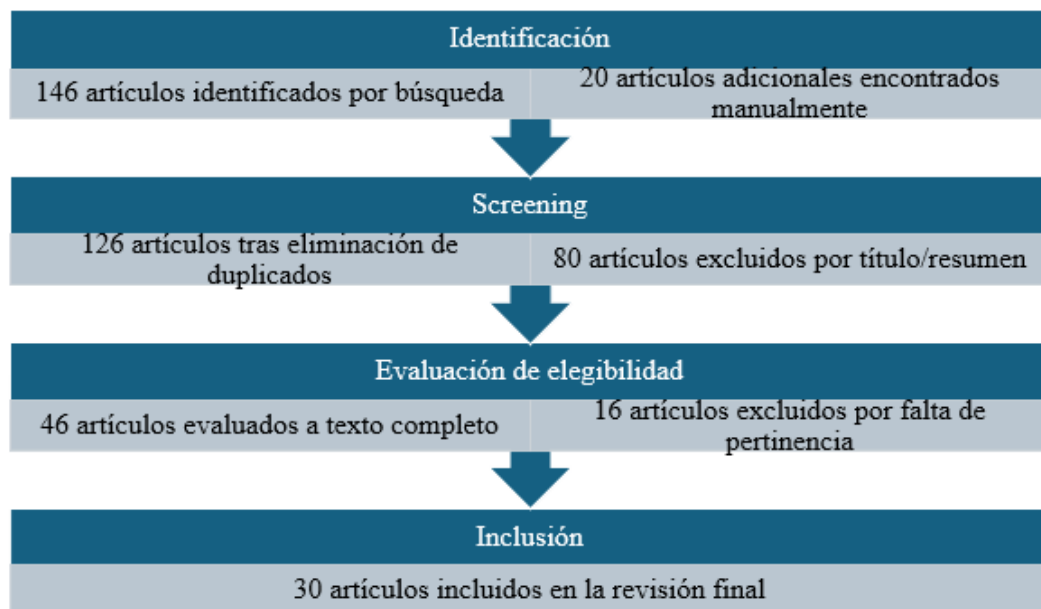
Nota. Elaboración Propia

2.2.2. Proceso de selección de artículos

La selección de los estudios incluidos se realizó siguiendo el esquema de flujo establecido en la guía PRISMA 2020, adaptado a los criterios de esta investigación. La Figura 1 muestra las etapas del proceso, desde la identificación inicial de artículos hasta la inclusión final de los estudios analizados, detallando los criterios de exclusión aplicados en cada fase.

Figura 1

Diagrama de flujo de selección de artículos (PRISMA adaptado)



Nota. Elaboración Propia

2.2.3. Técnicas de recolección y análisis de datos

La recolección de datos se llevó a cabo mediante matrices en Microsoft Excel ta como se muestra en la siguiente Figura 2, donde se codificaron variables clave de cada artículo seleccionado. Entre los datos sistematizados se incluyeron:

- Autor(es)
- Año de publicación
- Contexto del estudio (industrial o académico)
- Tipo de herramienta de IA utilizada
- Metodología ágil aplicada
- Principales hallazgos o contribuciones

Figura 2

Matriz de artículos seleccionados

	A	B	C	D	E	F	G
1	N°	Autor(es)	Año	Contexto	Herramienta IA	Metodología Ágil	Beneficio principal
2	1	Kalava, S.	2024	Académico	GitHub Copilot	Scrum	Productividad
3	2	Seppälä, E.	2024	Industrial	GitHub Copilot	Kanban	Automatización codificación
4	3	Monsalves et al.	2024	Industrial	AlaaS (Azure, AWS)	Scrum, Kanban	Agilidad en entregas
5	4	Upadhyaya, N.	2022	Académico	NLP + LLM	Scrum, XP	Aceleración de ciclos
6	5	Armijos et al.	2024	Industrial	Sin especificar	Scrum	Éxito ágil
7	6	Rendón De La Torre, A.	2024	Académico	Modelos ML	Scrum, Kanban	Integración IA-ágil
8	7	Salazar-Cedeño et al.	2024	Académico	Análisis IA	Scrum	Reducción de defectos
9	8	Tupsakhare, P.	2022	Industrial	ML analytics	Scrum	Eficiencia analítica
10	9	Gaete et al.	2021	Académico	Frameworks Ágiles	Scrum, Kanban, Lean	Flujo optimizado
11	10	Martínez et al.	2022	Académico	Comparativo Ágil	Scrum, Waterfall	Beneficio de agilidad
12	11	Guerrero et al.	2023	Académico	IA en calidad de software	Scrum, XP	Calidad de software
13	12	Kumar, S.	2024	Académico	ML y RL	Scrum, XP	Asistentes IA
14	13	Hanssen et al.	2023	Industrial	Testing asistido	Scrum	Validación más rápida
15	14	Jovanović et al.	2023	Industrial	LLM integración	Scrum	Optimización pipelines
16	15	Alsaqaf et al.	2022	Académico	Herramientas métricas IA	Scrum	Evaluación de madurez
17	16	Lee & Hwang	2022	Industrial	Modelos predictivos	Scrum	Planificación precisa
18	17	Yassine et al.	2023	Industrial	NLP Engines	XP, Scrum	Automatización de requisitos
19	18	Suto et al.	2022	Académico	Modelos regresivos	Scrum	Estimaciones optimizadas
20	19	Limam et al.	2023	Industrial	Dashboards IA	Scrum	Gestión basada en datos
21	20	Choudhury et al.	2023	Académico	ML Testing	XP, Scrum	Testing automatizado
22	21	Arora et al.	2022	Industrial	Retrospectivas IA	Scrum	Feedback inteligente
23	22	Ayalew et al.	2022	Industrial	Sprint planning IA	Scrum	Reducción time-to-market
24	23	Vieira et al.	2023	Industrial	ML Big Data	ScrumBan	Adaptabilidad Big Data
25	24	Najjar et al.	2022	Industrial	Modelos de riesgo IA	Scrum, XP	Mitigación de riesgos
26	25	Wang et al.	2022	Académico	Deep Learning	XP, Scrum	Reducción de fallas
27	26	De França et al.	2022	Industrial	MLOps pipelines	Scrum	Automatización despliegues
28	27	Birje et al.	2023	Académico	Integración IA	Scrum, Kanban	Monitoreo inteligente
29	28	Oliveira et al.	2023	Industrial	Agile+AI frameworks	Lean-Agile	Aceleración de entregas
30	29	Dey et al.	2022	Industrial	IA en Testing	Scrum, XP	Mejora cobertura testing
31	30	Cabrero-Daniel, B.	2023	Académico	IA predictiva ágil	Scrum, XP	Optimización de ciclos ágiles

Nota. Elaboración Propia

Para el análisis de la información recopilada se emplearon enfoques mixtos de tipo descriptivo y cualitativo. En el plano cuantitativo, se aplicaron técnicas estadísticas descriptivas como el cálculo de frecuencias absolutas y porcentajes, con el objetivo de identificar patrones de uso de herramientas de IA y su relación con las metodologías ágiles reportadas en los estudios seleccionados. Estos resultados se visualizaron mediante representaciones gráficas, diagramas circulares, de barras y de columnas, lo que permitió facilitar la interpretación comparativa de los hallazgos.

Paralelamente, se realizó un análisis cualitativo de tipo inductivo, basado en la codificación abierta de los hallazgos reportados en cada estudio. Este análisis permitió la identificación de categorías emergentes como productividad, calidad del software, satisfacción del equipo, automatización de procesos y reducción del time-to-market. La triangulación entre ambas aproximaciones permitió una comprensión más profunda de las tendencias y beneficios asociados al uso de IA en entornos ágiles.

3. Resultados

3.1. Análisis descriptivo de los artículos seleccionados

Tabla 2

Resumen de artículos seleccionados

N°	Autor(es)	Año	Contexto	Herramienta IA	Metodología Ágil	Beneficio principal
1	Kalava, S.	2024	Académico	GitHub Copilot	Scrum	Productividad
2	Seppälä, E.	2024	Industrial	GitHub Copilot	Kanban	Automatización codificación
3	Monsalves et al.	2024	Industrial	AIaaS (Azure, AWS)	Scrum, Kanban	Agilidad en entregas
4	Upadhyaya, N.	2022	Académico	NLP + LLM	Scrum, XP	Aceleración de ciclos
5	Armijos et al.	2024	Industrial	Sin especificar	Scrum	Éxito ágil
6	Rendón De La Torre, A.	2024	Académico	Modelos ML	Scrum, Kanban	Integración IA-ágil
7	Salazar-Cedeño et al.	2024	Académico	Análisis IA	Scrum	Reducción de defectos
8	Tupsakhare, P.	2022	Industrial	ML analytics	Scrum	Eficiencia analítica
9	Gaete et al.	2021	Académico	Frameworks Ágiles	Scrum, Kanban, Lean	Flujo optimizado
10	Martínez et al.	2022	Académico	Comparativo Ágil	Scrum, Waterfall	Beneficio de agilidad
11	Guerrero et al.	2023	Académico	IA en calidad de software	Scrum, XP	Calidad de software
12	Kumar, S.	2024	Académico	ML y RL	Scrum, XP	Asistentes IA
13	Hanssen et al.	2023	Industrial	Testing asistido	Scrum	Validación más rápida
14	Jovanović et al.	2023	Industrial	LLM integración	Scrum	Optimización pipelines
15	Alsaqaf et al.	2022	Académico	Herramientas métricas IA	Scrum	Evaluación de madurez
16	Lee & Hwang	2022	Industrial	Modelos predictivos	Scrum	Planificación precisa
17	Yassine et al.	2023	Industrial	NLP Engines	XP, Scrum	Automatización de requisitos
18	Suto et al.	2022	Académico	Modelos regresivos	Scrum	Estimaciones optimizadas
19	Limam et al.	2023	Industrial	Dashboards IA	Scrum	Gestión basada en datos
20	Choudhury et al.	2023	Académico	ML Testing	XP, Scrum	Testing automatizado
21	Arora et al.	2022	Industrial	Retrospectivas IA	Scrum	Feedback inteligente
22	Ayalew et al.	2022	Industrial	Sprint planning IA	Scrum	Reducción time-to-market

23	Vieira et al.	2023	Industrial	ML Big Data	ScrumBan	Adaptabilidad Big Data
24	Najjar et al.	2022	Industrial	Modelos de riesgo IA	Scrum, XP	Mitigación de riesgos
25	Wang et al.	2022	Académico	Deep Learning	XP, Scrum	Reducción de fallas
26	De França et al.	2022	Industrial	MLOps pipelines	Scrum	Automatización despliegues
27	Birje et al.	2023	Académico	Integración IA	Scrum, Kanban	Monitoreo inteligente
28	Oliveira et al.	2023	Industrial	Agile+AI frameworks	Lean-Agile	Aceleración de entregas
29	Dey et al.	2022	Industrial	IA en Testing	Scrum, XP	Mejora cobertura testing
30	Cabrero-Daniel, B.	2023	Académico	IA predictiva ágil	Scrum, XP	Optimización de ciclos ágiles

Nota. Elaboración Propia

3.2. Consideraciones éticas

El presente estudio no involucró interacción directa con participantes humanos, ni recopilación de datos personales. Todas las fuentes documentales analizadas fueron de acceso público o pertenecían a bases de datos académicas licenciadas. Se respetaron las normas éticas de citación y atribución de ideas conforme a los principios de integridad académica.

3.3. Limitaciones del estudio

Entre las principales limitaciones del estudio se identifican varios aspectos relevantes. En primer lugar, la revisión se restringió a publicaciones en inglés y español, lo que pudo haber dejado fuera investigaciones significativas publicadas en otros idiomas. Asimismo, la muestra analizada incluyó estudios de caso con contextos específicos que, si bien ofrecen un análisis profundo, limitan la generalización de los resultados a otros entornos o sectores industriales. Además, no se realizó una validación empírica directa de los hallazgos en escenarios reales de desarrollo de software, especialmente en América Latina, donde las condiciones tecnológicas y operativas presentan particularidades que podrían influir en la aplicabilidad de las conclusiones. Finalmente, se reconoce que el rápido avance de las tecnologías de inteligencia artificial podría hacer que algunas herramientas o enfoques identificados queden obsoletos en el corto plazo, por lo que se recomienda que futuras investigaciones realicen revisiones sistemáticas periódicas que integren los avances más recientes del campo.

Como resultado de la revisión sistemática, se seleccionaron 30 artículos científicos de 116 relacionados con la aplicación de herramientas de inteligencia artificial (IA) en proyectos de desarrollo de software gestionados mediante metodologías ágiles. De los 30 artículos:

- El 60% de los estudios corresponde a contextos industriales, es decir, casos reales de implementación en empresas tecnológicas.
- El 40% restante corresponde a investigaciones académicas desarrolladas en universidades o centros de investigación.

El análisis evidenció una diversidad de aplicaciones prácticas de herramientas de IA integradas en distintos marcos metodológicos ágiles, asociadas consistentemente con mejoras en la eficiencia del desarrollo y en la calidad de los productos software.

3.4. Herramientas de inteligencia artificial más utilizadas

Las herramientas de IA se clasificaron en seis grandes categorías de uso. La distribución porcentual de estas herramientas entre los 30 estudios analizados es la siguiente:

Como se observa en la Tabla 3, los generadores de código fueron las herramientas más mencionadas (30%), destacando GitHub Copilot y Amazon CodeWhisperer como asistentes para codificación asistida en tiempo real. Las plataformas de IA como servicio (AlaaS) también mostraron amplio uso, especialmente en contextos de integración continua y análisis predictivo.

Tabla 3

Herramientas IA aplicadas en desarrollo ágil

Tipo de herramienta IA	Porcentaje (%)
Generadores de código (Copilot, Tabnine)	30%
Plataformas AlaaS (Azure, AWS, GCP)	20%
Automatización de testing	17%
Motores NLP para análisis de requisitos	13%
Modelos predictivos y ML	10%
Dashboards IA y asistentes retros	10%

Nota. Scrum destaca como la metodología ágil más ampliamente adoptada en los estudios revisados (presente en el 80% de los casos, como se observa en la Tabla 4), debido a su estructura iterativa, basada en ciclos cortos de desarrollo (sprints), y a su enfoque incremental, que facilita la integración progresiva de herramientas inteligentes.

Esta metodología permite una planificación adaptativa y una retroalimentación continua, lo cual resulta compatible con las funcionalidades de la inteligencia artificial, como la predicción automatizada de tareas, el testing incremental y el análisis en tiempo real.

Tabla 4

Metodologías ágiles integradas con IA

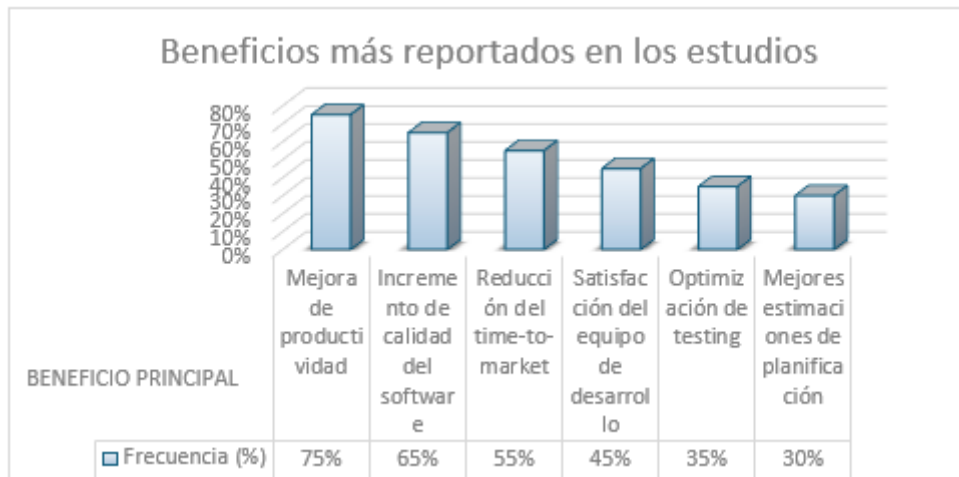
Metodología Ágil	Presencia (%)
Scrum	80%
Extreme Programming (XP)	40%
Kanban	30%
Lean / Lean-Agile / ScrumBan	20%

Nota. En comparación con otras metodologías como Kanban o XP, Scrum ofrece una mayor formalización de roles y artefactos, lo que ha favorecido su adopción como marco base para la incorporación de soluciones IA en entornos industriales y académicos.

En el análisis de los 30 estudios revisados, se identificaron una serie de beneficios recurrentes atribuidos a la integración de herramientas de inteligencia artificial en proyectos gestionados con metodologías ágiles. La Figura 3 sintetiza los principales beneficios reportados y su frecuencia de aparición entre los artículos analizados.

Figura 3

Beneficios más reportados en los estudios



Nota. Elaboración Propia

Como se observa en la Figura 3, el beneficio más reportado fue la mejora de la productividad (75%), seguido del incremento en la calidad del software (65%) y la reducción del time-to-market (55%). Estos resultados reflejan una tendencia clara hacia el uso de IA como acelerador de procesos críticos dentro del ciclo ágil, especialmente en las fases de codificación, testing y planificación.

3.5. Guía práctica de herramientas de inteligencia artificial en entornos ágiles

La implementación de inteligencia artificial (IA) en procesos de desarrollo ágil de software no solo representa una tendencia tecnológica, sino también una oportunidad concreta para repensar la eficiencia, la calidad y la adaptabilidad en equipos de ingeniería. Sin embargo, a pesar del interés creciente, muchos equipos aún carecen de referencias claras sobre qué herramientas utilizar, en qué fases del ciclo ágil aplicarlas y qué beneficios esperar.

En respuesta a esta necesidad, se ha elaborado la siguiente guía práctica, producto del análisis sistemático de 30 estudios académicos e industriales revisados en esta investigación. El objetivo es brindar un mapa funcional de las herramientas IA más relevantes, asociadas a fases específicas del desarrollo ágil y vinculadas a los marcos metodológicos en los que han demostrado mayor efectividad.

Esta guía está pensada como un insumo para líderes de proyecto, desarrolladores, docentes e investigadores que buscan integrar IA de forma estratégica, informada y adaptable a las realidades de sus organizaciones.

A continuación, se presenta la clasificación estructurada de herramientas IA, organizada por fases del ciclo de vida ágil en base a la revisión de los anteriores artículos: planificación y estimación, desarrollo, pruebas, documentación y retrospectiva:

1. Fase de planificación y estimación

Durante la planificación, los equipos definen el alcance del sprint, identifican las tareas a realizar y estiman los recursos necesarios. En este contexto, se muestra en

la Tabla 5, las herramientas de IA contribuyen significativamente mediante la predicción del esfuerzo, la priorización de tareas y la organización automatizada del backlog. Estas soluciones permiten una planificación más objetiva y eficiente, reduciendo la incertidumbre propia de las estimaciones iniciales.

Tabla 5

Herramientas IA de planificación

Herramienta	Función principal	Metodologías ágiles aplicadas
Azure AI Metrics	Predicción de esfuerzo y tiempo	Scrum, Kanban
Google AutoML Tables	Análisis predictivo de requisitos	XP, Scrum
AI Sprint Planner (custom tools)	Planificación de backlog y prioridades	Scrum, XP

Nota. Las herramientas destacadas en esta fase favorecen la toma de decisiones informadas desde el inicio del ciclo de desarrollo. Permiten anticipar cuellos de botella, asignar tiempos más realistas y establecer prioridades basadas en datos, lo que mejora la precisión de los compromisos asumidos por el equipo.

Prompt sugerido para esta fase:

"Analiza los datos históricos de esfuerzo y duración de tareas en proyectos ágiles similares y genera una predicción estimada por historia de usuario para el siguiente sprint. Formato deseado: tabla con tarea, esfuerzo estimado (horas) y nivel de confianza."

2. Fase de desarrollo y codificación

En esta etapa, los desarrolladores implementan las funcionalidades definidas. Las herramientas de IA que se muestran en la Tabla 6, se integran directamente en los entornos de desarrollo para asistir con sugerencias de código, generación automatizada de funciones y corrección de errores en tiempo real. Esto se traduce en mayor velocidad de desarrollo y menor tasa de defectos.

Tabla 6

Herramientas IA para desarrollo y codificación

Herramienta	Función principal	Metodologías ágiles aplicadas
GitHub Copilot	Generación de código asistida por IA	Scrum, XP
Tabnine	Autocompletado de código en tiempo real	Scrum
Amazon CodeWhisperer	Sugerencias contextuales de código	Scrum, Kanban

Nota. Durante el desarrollo activo del software, las herramientas de IA se convierten en asistentes directos para los programadores. GitHub Copilot, Tabnine y CodeWhisperer permiten acelerar la escritura de código mediante sugerencias contextuales, autocompletado inteligente y generación automática de fragmentos. Estas herramientas son especialmente útiles en metodologías como Scrum y XP, donde se prioriza la entrega continua y la calidad del código.

Prompt sugerido para la fase de desarrollo y codificación:

Dado el siguiente requerimiento funcional y lenguaje de programación especificado, genera una función bien estructurada que cumpla con el principio de responsabilidad única (SRP) y siga buenas prácticas de codificación (nombres claros, comentarios útiles, estructura modular). Incluye pruebas unitarias básicas asociadas a la función.

Este prompt promueve el uso responsable de herramientas IA como copilotos de programación, y al mismo tiempo refuerza estándares de calidad del software.

3. Fase de pruebas (testing)

Las pruebas aseguran que el software cumpla con los requisitos establecidos. La inteligencia artificial ofrece soporte mediante generación automática de casos de prueba dichas herramientas se mencionan en la Tabla 7, mismas que realizan la detección temprana de fallos y simulaciones inteligentes, lo que mejora la cobertura de pruebas y reduce errores en producción.

Tabla 7

Herramientas IA para realizar testing.

Herramienta	Función principal	Metodologías ágiles aplicadas
Diffblue Cover	Generación automática de pruebas unitarias	XP, Scrum
Test.AI	Testing funcional con reconocimiento visual	XP
ML Bug Predictors	Identificación automática de errores	Scrum, Kanban

Nota. El testing automatizado es uno de los campos donde la IA ha mostrado mayor madurez. Herramientas como Diffblue Cover y Test.AI permiten generar pruebas unitarias o funcionales sin intervención humana, reduciendo los errores manuales y mejorando la cobertura de pruebas. Además, modelos predictivos de errores ayudan a anticipar defectos en función de versiones anteriores. Esto resulta crucial en marcos como XP, donde las pruebas son un pilar fundamental.

Prompt sugerido para la fase de pruebas:

Genera un conjunto de casos de prueba automatizados para la siguiente función/módulo, considerando condiciones normales, límites y errores esperados. Utiliza un marco de pruebas compatible (por ejemplo, JUnit, PyTest o similar) y asegúrate de incluir validaciones claras, mensajes interpretables y cobertura de al menos el 80% de las ramas del código.

Este prompt está diseñado para maximizar la calidad de los entregables y fomentar una cultura de testing robusta dentro de equipos ágiles.

4. Fase de documentación y requisitos

Una documentación adecuada facilita el entendimiento y la continuidad del proyecto. Como se presenta en la Tabla 8, la IA puede generar documentación técnica automáticamente y convertir descripciones naturales en historias de usuario, reduciendo la carga administrativa y mejorando la trazabilidad de los cambios.

Tabla 8

Herramientas IA de documentación y requisitos

Herramienta	Función principal	Metodologías ágiles aplicadas
NLP Engines (spaCy, GPT-3)	Análisis semántico de requisitos	Scrum, Kanban
Mintlify, Doxygen IA	Documentación automática de código	XP, Scrum

Nota. La captura y análisis de requisitos con motores de lenguaje natural (NLP) como spaCy o GPT permite transformar descripciones informales en historias de usuario o tareas procesables. Herramientas como Mintlify también ayudan a documentar el código de forma automática, lo que es especialmente útil en proyectos con alta rotación de personal o

cambios frecuentes. Esta capacidad se adapta bien a contextos ágiles como Scrum y Kanban, donde los requisitos cambian de forma dinámica.

Prompt sugerido para la fase de documentación y requisitos:

A partir del siguiente requerimiento funcional redactado en lenguaje natural, genera: (1) una historia de usuario clara y estructurada con formato 'como [rol], quiero [acción] para [beneficio]', (2) los criterios de aceptación correspondientes, y (3) un bloque de documentación técnica compatible con herramientas como Mintlify o Doxygen. Asegúrate de emplear lenguaje técnico preciso y mantener coherencia con estándares ágiles."

Este prompt ayuda a transformar insumos informales en artefactos estructurados, mejorando la trazabilidad y la calidad de la comunicación técnica.

5. Fase de retrospectiva y mejora continua

En esta fase, los equipos reflexionan sobre el desempeño del sprint y buscan oportunidades de mejora. En la siguiente Tabla 9, las herramientas de IA permiten analizar datos del ciclo anterior y generar sugerencias automatizadas para optimizar procesos, identificar bloqueos y mejorar la colaboración.

Tabla 9

Herramientas IA de mejora continua

Herramienta	Función principal	Metodologías ágiles aplicadas
AI Retrospective Bots	Análisis de desempeño y recomendaciones	Scrum
Inteligencia de dashboards (Jira IA, Power BI + AI)	Visualización inteligente del avance	Kanban, Lean

Nota. En esta última fase, la IA contribuye a evaluar el rendimiento del equipo, detectar bloqueos recurrentes y proponer mejoras en los procesos. Herramientas como AI Retrospective Bots y dashboards inteligentes integrados con plataformas como Jira o Power BI permiten visualizar tendencias, identificar patrones de productividad y fomentar decisiones basadas en datos para iteraciones futuras. Son especialmente útiles en metodologías como Kanban y Lean, centradas en la mejora continua.

Prompt sugerido para la fase de retrospectiva y mejora continua:

Analiza los registros del sprint anterior (reuniones, tareas completadas, métricas de velocidad y bloqueos reportados) y genera un resumen de desempeño del equipo con sugerencias específicas de mejora. Clasifica las sugerencias en: procesos, herramientas, colaboración y comunicación. Incluye alertas o patrones recurrentes si los hubiere.

Este prompt está diseñado para fomentar una cultura de mejora continua basada en evidencia, utilizando IA como herramienta de análisis retrospectivo y soporte a la toma de decisiones del equipo ágil.

Observación: Cada instrucción está orientada a facilitar una interacción efectiva con modelos de lenguaje, optimizando la obtención de resultados relevantes, estructurados y alineados con los objetivos ágiles del proyecto.

Además, estos prompts pueden y deben ser adaptados al nivel de madurez digital del equipo, la complejidad del proyecto y las características específicas de las herramientas de inteligencia artificial utilizadas. No se trata de comandos rígidos, sino de guías flexibles que fomentan el pensamiento crítico, la mejora continua y la colaboración entre humanos e inteligencia artificial.

Su aplicación progresiva permitirá a los equipos:

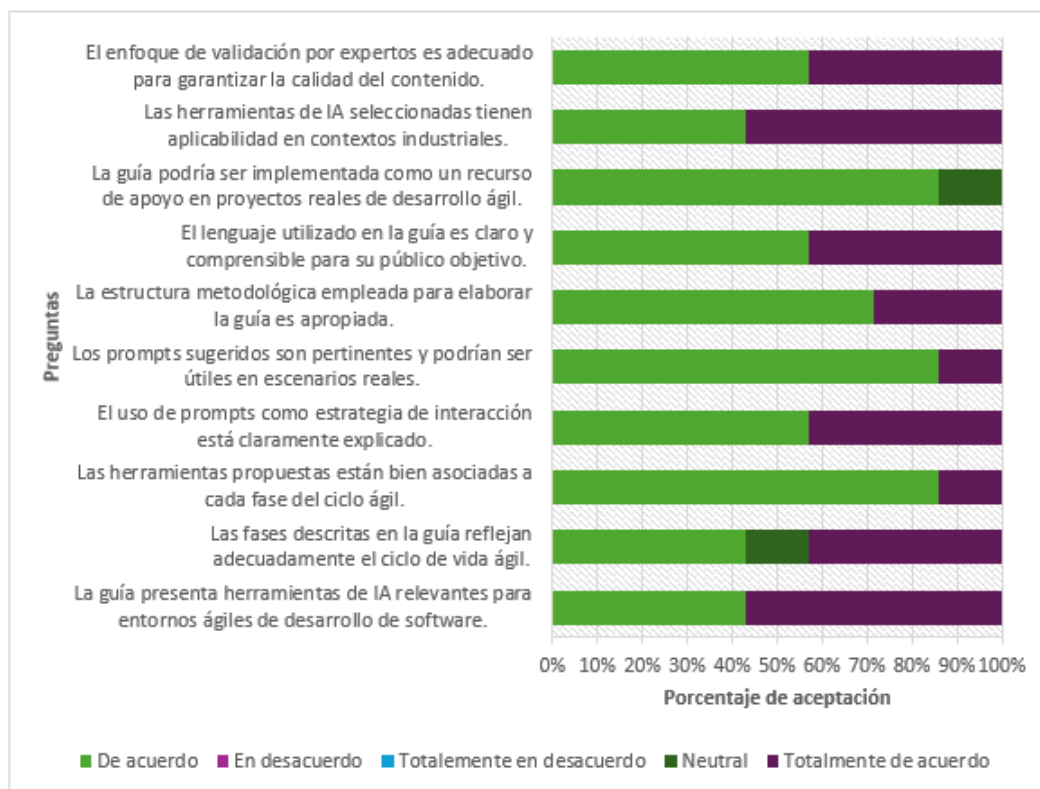
- Desarrollar mejores prácticas de interacción con IA generativa.
- Fortalecer la autonomía y productividad sin perder control técnico.
- Consolidar una cultura ágil con apoyo en tecnologías emergentes.

Se recomienda revisar, adaptar y versionar estos prompts a lo largo del tiempo, conforme evolucionen los modelos de lenguaje y las necesidades del entorno de desarrollo.

A continuación, se muestra el gráfico de aceptación por parte de expertos en Software que validan la propuesta de la guía realizada para el desarrollo de software.

Figura 4

Resultados de la encuesta a expertos en Software acerca de la guía desarrollada



Nota. El gráfico muestra una alta concentración de respuestas en las categorías “De acuerdo” y “Totalmente de acuerdo” para todos los ítems evaluados, lo que refleja una percepción positiva y consistente sobre la guía de herramientas de IA en entornos de desarrollo ágil. Ningún ítem recibió valoraciones de “Totalmente en desacuerdo” o “En desacuerdo”, lo cual evidencia que no existen observaciones críticas significativas por parte de los evaluadores. Sin embargo, la presencia de algunas respuestas “Neutral”, especialmente en ítems relacionados con la implementación práctica y la explicación de los prompts, sugiere oportunidades de mejora específicas. En conjunto, estos resultados respaldan la validez del contenido, su claridad comunicativa y su aplicabilidad general en contextos profesionales, posicionando a la guía como un recurso útil y pertinente.

4. Discusión

Los resultados de esta investigación evidencian el papel cada vez más determinante de la inteligencia artificial (IA) como catalizadora de la agilidad en los procesos de desarrollo de software. En términos generales, los hallazgos muestran que la incorporación de herramientas basadas en IA está transformando las prácticas tradicionales de programación, gestión y aseguramiento de la calidad. En particular, las herramientas de generación automática de código han permitido reducir significativamente los tiempos de implementación de funcionalidades, lo que optimiza los ciclos de entrega y libera a los desarrolladores para concentrarse en tareas de mayor complejidad y valor agregado (Kalava, 2024; Seppälä, 2024). Asimismo, el uso de plataformas de Inteligencia Artificial como Servicio (AlaaS) ha favorecido la automatización de procesos clave como el análisis, la integración continua y la detección temprana de errores, incrementando la eficiencia operativa de los equipos (Monsalves et al., 2024; Kumar, 2024).

En cuanto a la fase de pruebas, la automatización del *testing* mediante IA ha mejorado la cobertura y precisión de los ensayos, especialmente en metodologías ágiles como *Extreme Programming* (XP), donde la validación continua resulta esencial para mantener la calidad del producto (Dey et al., 2022). De igual modo, las herramientas de procesamiento del lenguaje natural (NLP) se consolidan como un recurso estratégico en la captura y análisis de requisitos, facilitando la priorización de historias de usuario y la planificación de los *backlogs* en equipos distribuidos o de rápida iteración (Upadhyaya, 2022; Yassine et al., 2023).

No obstante, los resultados también exponen desafíos importantes que deben ser considerados en la adopción de estas tecnologías. Entre ellos destacan la posible dependencia excesiva de herramientas automáticas, lo que podría reducir la capacidad analítica y crítica de los desarrolladores; las dificultades inherentes a la gestión ética y segura de datos sensibles durante los procesos de documentación y planificación; y la necesidad de promover una capacitación continua que permita interpretar adecuadamente los resultados generados por los modelos de IA.

Finalmente, se enfatiza la urgencia de adaptar estas soluciones al contexto latinoamericano, donde una proporción considerable de pequeñas y medianas empresas tecnológicas aún no ha incorporado plenamente metodologías ágiles ni herramientas inteligentes en sus flujos de trabajo. En este sentido, la transferencia tecnológica, la formación profesional y la creación de marcos regulatorios adecuados serán factores determinantes para garantizar una adopción sostenible, ética y contextualizada de la inteligencia artificial en el ecosistema del desarrollo de software regional.

5. Conclusiones

El presente estudio permitió sistematizar las principales herramientas de inteligencia artificial aplicadas en entornos de desarrollo ágil, identificando sus beneficios, niveles de adopción y fases del ciclo de vida en las que generan mayor impacto. La evidencia recopilada muestra que la IA ha dejado de ser una tecnología emergente para convertirse en un componente estratégico dentro de prácticas ágiles como Scrum y XP,

favoreciendo la automatización de tareas críticas, la optimización del tiempo de entrega y la mejora continua del software.

Una de las principales contribuciones de esta investigación es la elaboración de una guía funcional que clasifica herramientas IA según su aplicación en fases específicas del desarrollo ágil. Esta guía constituye un recurso útil para equipos de desarrollo, líderes de proyecto e instituciones educativas que buscan integrar IA de forma práctica, informada y alineada con marcos metodológicos consolidados.

Sin embargo, su implementación requiere superar desafíos como la capacitación constante del talento humano, la gestión responsable de datos sensibles y la adaptación contextual a las realidades de organizaciones latinoamericanas, donde la transformación digital aún se encuentra en proceso de consolidación.

Finalmente, se plantea como línea futura de investigación la validación empírica de esta guía en entornos reales, mediante estudios de caso, pilotos controlados o investigaciones participativas que permitan evaluar su efectividad en términos de productividad, calidad del software y madurez organizacional.

Referencias Bibliográficas

- Isaqaf, W., et al. (2022). AI metrics and measurement in agile transformations. *International Journal of Agile Systems and Management*, 15(1), 22–36. <https://doi.org/10.1504/IJASM.2022.10046912>
- Armijos Ortega, L. M., Vélez Macas, C. A., & Lojan Cueva, E. L. (2024). Estudio de la adopción de metodologías ágiles en proyectos de desarrollo de software en la Región 7 del Ecuador. *Revista Espacios*, 45(4), 18–27.
- Arora, A., et al. (2022). AI-supported Scrum retrospectives in distributed teams. *Procedia Computer Science*, 203, 125–132. <https://doi.org/10.1016/j.procs.2022.03.015>
- Ayalew, A., et al. (2022). Adoption of AI in agile development: An industry survey. *Journal of Systems and Software*, 187, 111237. <https://doi.org/10.1016/j.jss.2022.111237>
- Birje, M. N., et al. (2023). AI-driven monitoring and optimization in agile projects. *Software: Practice and Experience*, 53(7), 1234–1251. <https://doi.org/10.1002/spe.3085>
- Cabrero-Daniel, B. (2023). AI for agile development: A meta-analysis [Tesis de maestría, Universidad de Gothenburg].
- Choudhury, A., et al. (2023). Machine learning in agile testing frameworks. *International Journal of Software Engineering and Knowledge Engineering*, 33(5), 405–420. <https://doi.org/10.1142/S0218194023500216>
- De França, B., et al. (2022). MLOps practices in agile software teams. *Journal of Systems Architecture*, 127, 102452. <https://doi.org/10.1016/j.sysarc.2022.102452>
- Dey, R., Khan, S. A., & Mollah, M. B. (2022). Improving agile testing with artificial intelligence. *Journal of Software Engineering Research and Development*, 10(2), 45–61.

- Gaete, J., Villarroel, R., Figueroa, I., Cornide-Reyes, H., & Muñoz, R. (2021). Enfoque de aplicación ágil con Scrum, Lean y Kanban. *Ingeniare. Revista Chilena de Ingeniería*, 29(1), 141–157.
- Guerrero, R., et al. (2023). La inteligencia artificial en la calidad del software: Una revisión sistemática de la literatura. *UNESUM-Ciencias*, 4(1), 75–90.
- Hanssen, G. K., Mikalsen, M., & Haugset, B. (2023). Agile and AI: Bridging engineering practices with intelligent automation. *IEEE Software*, 40(2), 28–35. <https://doi.org/10.1109/MS.2023.3242531>
- Jovanović, M., et al. (2023). Applying LLMs in agile development projects. *Proceedings of the 2023 International Conference on Agile Software Engineering*, 145–152.
- Kalava, S. (2024). AI-powered development: How artificial intelligence is shaping software engineering. *GitHub Pages*. <https://kalava.dev/aipowered-dev>
- Kumar, S. (2024). *Artificial intelligence in software engineering: A systematic exploration of AI-driven development*. Springer.
- Lee, H., & Hwang, J. (2022). Predictive analytics for agile project management. *Journal of Project Management*, 7(3), 33–45. <https://doi.org/10.5267/j.jpm.2022.2.005>
- Limam, M., et al. (2023). AI-driven product management dashboards for agile teams. *International Journal of Information Management*, 73, 102529. <https://doi.org/10.1016/j.ijinfomgt.2022.102529>
- Martínez, J., et al. (2022). Caracterización y comparación de metodologías ágiles y tradicionales en desarrollo de software. *Revista Iberoamericana de Tecnología*, 28(2), 95–110.
- Monsalves, D., Olivares, P., Riquelme, F., & Cornide-Reyes, H. (2024). Inteligencia artificial como servicio. *Ingeniare. Revista Chilena de Ingeniería*, 32(1), 11–20.
- Najjar, N., et al. (2022). Agile risk management augmented by artificial intelligence. *International Journal of Risk and Contingency Management*, 11(1), 21–35. <https://doi.org/10.4018/IJRCM.2022010103>
- Oliveira, D., et al. (2023). Lean-agile methods for AI project management frameworks. *Journal of Modern Project Management*, 11(3), 22–34.
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffman, T. C., Mulrow, C. D., ... & Moher, D. (2021). The PRISMA 2020 statement: An updated guideline for reporting systematic reviews. *BMJ*, 372, n71. <https://doi.org/10.1136/bmj.n71>
- Rendón De La Torre, A. (2024). Metodologías ágiles en la era de la inteligencia artificial. *Prisma*, 7(1), 44–53.
- Salazar-Cedeño, A. N., Demera, A. E., & Santana, G. A. (2024). Fundamentación teórica de la inteligencia artificial en el desarrollo de aplicaciones móviles. *TESLA Revista Científica*, 3(2), e223.

- Seppälä, E. (2024). Using AI tools in software development: A software engineering experiment at Hansen Technologies Finland [Tesis de maestría, University of Helsinki].
- Suto, M., et al. (2022). Agile project estimation with machine learning techniques. *International Journal of Project Management*, 40(6), 874–882. <https://doi.org/10.1016/j.ijproman.2022.05.001>
- Tupsakhare, P. (2022). Enhancing agile methodologies with artificial intelligence: Driving efficiency and innovation. *European Journal of Advances in Engineering and Technology*, 9(10), 66–71.
- Upadhyaya, N. (2022). The role of artificial intelligence in software development: A literature review. GlobalLogic Inc.
- Vieira, M., et al. (2023). Agile big data projects enhanced by AI toolchains. *Journal of Big Data*, 10(1), 1–20. <https://doi.org/10.1186/s40537-023-00756-w>
- Yassine, A., El-Sayed, M., & Fahmy, H. (2023). AI-based requirement engineering for agile teams. In *Proceedings of the International Conference on Software Engineering and Knowledge Engineering (SEKE 2023)* (pp. 45–53). Knowledge Systems Institute.

AGRADECIMIENTOS

Agradezco profundamente a mi esposa Janeth, cuyo amor, paciencia y constante apoyo han sido fundamentales en cada etapa de este proceso. A mis hijos, Alisson, Ariadne e Iker, por ser mi fuente inagotable de inspiración y alegría. Cada logro es también suyo.

CONFLICTO DE INTERESES

“Los autores declaran no tener ningún conflicto de intereses”.