



UNIVERSIDAD NACIONAL DE CHIMBORAZO
FACULTAD DE CIENCIAS DE LA EDUCACIÓN HUMANAS Y
TECNOLÓGICAS
CARRERA DE PEDAGOGÍA DE LAS CIENCIAS
EXPERIMENTALES: INFORMÁTICA

Título:

Uso de la Inteligencia Artificial Generativa en el aprendizaje de la
programación: una revisión sistemática de la literatura

Trabajo de Titulación para optar al título de:

Licenciado en Pedagogía de las Ciencias Experimentales: Informática

Autor:

Viñamagua Sarango Iván Filiberto

Tutor:

Dr. Patricio Ricardo Humanante Ramos

Riobamba, Ecuador. 2026

DECLARATORIA DE AUTORÍA

Yo, **Iván Filiberto Viñamagua Sarango**, con cédula de ciudadanía **1105660557**, autor del trabajo de investigación titulado: **Uso de la Inteligencia Artificial Generativa en el aprendizaje de la programación: una revisión sistemática de la literatura**, certifico que la producción, ideas, opiniones, criterios, contenidos y conclusiones expuestas son de mi exclusiva responsabilidad.

Asimismo, cedo a la Universidad Nacional de Chimborazo, en forma no exclusiva, los derechos para su uso, comunicación pública, distribución, divulgación y/o reproducción total o parcial, por medio físico o digital; en esta cesión se entiende que el cesionario no podrá obtener beneficios económicos. La posible reclamación de terceros respecto de los derechos de autor (a) de la obra referida, será de mi entera responsabilidad; librando a la Universidad Nacional de Chimborazo de posibles obligaciones.

En Riobamba, 25 de junio del 2026.



Iván Filiberto Viñamagua Sarango

C.I: 1105660557

DICTAMEN FAVORABLE DEL PROFESOR TUTOR

Quien suscribe, **Dr. Patricio Ricardo Humanante Ramos**, catedrático adscrito a la Facultad de Ciencias de la Educación, Humanas y Tecnológicas, por medio del presente documento certifico haber asesorado y revisado el desarrollo del trabajo de investigación titulado: **Uso de la Inteligencia Artificial Generativa en el aprendizaje de la programación: una revisión sistemática de la literatura**, bajo la autoría de **Iván Filiberto Viñamagua Sarango**; por lo que se autoriza ejecutar los trámites legales para su sustentación.

Es todo cuanto informar en honor a la verdad; en Riobamba, a los 25 del mes de junio de 2026



Dr. Patricio Ricardo Humanante Ramos

C.I:0602767204

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

Quienes suscribimos, catedráticos designados Miembros del Tribunal de Grado para la evaluación del trabajo de investigación **Uso de la Inteligencia Artificial Generativa en el aprendizaje de la programación: una revisión sistemática de la literatura**, presentado por **Iván Filiberto Viñamagua Sarango**, con cédula de identidad número **1105660557**, bajo la tutoría de **Dr. Patricio Ricardo Humanante Ramos**; certificamos que recomendamos la APROBACIÓN de este con fines de titulación. Previamente se ha evaluado el trabajo de investigación y escuchada la sustentación por parte de su autor; no teniendo más nada que observar.

De conformidad a la normativa aplicable firmamos, en Riobamba 25 de junio de 2026

Jorge Eduardo Fernández Acevedo, Mgs.
PRESIDENTE DEL TRIBUNAL DE GRADO



María Belén Piñas Morales, Mgs.
MIEMBRO DEL TRIBUNAL DE GRADO



Hernan Ramiro Pailiacho Yucta, Mgs.
MIEMBRO DEL TRIBUNAL DE GRADO



CERTIFICADO ANTIPLAGIO



Dirección
Académica
VICERRECTORADO ACADÉMICO



UNACH-RGF-01-04-08.15
VERSIÓN 01: 06-09-2021

CERTIFICACIÓN

Que, **IVÁN FILIBERTO VIÑAMAGUA SARANGO** con CC: **1105660557** estudiante de la Carrera de **PEDAGOGÍA DE LAS CIENCIAS EXPERIMENTALES: INFORMÁTICA**, Facultad de **CIENCIAS DE LA EDUCACIÓN, HUMANAS Y TECNOLOGÍAS**; ha trabajado bajo mi tutoría el trabajo de investigación titulado "**USO DE LA INTELIGENCIA ARTIFICIAL GENERATIVA EN EL APRENDIZAJE DE LA PROGRAMACIÓN: UNA REVISIÓN SISTEMÁTICA DE LA LITERATURA**", cumple con el **7 %**, de acuerdo al reporte del sistema Anti plagio **COMPILATIO**, porcentaje aceptado de acuerdo a la reglamentación institucional, por consiguiente autorizo continuar con el proceso.

Riobamba, 12 de junio de 2026

PhD. Patricio Humanante Ramos
TUTOR

DEDICATORIA

En primer lugar, toda mi gratitud a Dios, por ser mi guía y mi fortaleza en los momentos más duros de mi vida, por iluminar mi camino y darme la sabiduría y la perseverancia para culminar este proyecto.

Este logro está dedicado a mis tres pilares fundamentales de mi vida: a mi madre, Melva Sarango y a mi padre, Ángel Viñamagua que ha luchado incansablemente por verme cumplir mis sueños, cuyo amor incondicional y sacrificios han sido el soporte para ser todo lo que soy y lo que, alcanzado, también a mis hermanos, Mayquel Viñamagua y William Viñamagua por su apoyo, complicidad y por ser mis compañeros constantes que me han ayudado a cumplir este sueño en mi vida.

A ellos, les debo todo.

Iván Filiberto Viñamagua Sarango

AGRADECIMIENTO

Agradezco, en primer lugar, a Dios por brindarme la fortaleza, la constancia y la sabiduría necesarias para culminar este trabajo de investigación.

Expreso mi más sincero agradecimiento a mi tutor Dr. Patricio Ricardo Humanante Ramos quien con su orientación, conocimientos y acompañamiento académico contribuyeron de manera fundamental al desarrollo de la tesis. Sus observaciones y recomendaciones permitieron enriquecer el proceso investigativo y fortalecer los resultados obtenidos.

Finalmente, agradezco a mi familia y a todas las personas que, de manera directa o indirecta, brindaron su apoyo moral y motivación constante, lo cual fue esencial para alcanzar este objetivo académico.

Iván Filiberto Viñamagua Sarango

ÍNDICE GENERAL

DECLARATORIA DE AUTORÍA

DICTAMEN FAVORABLE DEL PROFESOR TUTOR

CERTIFICADO DE LOS MIEMBROS DEL TRIBUNAL

CERTIFICADO ANTIPLAGIO

DEDICATORIA

AGRADECIMIENTO

RESUMEN

ABSTRACT

CAPÍTULO I.	14
1. INTRODUCCIÓN.....	14
1.1 ANTECEDENTES	15
1.2 PLANTEAMIENTO DEL PROBLEMA	15
1.3 JUSTIFICACIÓN	16
1.4 Objetivos.....	17
1.4.1 General.....	17
1.4.2 Específicos.....	17
CAPÍTULO II.....	18
2. MARCO TEÓRICO.	18
2.1 Estado del arte	18
2.2 Inteligencia artificial.....	19
2.2.1 Fundamentos de la inteligencia artificial generativa	19
2.3 Modelos de lenguaje a gran escala (LLM) y programación	21
2.4 Otras herramientas de IAG en la programación	21
2.5 Herramientas que utilizan IAG en el entorno de programación	22
2.5 ChatGPT en la enseñanza de la programación	23
2.6 GitHub copilot y la programación asistida	24
2.7 Evolución histórica del aprendizaje de la programación.....	24
2.8 Retos cognitivos del aprendizaje de la programación	25
2.9 La IA generativa como mediadora del aprendizaje	26
2.10 Beneficios pedagógicos del uso de la IAG en programación.....	27
2.11 Riesgos y desafíos del uso de IAG en educación	28

2.12	Aprendizaje personalizado e IAG.....	29
2.13	Metodologías pedagógicas que integran IAG	30
2.14	Ética e integridad académica con IAG	31
2.15	Brecha digital y equidad en el acceso a la IAG	32
CAPÍTULO III.		33
3.	METODOLOGIA.....	33
3.1	Enfoque de la investigación.....	33
3.2	Alcance de la Investigación.....	33
3.3	Tipo de Investigación.	33
3.4	Diseño de investigación.....	33
CAPÍTULO IV.		37
4.	RESULTADOS Y DISCUSIÓN	37
4.1	Propósito de la revisión de la literatura	37
4.2	Protocolo de revisión.....	37
4.2.1	Preguntas de investigación	37
4.2.2	Estrategia de búsqueda	37
4.3	La búsqueda de la literatura.....	38
4.4	Pantalla de práctica.....	39
4.5	Procedimiento de extracción	40
4.6	Valoración de la calidad	41
4.7	Síntesis de la información.....	41
4.8	Redacción de la revisión.....	42
CAPÍTULO V.....		53
5.	CONCLUSIONES Y RECOMENDACIONES	53
5.1	CONCLUSIONES.....	53
5.2	RECOMENDACIONES	53
BIBLIOGRAFÍA		54
ANEXOS		56

ÍNDICE DE TABLAS

Tabla 1. Tipos de IA y su Aplicación en el Aprendizaje de la Programación.....	20
Tabla 2. Principales riesgos del uso de iag en el aprendizaje de programación.....	29
Tabla 3. Metodologías pedagógicas combinadas con IAG	30
Tabla 4. Búsqueda en bases de datos científicas	38
Tabla 5. Criterios para evaluar la calidad de las publicaciones.....	41
Tabla 6. Síntesis de la literatura sobre herramientas de Inteligencia Artificial Generativa	44
Tabla 7. Beneficios sobre el uso de la inteligencia artificial generativa en el aprendizaje de la programación	47
Tabla 8. Resumen de los estudios relacionados con los riesgos al utilizar Inteligencia Artificial Generativa	49
Tabla 9. Metodologías pedagógicas que se aplican en combinación con las herramientas IAG para el aprendizaje de la programación.	51

ÍNDICE DE FIGURAS

Figura 1. Pasos para realizar una SLR (Okoli, 2015).....	34
Figura 2. Resultados de la búsqueda de la literatura en bases de datos.....	39
Figura 3. Pantalla de práctica del proceso de selección de la literatura	40
Figura 4. Distribución temporal de las publicaciones incluidas en la revisión	42
Figura 5. Distribución.....	43
Figura 6. Uso de herramientas de inteligencia artificial generativa en el aprendizaje de la programación.....	44
Figura 7. Beneficios del uso de Inteligencia artificial generativa	47
Figura 8. Riesgos sobre el uso de la Inteligencia Artificial Generativa	49

RESUMEN

La presente investigación analiza el uso de la inteligencia artificial generativa en el aprendizaje de la programación, con énfasis en sus beneficios, riesgos y metodologías pedagógicas asociadas. A través de la revisión y análisis de estudios recientes, se examina la aplicación de herramientas como ChatGPT, GitHub, Copilot y otros, en diversos contextos educativos. Para esto se aplicó la metodología Systematic Literature Review, que incluye ocho pasos debidamente documentados, los cuales van desde el establecimiento del protocolo hasta la redacción de la revisión.

Los resultados evidencian que la inteligencia artificial generativa contribuye a reducir el costo y el esfuerzo del aprendizaje, facilita el acceso a recursos educativos, mejora la productividad y apoya la resolución de problemas de programación. No obstante, también se identifican limitaciones y riesgos, como la dependencia excesiva, la comprensión superficial de los contenidos y la necesidad de habilidades metacognitivas para un uso efectivo. Asimismo, se destaca que los beneficios reales en el rendimiento académico dependen de una adecuada orientación pedagógica y de la capacitación de estudiantes y docentes. En conclusión, la inteligencia artificial generativa representa un recurso complementario con alto potencial educativo, siempre que su integración se realice de manera guiada, crítica y alineada con los objetivos de aprendizaje.

Palabras claves: Inteligencia artificial generativa, aprendizaje de programación, IA generativa, aprendizaje asistido por IA y modelos de IA generativa para el aprendizaje.

ABSTRACT

This study analyzes the use of generative artificial intelligence in learning programming, with an emphasis on its benefits, risks, and associated pedagogical methodologies. Through the review and analysis of recent studies, the application of tools such as ChatGPT, GitHub Copilot, and other generative artificial intelligence systems in different educational contexts is examined. For this purpose, the Systematic Literature Review methodology was applied, comprising eight duly documented steps, ranging from the establishment of the review protocol to the writing of the final report. The results show that generative artificial intelligence contributes to reducing the cost and effort involved in learning, facilitates access to educational resources, improves productivity, and supports the resolution of programming problems. However, several limitations and risks were also identified, including excessive dependence on these tools, a superficial understanding of the content, and the need to develop metacognitive skills for their effective use. Furthermore, the findings highlight that the actual benefits for academic performance depend on appropriate pedagogical guidance and adequate training for both students and teachers. In conclusion, generative artificial intelligence represents a complementary resource with significant educational potential, provided that its integration is guided, critical, and aligned with the intended learning objectives.

Keywords: generative artificial intelligence, programming learning, generative AI, AI-assisted learning, generative AI models for learning.



Reviewed by:
MsC. Edison Damian Escudero
ENGLISH PROFESSOR
C.C.0601890593

CAPÍTULO I.

1. INTRODUCCIÓN.

La inteligencia artificial (IA) ha revolucionado múltiples disciplinas como en el ámbito educativo de la programación donde no ha sido la excepción. En este contexto, la inteligencia artificial generativa (IAG) ha asumido una función primordial, no solo por su capacidad para facilitar la creación de contenido, sino también por su potencial para mejorar el proceso de enseñanza-aprendizaje.

La Inteligencia Artificial Generativa emerge como una herramienta revolucionaria para la educación, capaz de cambiar las prácticas pedagógicas tradicionales. En Educación General Básica (EGB) la inteligencia artificial generativa (IAG) personaliza la enseñanza y el aprendizaje atendiendo las necesidades de cada estudiante, esto es especialmente relevante en un contexto donde las metodologías educativas se pueden adaptar a un alumnado diverso y a sus diferentes ritmos y estilos de aprendizaje (Salazar Sisalima et al., 2024).

Hoy en día el aprender a programar se ha vuelto un reto ya que los estudiantes cuentan con varios factores negativos como, por ejemplo, se enseña programación solo a estudiantes de niveles de educación superior o de instituciones de EGB privadas, sin incluir esta enseñanza en las instituciones fiscales. Al no desarrollar estas habilidades de programación a su debido tiempo los estudiantes realizan uso de la inteligencia artificial generativa para resolver estos problemas y en ocasiones solo copian y pegan sin emplear su debido pensamiento crítico.

Algunos defienden su potencial como una gran ayuda para los estudiantes en sus trabajos académicos, mientras que otros alertan sobre el riesgo de que su uso inadecuado afecte el desarrollo de las competencias para resolver problemas de forma autónoma.

Así, este proyecto de investigación tiene como objetivo principal analizar el uso de la Inteligencia Generativa en el aprendizaje de la programación mediante una revisión sistemática de la literatura. Para ello, se utilizará un protocolo de búsqueda riguroso, en este caso se utilizará SLR. Esta revisión permitirá identificar herramientas, beneficios, riesgos y metodologías, contribuyendo a optimizar la enseñanza de programación.

El presente trabajo está compuesto por cinco capítulos, complementados de la siguiente manera:

CAPÍTULO I. Se expone la introducción al tema de la investigación, antecedentes sobre el uso de la Inteligencia Artificial Generativa (IAG) en educación, el planteamiento del problema, la justificación y los objetivos de la investigación. Este capítulo permite contextualizar y establecer la base para analizar cómo la IAG impacta en el aprendizaje de la programación.

CAPÍTULO II. Comprende el marco teórico, donde se revisa la literatura científica sobre la Inteligencia Artificial, los modelos de lenguaje a gran escala, la evolución histórica del aprendizaje de la programación, los retos cognitivos de los estudiantes, los beneficios y riesgos del uso de IAG, así como las metodologías pedagógicas que se pueden usar dependiendo del contexto.

CAPÍTULO III. Se expone el marco metodológico de la investigación, que comprende: un enfoque cualitativo, alcance descriptivo, tipo documental y el diseño de revisión sistemática de la literatura (SLR), este capítulo define los protocolos para recopilar, analizar y síntesis de la evidencia científica sobre el uso de IAG en el aprendizaje de la programación.

CAPÍTULO IV. Se presenta los hallazgos obtenidos del protocolo de la revisión sistemática de la literatura también se estructura los hallazgos según las preguntas de investigación: herramientas de IAG utilizadas, beneficios de la IAG, riesgos identificados del uso de la IAG y metodologías pedagógicas aplicadas. Adicionalmente se presentan ilustraciones y tablas que sintetizan la información de una forma estructurada.

CAPÍTULO V. En este apartado se incluye las conclusiones y las recomendaciones resultantes del exhaustivo análisis de los datos recabados durante la investigación. También se destacan las implicaciones pedagógicas, los beneficios y riesgos de la integración de IAG en la enseñanza de programación y se sugieren estrategias para su implementación efectiva en entornos educativos.

1.1 ANTECEDENTES

En los últimos años, ha aumentado el interés por el uso de la Inteligencia Artificial Generativa en el aprendizaje de la programación, debido al alcance y a la facilidad para acceder y hacer uso de la (IA) en las tareas, lo que ayuda resolviendo cualquier tipo de problema consultado, de manera inmediata.

En el trabajo de investigación de Chávez (2024), titulado “Integración de la inteligencia artificial generativa para el aprendizaje de fundamentos de programación: un aprendizaje de fundamentos de programación, para realizar el presente trabajo se utilizó el protocolo PRISMA.

Por otro lado, el trabajo de investigación de Rivera, Gonzabay y Mendoza (2024). titulado “La Influencia de las IA Generativas en la Enseñanza de Programación en Universidades”. Este estudio examina la influencia de las herramientas de inteligencia artificial generativa, como ChatGPT y GitHub Copilot, en la enseñanza de programación en universidades. A través de entrevistas semiestructuradas realizadas a estudiantes de la carrera de desarrollo de software en la Universidad Península de Santa Elena UPSE.

1.2 PLANTEAMIENTO DEL PROBLEMA

El uso de la Inteligencia Artificial Generativa (IAG) en la educación, específicamente en el aprendizaje de la programación, ha generado debates sobre su impacto en el desarrollo de habilidades cognitivas y técnicas en los estudiantes. (Sánchez et al., 2022).

Aunque herramientas como GitHub Copilot, ChatGPT y Codex facilitan la generación automática de código, existe preocupación por posibles efectos negativos, como: la dependencia excesiva de la IA, limitando el desarrollo del pensamiento algorítmico, falta

de comprensión profunda de los conceptos de programación al recibir soluciones automatizadas y el riesgo de plagio o uso poco ético en evaluaciones académicas. Este fenómeno afecta principalmente a:

- Alumnos: Si no logran encontrar un equilibrio entre el uso de la Inteligencia artificial generativa y el aprendizaje autónomo podrían sufrir consecuencias.
- Profesores: Diseñar estrategias que fusione la Inteligencia artificial generativa sin sacrificar la formación esencial.
- Instituciones académicas: Deben renovar sus normativas educativas para controlar la ética en el uso de estas herramientas. En este sentido, surge el siguiente problema de investigación:

¿Cómo se usa la inteligencia artificial generativa en el proceso de aprendizaje de la programación?

1.3 JUSTIFICACIÓN

La Inteligencia artificial (IA) en el ámbito educativo no constituye un fenómeno novedoso; sin embargo, la evolución de la Inteligencia artificial Generativa (IAG) ha incrementado el interés de investigadores y educadores por su capacidad para transformar los procesos de enseñanza-aprendizaje. Desde los primeros esfuerzos por incorporar tecnologías inteligentes en las aulas, por ejemplo, sistemas tutoriales inteligentes, la educación ha buscado aprovechar la IA para optimizar la personalización y accesibilidad del aprendizaje (Salazar Sisalima et al., 2024).

La IAG está siendo utilizada ampliamente en la educación (Salazar Sisalima et al., 2024), así en una de las áreas donde se aplica esto, es en la programación, como se puede ver en estudios publicados (Margarita Chávez-Boza & Ramiro Erazo-Moreta, 2024.), sin embargo revelan que hay una dependencia excesiva en la IAG, lo que provoca una limitación en la comprensión del código, ya que al obtener solamente la respuesta a los problemas planteados, los estudiantes solo se limitan a copiar sin entender el resultado y sin desarrollar su pensamiento crítico.

Con el gran avance de la IA aparecen desafíos para los docentes, en donde éstos deben afrontar los desafíos que trae el aplicar la IA en el aprendizaje de la programación ya que no se les debe prohibir el uso, sino mejor buscar maneras de como incorporar de una manera adecuada, para evitar su mal uso.

Para que el docente integre la inteligencia artificial generativa en el aula, es esencial la preparación previa y el fomento de actitudes positivas, lo que permitirá mejores resultados de aprendizaje. Una actitud positiva del docente es esencial para planificar e impartir las asignaturas con IAG, como el caso de la programación.

1.4 OBJETIVOS

1.4.1 GENERAL

Analizar el uso de la Inteligencia Artificial Generativa en el aprendizaje de la programación, mediante una revisión sistemática de la literatura.

1.4.2 ESPECÍFICOS

- Definir un protocolo de búsqueda para realizar una Systematic Literature Review (SLR) sobre el uso de la inteligencia artificial generativa en el aprendizaje de la programación.
- Recopilar las principales publicaciones académicas, considerando criterios de inclusión y calidad.
- Sintetizar los principales resultados en función de las preguntas de investigación planteadas para la SLR

CAPÍTULO II.

2. MARCO TEÓRICO.

2.1 Estado del arte

Dentro de los principales hallazgos académicos publicados, respecto a la temática en cuestión, se tiene la investigación de Figueiredo y García-Peñalvo (2020), quienes señalan que, cuando se implementa Inteligencia Artificial (IA) en cursos de programación, se utilizan diversas estrategias de aprendizaje para mejorar la experiencia educativa de los estudiantes. Algunas de las estrategias comunes incluyen:

- **Personalización del aprendizaje:** La IA posibilita la adaptación del contenido y la complejidad de las lecciones en función de las necesidades individuales de cada estudiante, proporcionando un enfoque más individualizado y eficaz.
- **Respuesta inmediata:** Los sistemas basados en la IA tienen la capacidad de proporcionar retroalimentación inmediata sobre el rendimiento académico de los alumnos, detectando fallos y proponiendo recomendaciones para una mejorara del estudiante, lo que facilita el proceso de adquisición de conocimiento.
- **Orientación inteligente:** Los sistemas de asistencia inteligente simulan la conversación con una persona, guiando a los estudiantes por los conceptos esenciales y ofreciendo asistencia personalizada a cada estudiante.
- **Análisis de datos y monitoreo del avance:** La IA puede emplearse para realizar análisis de rendimiento de los estudiantes, la identificación de patrones de aprendizaje y propuestas de sugerencias para la optimización del proceso educativo, posibilitando un monitoreo minucioso de los avances de los estudiantes.

Otra investigación corresponde a “La inteligencia artificial en el aprendizaje de la programación, sus aportes y preocupaciones desde el punto de vista de los estudiantes de ingeniería” realizada en Ecuador, un estudio transversal y no experimental con estudiantes de ingeniería de software (N=143) en la Universidad de las Fuerzas Armadas (ESPE) mostró una amplia adopción de herramientas de Inteligencia Artificial. Los resultados muestran que el 94.4% de los participantes emplea estas tecnologías, y un 3.5% tiene planes de usarlas en el futuro. Solo un 2.1% mostró resistencia a su adopción, evidenciando la amplia aceptación entre los futuros profesionales del sector. El análisis utilizó un cuestionario validado por especialistas para medir la autopercepción de los estudiantes. (Suárez et al., 2022, como se cita en Cela, Granda & Villamarín pág. 114).

También Duong y otros autores (2022), identifican la práctica individualizada y la retroalimentación automática y oportuna como estrategias clave, ofreciendo a los estudiantes explicaciones detalladas y sugerencias para mejorar sus habilidades de codificación y la resolución de problemas.

Yadav y otros autores (2022), fundamentan su propuesta con el uso de herramientas basadas en Inteligencia Artificial Generativa, como ChatGPT de OpenAI, para apoyar el aprendizaje de fundamentos de programación. Estas herramientas permiten a los estudiantes

interactuar con un sistema de IA para recibir retroalimentación y ayuda en sus procesos de aprendizaje en programación.

2.2 Inteligencia artificial

La inteligencia artificial es una tecnología que permite que las computadoras y otros dispositivos electrónicos como teléfonos inteligentes, tabletas etc., cuenten con habilidades humanas como el aprendizaje, la comprensión, la resolución en situaciones, la decisión, la invención y la independencia plenas (Guacán Tandayamo et al., 2023).

Las aplicaciones y los dispositivos con IA pueden reconocer objetos y entender lo que ven. Esto les permite interpretar y responder al lenguaje y también aprender de nueva información y experiencias. También hacen recomendaciones precisas a usuarios y expertos. También toman decisiones independientes, lo que significa que no se requiere ninguna acción humana. Por ejemplo, en coches autosuficientes (Stryker & Kavlakoglu, 2024).

Las aplicaciones y dispositivos equipados con IA pueden:

- Ver e identificar objetos
- Entender y responder al lenguaje humano
- Aprender de nueva información y experiencia
- Hacer recomendaciones detalladas a usuarios y expertos
- Actuar de manera independiente, reemplazando la necesidad de inteligencia o intervención humana.

2.2.1 Fundamentos de la inteligencia artificial generativa

La inteligencia artificial generativa (IAG) constituye una rama de la inteligencia artificial que produce contenido nuevo a partir de patrones aprendidos en grandes corpus de datos mediante arquitecturas de redes neuronales profundas, además, según Lim et al. (2023), los modelos generativos, como el generador preentrenado de transformador (GPT, por sus siglas en inglés), operan mediante la predicción de la siguiente unidad lingüística condicionado al contexto anterior. Esto les permite imitar procesos creativos humanos, lo que en la educación permite ofrecer respuestas a las preguntas de los estudiantes. Lo que produce es semánticamente útil, funciona y tiene ventajas competitivas para una amplia gama de tareas.

Baidoo-Anu y Owusu (2023) sostienen que la capacidad de IAG para el entrenamiento no supervisado y auto atencional le confiere una gran flexibilidad para llevar a cabo distintos tipos de tareas educativas, desde tutoría conversacional hasta generación de ejercicios. Los autores destacan que esta tecnología cambia la naturaleza del conocimiento, que ya no es un almacenamiento de datos, sino un proceso que se co-construye con el usuario. La IA no proporciona la información, establece bajo un diálogo iterativo significados, esto transforma las prácticas de la búsqueda y el consumo informativos tradicionales.

Los autores también destacan que la IAG está apoyada por tres pilares técnicos que son arquitecturas de deep learning, datasets de aprendizaje masivo y poder de cómputo

escalable sobre GPU/TPU (Sabzalieva y Valentini, 2023). Durante la primera década de este siglo, la educación, el aprendizaje y la neurociencia se han ido enlazando. Esta situación, como se conoce en el ámbito educativo, explica el salto cualitativo que ha experimentado este campo, sobre todo a partir del lanzamiento público de ChatGPT en noviembre de 2022, el cual es considerado un hito histórico para la educación superior y la investigación pedagógica por las autoras.

Por lo tanto, análisis ofrecido por Bommasani et al. (2021) los sitúa bajo la clasificación de modelos fundacionales; estos son sistemas, que son entrenados en conjuntos de datos masivos, lo que hace que su comportamiento emerja de forma no explícita. Esta característica marca una diferencia clave con respecto a las generaciones anteriores de IAG, que eran programas que ejecutaban reglas específicas. En cambio, la IAG es un sistema que realiza comportamientos complejos. Esto se produce por el aprendizaje estocástico de distribuciones mixtas muy complejas, que acaban generando comportamientos sorprendentes. Esto incluso permite unas capacidades razonamiento aritmético, traducción en varias lenguas o generación de código en lenguajes programación que no estaban claramente etiquetados en el proceso de aprendizaje.

Los aportes coinciden en que la IAG no es solo un motor de búsqueda mejorado, sino un sistema que reconfigura las dinámicas del aprendizaje. Mientras Lim resalta el aspecto arquitectural, Baidoo-Anu enfoca en el aspecto epistemológico del cambio, Sabzalieva y Valentini lo sitúan en un contexto histórico de disrupción y Bommasani lo sitúa en una categoría más amplia de tecnología. En esta investigación las cuatro perspectivas se consideran complementarias, ya que la IAG como variable independiente requiere que se conozca tanto su funcionamiento interno como su impacto en la construcción del saber, en particular su aplicación en un ámbito tan procedimental y formal como lo es la programación.

Tabla 1
Tipos de IA y su Aplicación en el Aprendizaje de la Programación

Tipo de IA	Característica principal	Ejemplo en programación	Limitación
Reactiva	Responde a estímulos sin memoria del pasado.	Linters básicos que detectan errores de sintaxis.	Sin contexto histórico.
Predictiva	Estima resultados a partir de datos históricos.	Autocompletado tradicional basado en frecuencia.	Requiere reentrenamiento frecuente.
Generativa	Produce contenido nuevo a partir de modelos entrenados.	ChatGPT y Copilot que generan funciones completas.	Riesgo de alucinaciones.
Fundacional	Modelo amplio adaptable a múltiples tareas.	LLM que se ajustan para tutoría, depuración y generación.	Requiere mucho cómputo.

Nota. Tipos de IA y ejemplos aplicados al aprendizaje de programación.

2.3 Modelos de lenguaje a gran escala (LLM) y programación

Las herramientas de IAG para la programación se sustentan en los modelos de Lenguaje a gran escala (LLM). Brown et al. (2020), citados en Vaithilinga et al. (2022), revelaron que GPT-3, con 175 mil millones de parámetros, era capaz de resolver tareas de codificación sin necesidad de formación específica. Esta habilidad emergente ha consolidado en modelos posteriores como Codex, GitHub Copilot, y en familias más recientes como GPT-4, Claude, Gemini y Llama, donde se despliegan ventanas de contexto cada vez más amplias y un razonamiento multifase.

Asimismo, Chen et al. (2021) reportaron que Codex, ajustado a partir de GPT-3 con 159 GB de código en Python procedente de GitHub, alcanzó una precisión funcional del 28,8% en la primera muestra y superior al 70% al permitir 100 intentos sobre el conjunto de evaluación HumanEval. Esta prueba evidencia que no solamente reproducen partes memorizadas, sino que también generalizan estructuras algorítmicas. De esta manera, consiguen trasladar conocimiento de un lenguaje de programación a otro y de un tema a otro. Esto da muy buenas oportunidades para el multilinguaje pedagógico.

Complementariamente, Hou et al. (2024) encontraron que los modelos son usados en al menos cincuenta y cinco tareas distintas, las cuales pueden agruparse en cuatro categorías: comprensión de código, generación de código, mantenimiento y pruebas. Los autores subrayan que la generación es la categoría más reportada, lo que evidencia su centralidad para el aprendizaje, ya que es justamente la habilidad que más cuesta desarrollar a los principiantes y la que más motiva su inclusión en los currículos.

Por otro lado, Yue (2025), demostraron que la simple inclusión del prompt "piensa paso a paso" en una consulta a un LLM duplica el desempeño del modelo en tareas de razonamiento matemático y lógico. Este descubrimiento tiene importantes implicaciones pedagógicas: demostraría que los LLM son capaces de modelar e incluso de emular un pensamiento computacional, si así se les solicita. Esta característica hace que los LLM sean compañeros privilegiados para enseñar el proceso de razonamiento que se encuentra detrás del código, no sólo el resultado final.

Lo planteado muestra que los LLM han pasado de ser una curiosidad académica a formar parte del ecosistema de desarrollo. Esto en ámbito educativo, implica que alumnos trabajan con sistemas que entienden semántica del código y pueden actuar como pares programadores, modificando radicalmente las prácticas clásicas de aula. La SLR de esta investigación toma en cuenta esta evolución para clasificar las herramientas analizadas según

el tipo de modelo subyacente y la diversidad de tareas que dichos modelos pueden cubrir, desde la generación de funciones hasta la documentación automática de proyectos.

2.4 Otras herramientas de IAG en la programación

Más allá de ChatGPT y Copilot, el ecosistema incluye herramientas como Tabnine, Amazon CodeWhisperer, Replit Ghostwriter, Cursor, Codeium y Claude. Liang et al. (2024) realizaron un estudio cualitativo con 410 desarrolladores y encontraron que más del 80%

utiliza al menos un asistente basado en IAG, y que la elección depende del lenguaje de programación, el costo y la integración con el editor preferido. La diversidad de opciones también responde a preocupaciones específicas sobre privacidad: algunos equipos prefieren modelos locales como Tabnine para evitar enviar código propietario a servidores externos,

Por su parte, Barke et al. (2023) clasificaron los modos de interacción con asistentes IAG en dos: modo aceleración, donde el desarrollador conoce la solución y usa la IAG para escribirla más rápido, y modo exploración, donde la IAG ayuda a descubrir soluciones desconocidas. Esta dualidad tiene implicaciones pedagógicas directas: el modo aceleración favorece a expertos, mientras que el modo exploración puede beneficiar a novatos siempre que se acompañe de reflexión crítica posterior.

Adicionalmente, Khojah et al. (2024) entrevistaron a veinticuatro desarrolladores profesionales sobre su uso de asistentes de IAG y encontraron que las herramientas se utilizan principalmente para tres tareas: redacción de código rutinario, búsqueda de información sin abandonar el editor y exploración de tecnologías nuevas. Se debe tener en cuenta que la formación del asistente debe ser complementaria y nunca sustituta de lo humanos, o bien abre cuestionamientos relevantes a su implementación en el universo.

2.5 Herramientas que utilizan IAG en el entorno de programación

Existen diversas herramientas que utilizan IAG para mejorar el proceso de aprendizaje y desarrollo en programación. Algunas de las más destacadas son (Reasco , 2025):

GitHub Copilot: Desarrollado por OpenAI en colaboración con GitHub, Copilot es un asistente de programación en tiempo real. Funciona tomando tu código y generando automáticamente líneas de complemento mientras escribes. Copilot utiliza el modelo Codex de OpenAI, un modelo de inteligencia artificial generativo, para ofrecer fragmentos de código sugeridos, completar funciones completas basadas en los parámetros que se ha pasado y proponer cientos de otras soluciones en función del contexto que se establece.

A diferencia de otras herramientas de apoyo, copilot no solo terminará el código basándose en lo que escribes, sino que se adapta a cada usuario ofreciendo soluciones intuitivas para tareas complejas. Si eres un experto podrá descubrir múltiples soluciones con mayor rapidez.

OpenAI Codex: El propósito primordial de esta herramienta es que busca crear código a partir de descripciones en lenguaje natural. Codex ha avanzado intensamente al nivel que puede generar y comprender código escrito en diferentes lenguajes de programación. Para las personas que están aprendiendo programación, el Codex es una poderosa herramienta ya que puede tomar instrucciones en cualquier idioma, incluido el inglés, y produce código. Es una herramienta excelente no solo porque acelera el trabajo de programación, sino también porque sirve como recurso para aprender a implementar otro lenguaje de programación o cuando se enfrentan a un dilema específico.

Tabnine: Otro complemento de autocompletado de IA que permite a los programadores escribir código con mayor rapidez y precisión. TabNine emplea una IA que

opera en base de aprendizaje automático para proveer sugerencias de código cuando el usuario está escribiendo. TabNine soporta varios editores de texto y entornos de desarrollo integrado, y se adapta a los estilos de codificación y preferencias de los usuarios para facilitar su uso diario.

Replit: Se trata de una plataforma en línea que ayuda a los usuarios a redactar, poner en práctica y cooperar código. Usando IAG, replit contribuye a mejorar y a corregir errores en el código. Además, facilita a los alumnos el aprendizaje de la programación por que se trata de un ambiente interactivo de desarrollo donde tienen la posibilidad de duplicar y poner en funcionamiento ejemplos producidos por la IA para logra una mejor comprensión.

Kite: Es un asistente de codificación que utiliza inteligencia artificial. Se integra con varios editores de texto tipo: VSCode, Sublime Text o Atom y, utilizando IA, rellena automáticamente y da sugerencias de código. Puede examinar su código y dar información útil, el asistente incluso puede buscar repositorios y documentación de API, lo que le facilita el desarrollo de software.

Más allá de ser herramientas que mejoran la productividad y la eficiencia de los programadores, también cumplen un rol importante en la educación y en el aprendizaje de la programación: los estudiantes pueden aprender de forma activa mientras interactúan con ejemplos y soluciones generadas por IA (Reasco , 2025).

2.5 ChatGPT en la enseñanza de la programación

ChatGPT se ha consolidado como la herramienta de IAG más utilizada en contextos educativos, según Lo et al. (2024) realizaron una revisión sistemática de 50 estudios publicados entre 2022 y 2024 y concluyeron que ChatGPT mejora significativamente la comprensión conceptual, la depuración y la productividad del estudiante, aunque los efectos varían según el nivel educativo y la metodología pedagógica empleada. La revisión también detectó que los estudios sobre estudiantes de pregrado son los más abundantes, mientras que la educación secundaria y el bachillerato técnico permanecen subexplorados.

Por otro lado, Sun et al. (2024) condujeron un estudio con 105 estudiantes universitarios y demostraron que ChatGPT, cuando se usa como tutor socrático mediante prompts orientadores, fomenta el pensamiento crítico y evita la mera entrega de soluciones. La manera en que el alumno use a ChatGPT, sobre todo si lo utiliza como “oráculo” o “socio dialógico” determinará la efectividad pedagógica de esta herramienta en los procesos de enseñanza-aprendizaje. Si lo utiliza como oráculo, está consumiendo respuestas y, si lo utiliza como socio dialógico, está construyendo conocimiento a través de la negociación con el instrumento.

Además, Hellas et al. (2023) revisó la calidad de las respuestas que ofrece GPT-4 a las dudas conceptuales de los estudiantes principiantes. Verificaron que el modelo detecta el 78% de los fallos conceptuales, pero ofrece sugerencias valiosas en un 65% de los casos. Los investigadores señalan que los índices restantes abarcan explicaciones teóricamente incorrectas que podría generar ideas equivocadas en los estudiantes si no son detectadas por un docente.

La recopilación de datos reafirma la disparidad del impacto de ChatGPT como en el diseño del prompt, gestión pedagógica y la capacidad crítica de los estudiantes. Para esta SLR, es fundamental categorizar los estudios según el rol pedagógico que ChatGPT asume para producir conclusiones matizadas que reflejen la heterogeneidad de las prácticas reportadas en la literatura.

2.6 GitHub copilot y la programación asistida

GitHub Copilot, basado en el modelo Codex, integra sugerencias de código en tiempo real dentro del entorno de desarrollo. Vaithilingam et al. (2022) realizaron un estudio con 24 desarrolladores y observaron que, si bien Copilot no aceleraba significativamente el tiempo total de codificación, sí servía como punto de partida útil que reducía la fricción inicial y la búsqueda en documentación externa. Esta función de "reductor de fricción" puede ser tan importante como la aceleración bruta del trabajo, especialmente para principiantes que se bloquean ante la pantalla en blanco.

Adicionalmente, Puryear y Sprint (2022) analizaron el comportamiento de Copilot frente a 166 problemas de cursos introductorios de ciencias de la computación y encontraron que la herramienta resolvía correctamente alrededor del 50% de los ejercicios, lo que plantea retos serios para los educadores en cuanto al diseño de evaluaciones y la prevención del plagio. Según los autores, resulta recomendable el diseño de problemas contextualizados, abiertos y con artefactos no estandarizados inversamente proporcional a la efectividad que tenga el asistente y más comprometido al estudiante cognitiva y simuladamente.

Bird et al. (2023) realizaron una investigación interna en Microsoft sobre más de dos mil desarrolladores profesionales. Observa que Copilot permitió aumentar la velocidad de codificación en 55%, la tasa de tareas no finalizadas fue menor y la satisfacción laboral fue mayor. Para que esto sea cierto, el contexto en el que se encuentra la tecnología debe ser apropiado. En el caso de un trabajo, no es una tarea difícil.

Los aportes demuestran que Copilot transforma el trabajo del programador en el de revisor crítico del código que genera. Implica enseñar a leer y valorar código antes que programar, una inversión metodológica que debe registrar la SLR como tendencia emergente en contextos educativos. Asimismo, sugiere que las habilidades de los profesionales y los estudiantes divergen: los expertos se benefician más, mientras que los novatos requieren acompañamiento adicional para no incorporar prácticas erróneas.

2.7 Evolución histórica del aprendizaje de la programación

La enseñanza de la programación ha atravesado distintas etapas: desde las tarjetas perforadas y los lenguajes ensambladores en los años cincuenta, hasta los entornos visuales como Scratch y, recientemente, los asistentes de IAG. Según Becker et al. (2023), durante décadas el aprendizaje se centró en la memorización sintáctica y la depuración manual, lo que generaba altas tasas de abandono en las asignaturas introductorias de programación (CS1) y reforzaba la imagen de la programación como disciplina inaccesible para públicos no técnicos. Esta percepción contribuyó a la baja diversidad demográfica del campo durante décadas y al perfil masculino y anglófono dominante.

Por el otro lado, Prather et al. (2023) alegan que la irrupción de chatbots educativos en 2022-2023 constituyó un cambio de paradigma similar al que produjo la llegada de los entornos integrados de desarrollo en la década de 1990. Este cambio obligará a los docentes a replantear objetivos de asignaturas, como que el pensamiento computacional sea más importante que escribir código, y a desplazar la evaluación hacia tareas de mayor rango cognitivo como la crítica, la integración o el diseño arquitectónico.

Denny et al. (2023) también traza una genealogía de las prácticas pedagógicas, comenzando con el paradigma estructurado de los setenta, pasando por la programación de objetos del siglo noventa a la programación basada en bloques visuales del siglo XXI. Todos los cambios de métodos se produjeron de una ruptura con la barrera cognitiva inicial. La inteligencia artificial generativa es, la evolución más radical en este sentido: por primera vez, el alumno puede contar con un mediador conversacional que opera en lenguaje natural y traduce intenciones humanas en código ejecutable.

Las diferentes propuestas y reflexiones en torno a la enseñanza de la programación comenzaron a agruparse en un marco de referencia relativamente reciente, gracias a la influencia de autores como Denny et al. (2024), quienes han propuesto una clasificación por épocas. Esta cuarta era en la que se inscribe la IAG, se distingue por el hecho de que el esfuerzo cognitivo del estudiante se desplaza de la producción manual de código a la dirección estratégica de herramientas que generan ese mismo código, redefiniendo así muy profundamente la noción de «saber programar».

Los aportes de Lon coinciden en describir a un campo en permanente transformación, lo que podría decirse del punto de partida y del resultado del presente estudio, pero para lo que aquí se presenta esa perspectiva histórica tiene sentido porque permite leer que la IAG no aparece en un vacío pedagógico, sino que viene a ser una más dentro de una larga serie de perturbaciones tecnológicas en constante inclusión y exclusión que ha visto a la educación argentina en los últimos años. La SLR debe tener en cuenta que los estudios analizados pertenecen a una bisagra entre prácticas consolidadas y prácticas emergentes. Por lo que se deben tomar con precaución los hallazgos cuyos resultados podrían caducar con la incorporación de una nueva generación de modelos o con la modificación de las políticas institucionales sobre su uso IAG.

2.8 Retos cognitivos del aprendizaje de la programación

Aprender a programar requiere una articulación compleja de habilidades cognitivas: abstracción, descomposición de problemas, razonamiento lógico, depuración y comprensión de notaciones formales. Denny et al. (2024) identifican que entre el 30% y el 40% de los estudiantes de cursos introductorios reprueban o abandonan, principalmente por la dificultad para construir modelos mentales precisos del comportamiento del programa. La imposibilidad de ver, o imaginar, lo que ocurre dentro del computador al ejecutar instrucciones se ha conocido en la literatura como el problema del nocional machine.

Por otra parte, Margulieux et al. (2024) indican que la sobrecarga de la memoria de trabajo, la dificultad para transferir ideas entre ejercicios y la frustración debida a los mensajes de error poco claros del compilador constituyen los principales obstáculos

cognitivos. La motivación y autoeficacia estudiantil se ven afectadas en función de estas barreras y, sobre todo, de los estudiantes con baja experiencia previa en pensamiento lógico-matemático. Según las autoras, la sensación de fracaso que se viven de forma reiterada lleva a muchos a abandonar la carrera en su primer semestre.

Según Loksa et al. en el 2022, las habilidades metacognitivas son tan importantes como las técnicas. Los alumnos que no son capaces de autorregular su metacognición entran en circuitos erróneos de ensayo y error. Por ello, aprenden más lentamente y se rinden más fácil. Los autores proponen un entrenamiento metacognitivo explícito en cursos introductorios.

De acuerdo con un artículo de 2024 se fueron a identificar 5 áreas en los que habitualmente los estudiantes fracasan cuando aprenden a programar. Esto que se cobran por el sistema general (saber que hay que saber con un programan), máquina atónita (saber cómo sigue ordenan), notado (saber cómo se sufre la sintaxis), estructuras (control de patrones y datos), y pragmática (estilo de depuración y diseño). Esta clasificación clásica todavía tiene validez en la era de la IAG. Además, puede servir de mapa diagnóstico que orienta a las intervenciones que aprovechan a los asistentes de IA en los dominios donde tiene mayor dificultad cada estudiante.

La convergencia indica que la enseñanza de programación requiere un fuerte acompañamiento cognitivo y metacognitivo. La IA generativa puede ser la mediadora entre usuario y tecnología al traducir un mensaje de error a lenguaje natural, descomponer el problema y reducir la carga cognitiva extrínseca. Sin embargo, este mismo soporte puede, si se da incorrectamente, atrofiar la formación de un modelo mental robusto que demanda el ejercicio profesional, lo que justifica un cuidadoso análisis en SLR. Una intervención adecuadamente diseñada deberá, por un lado, anclar a la persona mediante andamiaje y, por otro lado, desafiarla en su producción, evitando sobrecarga, así como subdemanda.

2.9 La IA generativa como mediadora del aprendizaje

Desde el enfoque socio-constructivista de Vygotsky, todo aprendizaje complejo requiere un mediador que opere en la zona de desarrollo próximo (ZDP), espacio donde el aprendiz puede lograr con apoyo lo que aún no puede hacer solo. Kasneci et al. (2023) Argumentan que los chatbots basados en LLM facilitan la educación al ofrecer explicaciones contextualizadas, ejemplos personalizados y retroalimentación inmediata, multiplicando las oportunidades para la práctica deliberada. Esta mediación es escalable y no depende de la disponibilidad horaria del profesor, lo que democratiza el acceso a la tutoría.

Además, Yilmaz y Karaoglan (2023) llevaron a cabo un estudio cuasi-experimental con 45 estudiantes universitarios y revelaron que el grupo que usó ChatGPT mostró mejoras en motivación, autoeficacia computacional y pensamiento reflexivo con diferencias significativas ($p < 0,05$) con respecto al grupo control. Los autores creen que se deben a la disponibilidad permanente del recurso, a que permite adaptarse a su propio ritmo y a que lo hace a un ritmo que reduce la ansiedad de tener que plantear dudas ante pares o profesores.

Al utilizar el modelo UTAUT2 aplicado a estudiantes universitarios en Polonia, Strzelecki (2023) identificó que la expectativa de desempeño y el hábito son los predictores más fuertes de la intención de usar ChatGPT en contextos académicos. Se utiliza la teoría de prácticas para mostrar que las características del sistema de IAG no son suficientes. En otras palabras, el uso sistemático en el aula también es importante. Por lo tanto, es importante la política institucional que favorezca la habitualidad ética.

Kasneci et al. (2023) Proponer siete prácticas pedagógicas en las que el IAG puede actuar como mediador; a saber, el IAG puede actuar como tutor, como formador, como mentor, como compañero de equipo, como herramienta de simulación, como estudiante al que el alumno “enseña”, como evaluador

. Esta tipología es interesante porque nos ayuda a visibilizar que la mediación de la IA generativa no es sólo la de “oráculo que responde”, sino que admite configuraciones pedagógicas múltiples cuya toma de decisiones dependerá de los objetivos formativos que estemos persiguiendo en cada momento.

Los autores reflejan que la IAG puede funcionar como un tutor virtual que está, en esencia, permanentemente disponible. De esta forma, redistribuye el rol del docente hacia aquellas tareas que tienen un valor pedagógico más alto, como la selección, la evaluación auténtica o el acompañamiento socioemocional. No obstante, esto requiere alfabetización digital tanto del estudiante como del docente para aprovechar el potencial mediador sin caer en la dependencia, dimensión que esta SLR examina con detalle a través de las preguntas de investigación referidas a beneficios y riesgos.

2.10 Beneficios pedagógicos del uso de la IAG en programación

Diversos estudios han documentado beneficios concretos. Mellado et al. (2024), en una investigación cuasi-experimental con 60 estudiantes chilenos de educación superior, hallaron que el uso de IAG generativa incrementó en un 23% el rendimiento académico promedio respecto al grupo control y redujo el tiempo de resolución de ejercicios en un 35%. Los investigadores indican que estos resultados se deben a que se debatió sobre la retroalimentación, ejemplos contextualizados y la iteración en diferentes soluciones en un corto tiempo.

Según Rahman y Watanobe (2023), la IAG tiene tres aportes que se manifiestan en la personalización del aprendizaje, la retroalimentación inmediata y la democratización del acceso al conocimiento técnico. Resaltan que aquellos alumnos que no tienen mucha cultura previa en tecnología son los que más provecho sacan, dado que la IAG actúa como un “profesor particular” gratuito accesible desde el móvil y sin grandes costos económicos en sus versiones gratis.

Leinonen et al. (2023), en un estudio con 86 estudiantes, los que recibieron explicaciones de ayudas generadas por GPT-4, lograron resolver los errores un 50% más rápido a diferencia de los que utilizaron la salida estándar del compilador, lo que reduce la tensión por depuración.

Los datos, la sistematización y la evidencia experimental muestran que los beneficios son reales y medibles, no sólo discursivos. Para esta revisión de la literatura, la identificación de estos resultados beneficiarios les da respuestas a estas preguntas de investigación que se refieren a positivos impactos, Sin embargo, conviene matizar que los efectos de la IAG dependen del diseño didáctico y no de la simple accesibilidad a la herramienta: la IAG potencia procesos de aprendizaje bien diseñados, pero no compensa diseños deficientes.

Beneficios identificados en la literatura reciente:

- Personalización del proceso formativo de acuerdo con el ritmo del estudiante.
- Retroalimentación inmediata que reduce el tiempo de resolución de errores.
- Generación automática de ejemplos prácticos y casos de estudio.
- Mejora de la motivación intrínseca y la autoeficacia computacional.
- Acceso democratizado a tutoría especializada sin restricciones de horario.
- Apoyo a estudiantes con discapacidad o necesidades educativas específicas.
- Traducción de mensajes de error técnicos a explicaciones en lenguaje natural.
- Reducción de la ansiedad ante la exposición de dudas frente a docentes y pares.
- Generación de variantes de problemas para la práctica deliberada.
- Aceleración del prototipado y la experimentación con soluciones alternativas.

2.11 Riesgos y desafíos del uso de IAG en educación

La adopción acrítica de la IAG conlleva riesgos significativos, según Cotton y colegas (2024) señalan que el uso indebido de ChatGPT facilita formas sofisticadas de plagio y obstaculiza la evaluación genuina del aprendizaje, lo que ha motivado a instituciones a prohibir su empleo o rediseñar sus modalidades evaluativas. Los autores rechazan las soluciones simplemente prohibicionistas porque las consideran ineficaces. En vez de eso, proponen una respuesta estructural mediante el rediseño del currículo y la evaluación.

Garzón y sus colaboradores (2023) analizan dos riesgos adicionales que deben ser considerados: que el alumnado pueda adquirir una dependencia excesiva hacia estas herramientas que afecte habilidades cognitivas básicas y que los sesgos que estén en los datos de entrenamiento siempre se reflejan en las respuestas de la IA. Cuando el modelo “alucina” es capaz de generar contenido erróneo pero que no carece de sentido. En este caso el riesgo se amplía y de manera importante en los alumnos más principiantes que no tienen criterios. En el ámbito de la programación este problema deja de ser trivial con un código que parece correcto puede haber defectos semánticos invisibles.

Bender, Gebru, etc. (2021) definen a los LLM como un “loro estocástico” ya que producen un texto que puede llegar a ser muy coherente y adecuado, con escasa o nula comprensión semántica. Esta caracterización está diseñada para mostrar que existe un riesgo social real; y que esos modelos refuerzan los prejuicios, diseminan la desinformación, y tienen un coste ambiental considerable, que es el coste del entrenamiento de esos modelos de gran tamaño. Se establece que diversos autores establecen sus límites a la inteligencia artificial, advirtiéndole sobre sus costos. Se exhorta a la academia a verificar lo que cuesta incorporar, en forma masiva, inteligencia artificial generativa en lo educativo.

Estos estudios constituyen un registro de riesgos que toda revisión sistemática debe documentar. Esta investigación reconoce que estos riesgos coexisten con beneficios, así que la cuestión es qué tipo de marcos institucionales regulan ese uso mientras se genera en los estudiantes capacidad crítica para detectar errores, inconsistencias y sesgos en lo que producen esos sistemas.

Tabla 2

Principales riesgos del uso de la IAG en el aprendizaje de programación

Riesgo	Descripción
Plagio académico	Estudiantes que entregan código generado por IA sin comprenderlo.
Dependencia cognitiva	Pérdida de autonomía para resolver problemas sin asistencia.
Alucinaciones	Generación de soluciones incorrectas con apariencia de validez.
Sesgos algorítmicos	Reproducción de prejuicios presentes en los datos de entrenamiento.
Privacidad de datos	Exposición de información personal al subir consultas o código.
Costo ambiental	Alto consumo energético derivado del entrenamiento e inferencia.
Desinformación técnica	Difusión de prácticas de programación incorrectas o inseguras.

Nota. Riesgos que pueden afectar habilidades y ética del estudiante.

2.12 Aprendizaje personalizado e IAG

El aprendizaje personalizado consiste en adaptar contenido, ritmo y estrategias didácticas al perfil de cada estudiante. Zhang et al. (2024) sostienen que la IAG opera como un sistema tutor inteligente generativo que, a diferencia de los sistemas tutores clásicos basados en reglas, no requiere modelos del dominio precableados, sino que infiere las necesidades del aprendiz desde la conversación misma. Esto reduce drásticamente los costos de desarrollo de los sistemas tutores y los hace accesibles a instituciones con menos recursos.

Bhutoria (2022) realizó un estudio de revisión de la literatura y concluyó que los sistemas de IA generativa afectan el aprendizaje en tres aspectos con más eficacia que la plataforma tradicional y logran rendimientos de aprendizaje 30% más altos en cursos STEM. La autora indica que esta efectividad depende de añadir el enfoque human-in-the-loop, esto es, que la monitorización docente sobre las recomendaciones automáticas del sistema se emplee como condición necesaria para que sus efectos sean sostenibles.

Por su lado, Chiu et al. (2023) analizaron 92 estudios sobre IA en educación entre 2012 y 2022 y concluyeron que la personalización figura como uno de los cuatro grandes roles de la IA en el aula, junto con la evaluación, la predicción del desempeño y la asistencia administrativa. La IA generativa amplifica este rol al reunir esas funciones en una sola interfaz conversacional, lo que reduce las barreras para la adopción institucional.

Estas aportaciones belicistas demuestran que se puede personalizar eficazmente con una herramienta de IA generativa bien orientada y sin requerir una infraestructura compleja ni equipos de ingeniería pedagógica. No obstante, el hecho técnico no exime de la responsabilidad pedagógica de diseñar trayectorias formativas con sentido en la SLR, sino que problematiza dicho diseño al analizar las metodologías pedagógicas asociadas. Lo que se puede entender como personalización puede terminar siendo aislamiento y espaciado del aprendizaje colectivo.

2.13 Metodologías pedagógicas que integran IAG

La integración efectiva de la IAG requiere marcos metodológicos coherentes, Pradana et al. (2023) proponen el modelo Flipped Classroom potenciado con IAG, en el que los estudiantes interactúan con chatbots fuera del aula para preparar contenidos y reservan el tiempo presencial para la discusión y la resolución colaborativa de problemas. Los autores reportan incrementos del 40% en participación activa, lo que sugiere que la IAG libera tiempo de clase para actividades de mayor valor pedagógico.

Essel et al. (2024) utilizan un enfoque de Aprendizaje Basado en Problemas –ABP– con apoyo de ChatGPT y obtienen mejoras estadísticamente significativas en pensamiento crítico ($d = 0,72$) y en autonomía del aprendizaje, siendo especialmente notoria la mejora en estudiantes de bajo nivel socioeconómico sin tutoría externa. Según el estudio, la IAG puede ser un nivelador de oportunidades educativas, cerrando brechas que el aula no puede cerrar.

Los autores anteriormente mencionados también proponen a su vez una metodología de Aprendizaje Basado en Proyectos asistido por IAG en la que los estudiantes diseñan productos software completos con el apoyo del asistente Jin et al. 2023. Los resultados muestran que este tipo de metodología aumenta la motivación intrínseca y la transferencia a los espacios de trabajo, en comparación con cursos de ejercicios cerrados.

Los tres trabajos coinciden que la IAG potencia las metodologías activas pero no las reemplaza. Este hallazgo es central para la revisión sistemática de literatura: la efectividad de la IAG depende del andamiaje metodológico que la rodea, de modo que una clase tradicional con IAG puede producir efectos limitados, mientras que una clase activa con IAG multiplica el impacto formativo de forma sustancial.

Tabla 3

Metodologías pedagógicas combinadas con IAG

Metodología			Aporte de la IAG
Aula Invertida			Tutor previo a la sesión presencial.
Aprendizaje Basado en Problemas	Basado	en	Generación de casos contextualizados.
Aprendizaje Colaborativo			Mediación entre pares para depuración conjunta.

Aprendizaje por Proyectos	Generación de prototipos rápidos y prueba de hipótesis.
Pair Programming con IA	La IAG actúa como segundo programador ("navegador").
Gamificación	Generación de retos y narrativas adaptativas.
Microaprendizaje	Píldoras formativas adaptadas a la consulta puntual.

Nota. Resumen de metodologías activas integradas con herramientas IAG.

2.14 Ética e integridad académica con IAG

La ética del uso de IAG en educación abarca temas como autoría, atribución, equidad, privacidad y transparencia. Eaton (2023) propone el concepto de "integridad académica postplagio", reconociendo que las definiciones tradicionales de plagio resultan insuficientes ante una herramienta que coautoriza textos. La autora defiende marcos institucionales que regulen la coautoría humano-IA —en lugar de prohibirla— y exige declaraciones de uso obligatorias, similares a las declaraciones de conflicto de interés. La UNESCO (2023) ha publicado el documento ‘Guide to Generative AI in Education and Research’, el cual propone principios de centralidad de la persona, igualdad de oportunidades y comunicación clara. También prevé una edad mínima de uso de trece años con consentimiento parental y un entrenamiento académico para profesores y estudiantes en pensamiento crítico sobre IA. Este documento ha sido uno de los más citados en la reciente literatura educativa sobre este asunto. Chan (2023), aunque con un alcance más acotado a las universidades, propuso el "AI Ecological Education Policy Framework" —organizado en seis áreas: gobernanza, protección de datos, evaluación, alfabetización en IA, apoyo docente y apoyo estudiantil—, que varias universidades asiáticas y europeas han adoptado con resultados que la literatura describe como prometedores.

Las propuestas configuran un marco ético robusto que la SLR utiliza como referencia para evaluar la madurez ética de los estudios revisados, además un estudio que reporta uso de IAG sin discutir su dimensión ética debe ser considerado como evidencia parcial, lo que también justifica una pregunta de investigación específica sobre este punto. La gobernanza institucional emerge como factor crítico para que el uso de la IAG produzca efectos pedagógicos positivos sin comprometer la integridad del sistema educativo.

Principios éticos clave según la UNESCO (2023):

- Centralidad del ser humano y respeto a su agencia.
- Equidad e inclusión para reducir brechas digitales.
- Transparencia y explicabilidad de los sistemas algorítmicos.
- Protección de datos personales y consentimiento informado.
- Formación crítica del estudiantado y profesorado.
- Edad mínima recomendada de 13 años con autorización.
- Auditoría y rendición de cuentas sobre los sistemas usados.
- Sostenibilidad ambiental en la adopción tecnológica institucional.

2.15 Brecha digital y equidad en el acceso a la IAG

La adopción global de IAG no es uniforme, según Holmes y Tuomi (2022) advierten que la mayoría de los modelos comerciales se entrenan principalmente con datos en inglés y reflejan sesgos culturales del Norte global, lo que limita su utilidad en regiones de habla hispana, indígena o en lenguas minoritarias. Esta inequidad estructural amenaza con ampliar las brechas educativas existentes y consolidar dependencias tecnológicas asimétricas entre el Norte y el Sur global.

Además, Cela-Ranilla et al. (2017) hicieron un estudio en universidades latinoamericanas y encontraron que solamente el 47 por ciento de los estudiantes cuentan con un acceso estable a las versiones premium de herramientas de IAG, mientras que las gratuitas tienen importantes limitaciones en velocidad, calidad y disponibilidad. Esta desigualdad económica impacta el rendimiento escolar y origina una brecha intracurso entre quienes pueden costear suscripciones y los que no.

Según Selwyn (2024), no sólo existe una brecha de acceso en torno a la IAG, sino también de criterio. En este sentido, los estudiantes con un mayor capital cultural utilizan la IAG de forma más compleja y obtienen mejores resultados en sus trabajos escolares y académicos que los usuarios vulnerables. Estos últimos, al contrario, suelen hacer un uso más superficial y mimético de estas tecnologías. Esta brecha de uso amplifica las desigualdades preexistentes y exige políticas de alfabetización crítica explícita

Estos autores sitúan el debate de la IAG en el plano de la justicia educativa. Para esta SLR, realizada en el contexto ecuatoriano, este enfoque es crítico: las soluciones que funcionan en universidades de altos recursos pueden no replicarse en contextos del Sur global, lo que exige sensibilidad contextual al sintetizar la literatura. La inequidad opera en al menos tres niveles: acceso técnico, calidad lingüística y competencia de uso, dimensiones que las políticas educativas deben abordar de manera integral.

CAPÍTULO III.

3. METODOLOGIA.

3.1 Enfoque de la investigación

La Presente investigación tiene un enfoque cualitativo, ya que se centran en la interpretación y comprensión de los hallazgos obtenidos en la literatura. Creswell (2014) sostiene que la investigación cualitativa “Busca comprender los significados que los participantes otorga a un fenómeno y construir una interpretación integradora del mismo.

3.2 Alcance de la Investigación

Además, la investigación es de naturaleza descriptiva ya que tiene como objetivo identificar y categorizar herramientas, sus ventajas, limitaciones y tendencias en el uso de la inteligencia artificial generativa en la educación en programación. Hernández, Fernández y Baptista (2018) sostienen que la investigación descriptiva pretende “detallar cómo es y cómo se manifiesta un fenómeno” sin manipular variables.

3.3 Tipo de Investigación.

El presente estudio corresponde a una investigación de tipo documental, ya que se basa en la recolección, revisión y análisis de fuentes científicas ya publicadas (Fidias G. Arias, 2012).

3.4 Diseño de investigación

Esta investigación desarrolla una revisión sistemática de la literatura o SLR por sus siglas en inglés (Systematic Literature Review), la cual se utiliza para compilar, evaluar y sintetizar la totalidad de la evidencia científica relevante producida sobre un tema específico de manera rigurosa y objetiva. En este sentido, a diferencia de las revisiones tradicionales, las SLR precisan, en el protocolo, un conjunto predefinido de criterios de inclusión y exclusión. Asimismo, incorporan búsquedas integrales en varias bases de datos y evaluaciones de la calidad de los estudios abordados (Lopez y otros, 2024).

3.5. Metodología SLR

Este estudio se sostiene en la metodología de Kitchenham (2004) para realizar revisiones sistemáticas de la literatura, junto con las aportaciones de Okoli y Schabram (2010), siguiendo una secuencia meticulosa de pasos, cada etapa desarrollada permitió llevar a cabo la SLR. A continuación, desentrañaremos el protocolo empleado en la investigación, el cual puede ser replicado en diversas circunstancias y escenarios (Okoli, 2015). A continuación, en la Figura 1 se presentan los pasos para realizar una SLR.

Figura 1

Pasos para realizar una SLR (Okoli, 2015)



Nota. Esquema de la metodología de revisión sistemática de la literatura.

a) Propósito de la revisión de la literatura

Una de las claves para alcanzar el éxito en cualquier proyecto como, por ejemplo, una revisión de la literatura es poder definir claramente cuál es su propósito y objetivo principal. Es fundamental tener una visión clara y detallada de hacia dónde se quiere llegar y qué se quiere lograr con la realización de dicha tarea o proyecto. Esta definición precisa del propósito ayuda a mantener, estrictamente hablando, la importancia del mismo y requiere una cuidadosa consideración por parte de todos los involucrados (Okoli, 2015).

b) Protocolo y capacitación

Elaborar detalladamente el protocolo y brindar una completa capacitación al equipo de trabajo, ya que en el proceso participan múltiples revisores, será imprescindible. Para esto se debe contar con un protocolo exhaustivo, minucioso y consensuado que explique de manera detallada y precisa cada uno de los pasos a seguir.

Además, es fundamental que todo el equipo se someta a un proceso de formación y entrenamiento íntegro de dicho protocolo, con el objetivo de garantizar que cada miembro del equipo aplique de forma consistente y uniforme los criterios y procedimientos

establecidos, evitando así posibles discrepancias derivadas de interpretaciones individuales (Okoli, 2015).

c) Búsqueda de literatura

Esta fase se centra en la identificación y localización de las diversas fuentes de información relevantes para el estudio como: artículos de revistas, tesis, libros e informe científicos.

Este paso es crucial y debe llevarse a cabo siguiendo rigurosamente el protocolo preestablecido, lo cual garantizará la solidez y la fiabilidad de los resultados obtenidos en el estudio. Es importante destacar que este proceso, si bien puede resultar laborioso, es esencial para asegurar la validez y la coherencia de la investigación realizada (Okoli, 2015).

d) Pantalla de práctica

En esta fase los investigadores tienen acceso a una enorme cantidad de información adquirida, entonces se procede a filtrar los recursos cuyos contenidos se encuentren dentro de las temáticas establecidas para las interrogantes de la investigación. Para esto, solo revisar los resúmenes, las palabras claves, sin necesidad de proceder a una validación de calidad de los recursos, ya que esto se lleva a cabo en una fase posterior (Okoli, 2015).

e) Valoración de la Calidad

Una vez seleccionados todos los artículos potencialmente aptos para ser evaluados, los revisores deben llevar a cabo un examen más riguroso para determinar su calidad. No todos los estudios primarios presentan el mismo nivel de calidad. Por ello, en una revisión de la literatura es necesario evaluarlos según criterios establecidos, verificando en qué medida cumplen estándares metodológicos y de rigor científico (Okoli, 2015).

f) Extracción de los datos

La extracción de datos representa una fase esencial en el procedimiento de revisión sistemática. Durante esta etapa se extrae la información después de haber elegido solamente las fuentes que son apropiadas para la revisión sistemática final. Es recomendable, para ello, crear una matriz de extracción de datos que se ajuste a las preguntas de investigación y las características de los estudios seleccionados. Esta matriz es esencial para organizar la información y para la fase de síntesis (Okoli, 2015).

g) Síntesis de estudios

Una vez realizado el proceso de codificación y extracción de datos es esencial combinar los datos para obtener una perspectiva global de la investigación, mediante la categorización, comparación y contraste de la información. Sin embargo, el enfoque

analítico, tanto cualitativo como cuantitativo, se definirá estrictamente por la naturaleza metodológica de los estudios que se incluyan (Okoli, 2015).

h) Redacción de la revisión

El éxito de la redacción final esta intrínsecamente ligado a la solidez de las fases previas. Por ello, esta etapa no se debe subestimarse, pues una comunicación clara de los hallazgos garantizara la validez, la replicabilidad y el impacto de la investigación en la comunidad científica (Okoli, 2015).

CAPÍTULO IV.

4. RESULTADOS Y DISCUSIÓN

Siguiendo las etapas de la metodología Revisión Sistemática de la Literatura (SLR), a continuación, se presentan los resultados obtenidos en cada una de ellas:

4.1 Propósito de la revisión de la literatura

El propósito concreto de esta investigación fue: Conocer cómo la inteligencia artificial generativa se está utilizando para apoyar o mejorar el aprendizaje de la programación en contextos educativos universitarios.

4.2 Protocolo de revisión.

El proceso de revisión se desarrolló a partir de un protocolo metodológico que se estableció, en primer lugar, las preguntas de investigación, así como las estrategias de búsqueda y los criterios de inclusión, exclusión y evaluación de la calidad de los estudios. A continuación, se presenta la descripción detallada de cada uno de los elementos.

4.2.1 Preguntas de investigación

- a) ¿Qué tipos de herramientas de IAG se está utilizando para aprender programación?
- b) ¿Cuáles son los beneficios sobre el uso de la inteligencia artificial generativa en el aprendizaje de la programación?
- c) ¿Qué riesgos se identifican al utilizar inteligencia artificial generativa en los ámbitos educativos?
- d) ¿Qué metodologías pedagógicas se aplican en combinación con inteligencia artificial generativa para enseñar programación?

4.2.2 Estrategia de búsqueda

Se llevó a cabo una exhaustiva búsqueda en bases de datos científicas de renombre internacional para recopilar información relevante. La estrategia de búsqueda incluyó el meticuloso empleo de palabras clave específicas y operadores booleanos con el objetivo de asegurar la máxima relevancia y precisión en los estudios recuperados.

Palabras clave

- Generative Artificial Intelligence
- Programming Learning
- IA Generrativa
- IA-supported learning
- Generatie AI models for learning

Operadores booleanos

- OR y AND

En este sentido, en la Tabla 1 se muestran los nombres de las bases de datos.

Tabla 4

Búsqueda en bases de datos científicas

Fuente	Dirección web
Scopus	https://www.scopus.com/
Google Académico	https://scholar.google.com/
SciELO	https://scielo.org/

Nota. Bases de datos consultadas para recopilar literatura relevante.

Una vez realizadas las búsquedas, se seleccionaron únicamente los estudios que cumplieran con los siguientes criterios de inclusión establecidos:

- Publicaciones comprendidas entre los años del 2020 al 2025
- Documentos disponibles con el idioma español-inglés-portugués
- Enfocados únicamente al aprendizaje de la programación con IAG

Se eliminaron los estudios que cumplieran con los siguientes criterios de exclusión establecidos:

- Publicaciones antes del 2020
- Temas que no estén relacionado con la IAG
- Cuando no hagan referencia a contextos universitarios
- Que sean revisiones de la literatura
- No tratan sobre el aprendizaje de la programación con IAG
- Publicaciones repetidas

4.3 La búsqueda de la literatura

Al describir cuántos artículos se encontraron en cada base de datos es importante mencionar el número total de artículos recuperados en cada una de ellas. Esto permitió tener una visión más amplia y detallada de la cantidad de información disponible en cada base de datos. En este sentido en la Figura 2, se muestran los totales obtenidos en cada base de datos.

Figura 2

Resultados de la búsqueda de la literatura en bases de datos

SCOPUS	Operador logico	Palabra Clave	Total
Generative Artificial Intelligence	AND	Programming Learning	16
Artificial Intelligence	AND	Programming Learning	20
GOOGLE ACADEMICO	Operador logico	Palabra Clave	Total
IA Generrativa	and	Programming Learning	20
Programming Learning	OR	Generative Artificial Intelligence	40
Scielo	Operador logico	Palabra Clave	Total
Generative Artificial Intelligence	OR	Learning programming	20

4.4 Pantalla de práctica

Ante el volumen masivo de información recopilada, se aplicó un proceso de cribado inicial, este filtrado se basó en la concordancia del contenido con las preguntas de investigación predefinidas, realizando una revisión selectiva de resumen y palabras clave, aplicando de igual forma los criterios de inclusión y exclusión previamente definidos, cabe destacar que en esta instancia el enfoque se centró exclusivamente en la relevancia temática, postergando la evaluación crítica de la calidad técnica de los recursos para su fase siguiente.

Un ejemplo de la pantalla práctica se presenta en la Figura 3.

Figura 3

Pantalla de práctica del proceso de selección de la literatura.

1	2023	Mechanisms of enhancing learning with unequal preparation: An experimental study on generative artificial intelligence use and proficiency in programming learning (Mecanismos para mejorar el aprendizaje con una preparación desigual: Un estudio experimental sobre el uso y la competencia de la inteligencia artificial generativa en el aprendizaje de programación)	Shi, Yuesi, et al.; Wei, Shen, Junjie, Hu, Bingyan	Artículo	Computadoras y educación: Inteligencia artificial	USA	La Tech University	Cuantitativo	Explicativo	De Campo	Diseño Experimental	Identificar los efectos del uso de herramientas GenAI en los resultados de aprendizaje a través de su impacto en el beneficio informativo percibido, el costo del aprendizaje y la adecuación del conocimiento.	En el ámbito educativo general, los hallazgos subrayan que los docentes deben enfatizar el uso adecuado de GenAI, adaptar las orientaciones según el nivel de preparación de los estudiantes y brindar apoyo específico a los más desfavorecidos, combinando el uso de GenAI para aliviar cargas cognitivas excesivas pero promoviendo al mismo tiempo el esfuerzo y el compromiso, con el fin de evitar una dependencia excesiva que termine perjudicando la eficacia del aprendizaje.	He et al., 2023	https://www.sciencedirect.com/journal/Computational Education
2	2024	Artificial Intelligence Supporting Independent Student Learning: An Evaluative Case Study of ChatGPT and Learning to Code (Inteligencia artificial que apoya el aprendizaje independiente de los estudiantes: un estudio de caso evaluativo de ChatGPT y Learning to Code)	Randall Hartley, ORCID, Minseok Hwang, Unhyuk Ko	Artículo	Education Sciences (EDS)	Estados Unidos	University of Nevada, Las Vegas, San-Diego University	Cuantitativo y cualitativo (estudio evaluativo con análisis mixto)	Estudio de caso evaluativo	Investigación aplicada / evaluativa	Diseño de estudio de caso evaluativo	Este estudio investigó el potencial de ChatGPT para apoyar el aprendizaje autorregulado y el aprendizaje de un lenguaje de programación.	Este estudio analizó el potencial de ChatGPT para apoyar el aprendizaje autorregulado y la programación en Python en contextos de materiales didácticos, herramientas de contenido, planificación y evaluación. Los resultados muestran que ChatGPT es más efectivo para proporcionar recursos educativos y ayudar a clarificar el aprendizaje independiente, así como para dar retroalimentación inmediata y orientación específica en programación. Sin embargo, presenta limitaciones importantes para realizar evaluaciones formativas, ofrecer experiencias prácticas e interactivas comparables a un IDE, e individualizar de forma consistente la enseñanza. Su utilidad depende en gran medida de las habilidades de autorregulación y metacognición del estudiante, pues el sistema apoya principalmente a quienes formulan buenas preguntas y refinan sus indicaciones. Aunque la IA puede facilitar la planificación y el acceso a información, sigue siendo poco adecuada para aspectos como la motivación, el monitoreo profundo de la comprensión y la personalización de la enseñanza.	Hartley et al., 2024	https://www.mdpi.com/2222-0272/14/2/320
3	2024	The Widespread Gap: The Benefits and Harms of Generative AI for Novice Programmers (La brecha cada vez mayor: beneficios y perjuicios de la IA generativa para programadores novatos)	James Prather, Brent H. Reeves, John Lomonax, Stephen Mitchell, Ariana S. Rinaldesando, Brent A. Becker, Bailey Kinneel, Janet Wright, Ben Briggs	Artículo de investigación	ICER 2024 (Proceedings)	Información al profesor en EE. UU., Finlandia, Irlanda	Albion Christian University, Dublin University, Temple University, College Dublin	Mixto	Experimento / observacional	Investigación aplicada / empírica	Comparativo experimental / observacional entre grupos	Analizar beneficios y perjuicios del uso de la IA generativa (Generative AI) para programadores novatos.	El estudio muestra que la IA generativa puede facilitar la resolución de problemas y acelerar la producción de soluciones entre programadores novatos, proporcionando ejemplos, fragmentos de código y sugerencias que aumentan la productividad y permiten avanzar más rápido en tareas prácticas; sin embargo, las observaciones comparativas entre grupos revelan efectos adversos importantes, como dificultades metacognitivas (insights que no detectan errores conceptuales), una falsa sensación de progreso que oculta errores persistentes y una explotación de la brecha entre quienes saben evaluar críticamente las salidas de la IA y quienes no.	Prather et al., 2024	https://doi.org/10.1145/9743282.9743446
4	2025	Generative AI Assisting Programming Learning: Functional Roles, Comparative Analysis, and Emerging Challenges (IA generativa que asiste en el aprendizaje de la programación: Roles funcionales, análisis comparativo y desafíos emergentes)	Hélida Aczo Tivez, Isidre, Mariana Rago Marión, Patricia Jaques	Artículo de conferencia	XIV Congreso Brasileiro de Informática em Educação (CIBIE 2025)	Brasil	Federal University of Paraná - Curitiba - UFPR - Brasil, Zufferean University Center of Palmas (UEZUPA) - TO - Brasil	Mixto	Exploratorio y comparativo	Investigación aplicada / evaluativa	Estudio de caso evaluativo por resolución de datos cualitativos y cuantitativos	Identificar roles funcionales, comparar herramientas y evaluar desafíos.	El estudio muestra que la IA generativa respalda aspectos concretos del aprendizaje de programación: asiste la resolución de problemas, ofrece retroalimentación inmediata y facilita la generación de ejemplos y fragmentos de código adaptados al nivel del estudiante. Sin embargo, los resultados también evidencian variabilidad en la calidad de las respuestas (errores, omisiones y sesgo) y un riesgo medible de dependencia por parte del alumnado cuando la herramienta se usa sin mediación. En los debates comparativos, las herramientas funcionan mejor en tareas de apoyo contextualizado, sugerencias y correcciones puntuales que en evaluaciones de comprensión profunda, y la efectividad aumenta notablemente cuando se combinan con actividades que exigen verificación, pruebas y justificación por parte del estudiante.	Hélida Aczo Tivez, Isidre, Mariana Rago Marión, Patricia Jaques	https://doi.org/10.1145/9743282.9743446

Nota. Filtrado de artículos según relevancia para la investigación.

4.5 Procedimiento de extracción

Para la extracción de datos se elaboró un formulario estandarizado (matriz en Microsoft Excel) para recoger de manera sistemática la información metodológica y los resultados más importantes de cada estudio incluido. El usar el formulario y procedimientos establecidos para la recopilación de datos, ayuda a mejorar la consistencia y reducir los errores humanos. (Okoli, 2015).

La matriz de extracción consideró variables bibliográficas y metodológicas como:

- Año
- Título,
- Autores,
- Tipo de publicación,
- Revista/evento,
- País,
- Institución,
- Enfoque,

- Alcance,
- Tipo, diseño,
- Objetivo del estudio
- Resultados/conclusiones.

4.6 Valoración de la calidad

Como resultado del proceso de búsqueda, se identificó inicialmente un total de 116 recursos (entre los años 2020 y 2025), posteriormente se realizó una depuración de aquellos estudios que no incluían los términos de búsqueda establecidos o correspondían a términos homónimos asociados a otras áreas de conocimiento.

Tras la aplicación de estos filtros, el número de documentos se redujo a 16, de los cuales forman el cuerpo, sin embargo, no todos estos fueron un aporte directo para responder a las preguntas planteadas en esta revisión.

Una vez asignadas las puntuaciones a cada documento por parte de los revisores, se calcularon los promedios correspondientes, para posteriormente, seleccionar únicamente aquellos estudios que obtuvieron una calificación mayor o igual a 3, como se presenta en la Tabla 2 con los criterios y la puntuación a tomarse en cuenta. Como resultado de este proceso, el conjunto final quedó conformado por 12 estudios, a partir de los cuales se desarrolló la síntesis de la información.

Tabla 5

Criterios para evaluar la calidad de las publicaciones

Puntuación	Criterio
1	Constituyen estudios bibliográficos descriptivos sin mayor profundidad.
2	Son aportes teóricos que sólo proponen nuevos enfoques o tendencias.
3	Incluyen propuestas de diseño y/o implementaciones debidamente fundamentadas.
4	Presentan experiencias prácticas de implementación sobre contextos reales de aprendizaje.
5	Además de lo contemplado en el nivel 4, evalúan dichas implementaciones, presentando resultados claros y replicables.

Nota. Puntuación y criterios para valorar rigor de las publicaciones.

4.7 Síntesis de la información

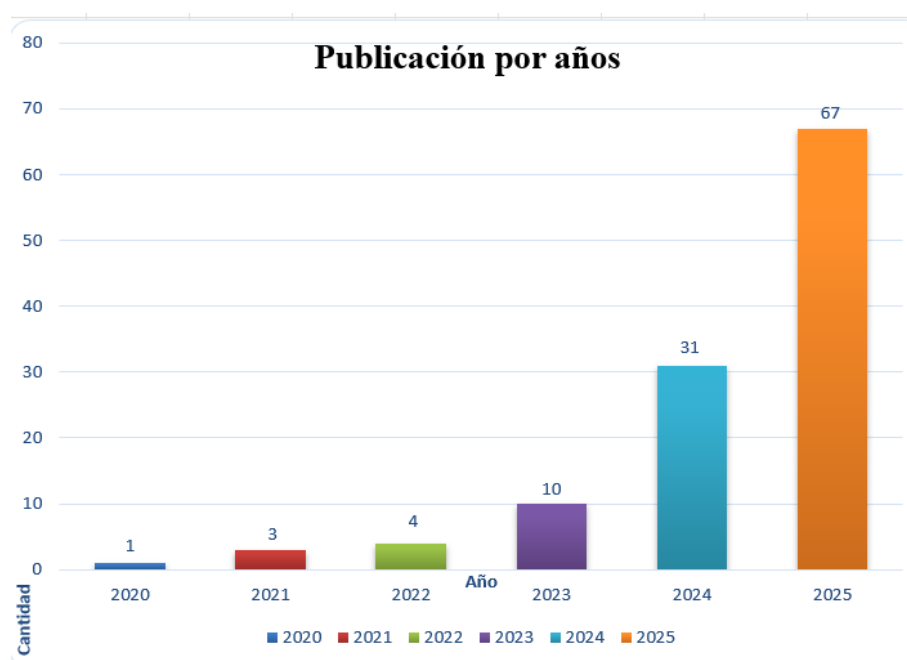
Basándose en la información recolectada, se procedió a ejecutar la fase de síntesis orientada a integrar los hallazgos de cada documento. Ese proceso permitió obtener una perspectiva global de la literatura y determinar la contribución específica de cada estudio a las preguntas de investigación (Okoli, 2015).

4.8 Redacción de la revisión

La distribución temporal de los estudios incluidos en la revisión sistemática evidencia un crecimiento progresivo en la producción científica a lo largo del periodo. Analizado en los primeros años 2020, 2021 y 2022 se observa una baja cantidad de publicaciones, con uno, tres y cuatro estudios respectivamente, lo que refleja un interés inicial limitado en la temática, como se observa en la Figura 4.

Figura 4

Distribución temporal de las publicaciones incluidas en la revisión



Nota. Evolución del número de publicaciones entre 2020 y 2025.

Sin embargo, a partir del 2023 se identifica un aumento significativo, alcanzando 10 publicaciones, seguido de un crecimiento marcado en 2024 con 31 estudios y en el año 2025 con 67 investigaciones.

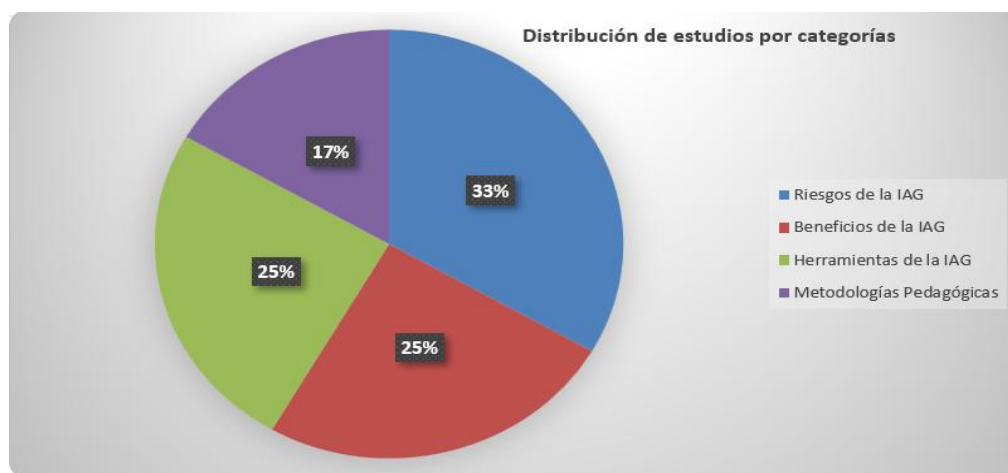
Esta tendencia confirma el carácter emergente y de rápida expansión del uso de la inteligencia artificial generativa en el aprendizaje de la programación, así como el creciente interés de la comunidad académica en los años más recientes.

La distribución de las publicaciones por categorías, de acuerdo con las preguntas de investigación, tomando en cuenta únicamente los 12 estudios considerados luego de la valoración de la calidad, se evidenció que la mayor proporción de trabajos se concentran en el análisis de los riesgos asociados al uso de la inteligencia artificial generativa en el aprendizaje de la programación, representando aproximadamente el 33% de los estudios.

En segundo lugar, se ubican las investigaciones centradas tanto en los beneficios de la inteligencia artificial generativa como en las herramientas empleadas, cada una con cerca del 25% del total. Finalmente, en menor proporción, se identifican los estudios que analizan las metodologías pedagógicas aplicadas en conjunto con la inteligencia artificial generativa los cuales representan alrededor del 17%, como se observa en la Figura 5.

Figura 5

Distribución



Nota. Distribución de estudios por categorías

A continuación, se detalla de manera cualitativa los estudios cuidadosamente seleccionados de acuerdo con las preguntas de investigación que fueron planteadas al inicio del estudio.

a) ¿Qué tipos de herramientas de IAG se está utilizando para aprender programación?

Respecto a la primera interrogante de investigación acerca de ¿Los tipos de herramientas de inteligencia artificial generativa utilizadas para aprender programación?, los estudios se desarrollan principalmente en contextos universitarios.

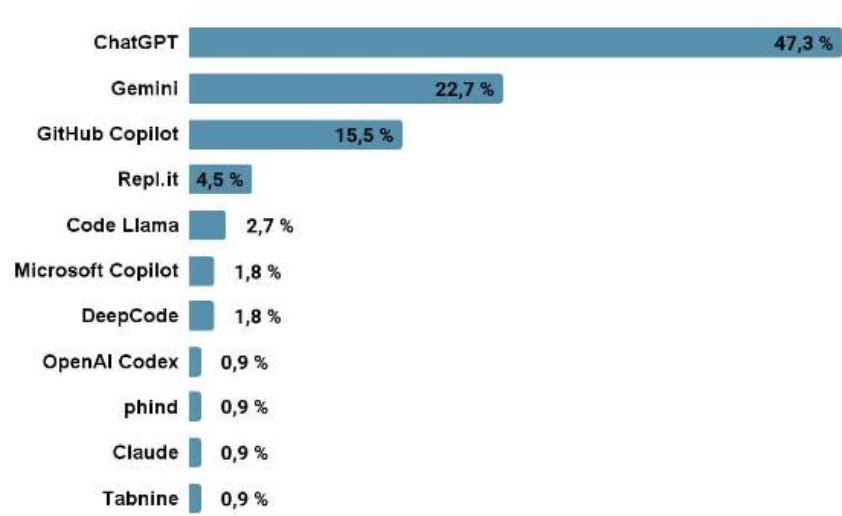
Como menciona Silva y otros autores (2025) el aprendizaje de la programación con apoyo de herramientas de inteligencia artificial generativa se centra en el uso de asistentes conversacionales y generadores de código, en particular usan ChatGPT; este estudio se realizó con una muestra $n = 859$, el uso frecuente de herramientas de IAG tuvo el predominio de ChatGPT con 86.2 % seguido por Gemini y GitHub Copilot, lo que refleja una preferencia por plataformas capaces de ofrecer explicaciones, ejemplos de código y retroalimentación inmediata. En menor medida, se identifican otras herramientas como Replit, Code Llama, Microsoft Copilot y DeepCode, así como herramientas menos utilizadas como OpenAI, Codex, Phind, Claude y Tabnine como se observa en la Figura 6.

De igual manera, Cubillos et al., (2025) llevaron a cabo un estudio en el que experimento con estudiantes de ingeniería, donde se emplea Google Gemini 1.5 como herramienta de IAG para apoyar el aprendizaje, comparándose su uso con un enfoque tradicional basado en videos educativos.

Los descubrimientos nos revelan qué ambos métodos mejoran el rendimiento académico tanto en el pretest como en el post-test; no se detectaron diferencias numéricas notables en los grupos. Esto indica que las herramientas de inteligencia artificial como Google Gemini 1.5 se emplean mayormente como aliados auxiliares en el aprendizaje de la programación en vez de ser un sustituto directo de las estrategias clásicas.

Figura 6

Uso de herramientas de inteligencia artificial generativa en el aprendizaje de la programación



Nota. Tipos de herramientas y su frecuencia de uso en estudios.

Otro trabajo es el de Lepp y Kaimre (2025) en donde participaron 21 estudiantes principiantes de programación, quienes emplearon herramientas de GenAI para generar código, recibir sugerencias, obtener pistas para la resolución de problemas y comprender mensajes de error. Las herramientas utilizadas fueron ChatGPT y GitHub Copilot que beneficiaron a estudiantes con mayor preparación previa, pero que mostraron dificultades metacognitivas en estudiantes con bajo desempeño.

Nora y otros autores (2024) presenta un estudio acerca del desempeño de ChatGPT-3.5 frente a la plataforma de evaluación automática Kattis en tareas de programación de nivel introductorio. Los resultados muestran que ChatGPT logra resolver con éxito 19 de los 127 trabajos, únicamente pudo resolver problemas de nivel fácil, lo que evidencia sus limitaciones para apoyar a estudiantes en cursos básicos de programación. Los autores señalan varias restricciones metodológicas.

Tabla 6

Síntesis de la literatura sobre herramientas de Inteligencia Artificial Generativa

Descripción	Documentos
Herramientas de inteligencia artificial generativa utilizadas como asistentes conversacionales y generadores de código para apoyar el aprendizaje de la programación, tanto en contextos formales como informales.	(Silva et al., 2025)

Herramientas de IAG empleadas como apoyo educativo comparadas con métodos tradicionales, tales como videos instructivos, para evaluar su impacto en el rendimiento académico, aceptación tecnológica y motivación estudiantil.	(Mellado et al., 2024)
Ayuda de la inteligencia Artificial generativa como herramienta en el aprendizaje de la programación	(Lepp & Kaimre, 2025)
El estudio compara ChatGPT con la plataforma Kattis en tareas de programación, mostrando que la IA generativa resuelve problemas simples, pero presenta limitaciones en niveles más complejos, lo que evidencia la necesidad de un uso pedagógico responsable.	(Nora, Dunder et al., 2024)

Nota. Principales hallazgos sobre herramientas IAG y su aplicación.

b) ¿Cuáles son los beneficios sobre el uso de la inteligencia artificial generativa en el aprendizaje de la programación?

En respuesta a la segunda pregunta de investigación según Jin (2024), el uso de herramientas de inteligencia artificial generativa reduce el costo de aprendizaje y hace que el proceso educativo sea percibido como más sencillo; sin embargo, no garantiza mejoras en el beneficio informativo ni en la compatibilidad del conocimiento con los esquemas mentales de los estudiantes. Por ello, aunque estas herramientas pueden agilizar el aprendizaje, su uso por sí solo no asegura una construcción profunda y personalizada del conocimiento, siendo necesaria la formación u orientación de los estudiantes para lograr beneficios educativos reales.

Por otro lado, la investigación de Prather y otros autores (2024), demostró que las herramientas de inteligencia artificial generativa ofrecen beneficios claros para los estudiantes con mayor preparación y autoeficacia, quienes logran acelerar su aprendizaje consiguiendo reducir dificultades metacognitivas. Los principiantes con bases débiles pueden generar una ilusión de competencia y reforzar conceptos erróneos, lo cual obstaculiza su avance.

No obstante, es crucial admitir que estas ventajas brillan con mayor claridad en quienes poseen ya bases sólidas. Esto demanda la creación de métodos que permiten a los aspirantes utilizar con atención estas herramientas, evitar el engaño a uno mismo y asegurar un aprendizaje profundo y duradero.

En otra investigación velho (2025), lleva a cabo un experimento meticuloso en dos etapas con estudiantes de grado. En la etapa inicial, un grupo utilizo copilot para ejecutar tareas de organización de datos mientras que otro lo hizo sin utilizar dicha herramienta. Este estudio se dedica a descubrir como copilot influye en la obtención y conservación de saberes en el ámbito de la programación.

El estudio reveló que copilot incrementó notablemente la productividad, el compromiso y la retención de saberes; El grupo experimental completó más rápido las tareas con más éxito también hubo una disminución de errores en las evaluaciones posteriores, además en la partida inicial y en la evaluación de un mes después nos revelaron que una intervención temprana en Copilot mejora la retención de conceptos de programación en comparación con el grupo de control.

En conjunto, los hallazgos apoyan la utilidad del Copilot en la educación asistida por tecnología, destacando beneficios duraderos en la retención de conocimiento y la importancia de un uso pedagógicamente guiado

Silva y otros autores (2025), en su estudio acerca de la aplicación de herramientas de IA generativa en la instrucción de programación, resaltan que estas tecnologías benefician la autonomía del alumno y enriquecen el proceso educativo al proporcionar apoyo inmediato y personalizado, como se presenta en la Figura 7 sobre los beneficios que ofrece la IAG.

Los hallazgos de esta investigación revelan que los estudiantes destacan beneficios como la resolución de dudas puntuales con un (16,6%), entrega de ejemplos y ejercicios personalizados (15,2%), aceleración para redactar código (15,2%) y la corrección automática de errores (14,3%).

Además, se tienen en cuenta otros beneficios en menor porcentaje como la búsqueda de información, el feedback en tiempo real, la solución de problemas complicados y el análisis de contenidos concretos. Estos hallazgos demuestran que la IA generativa no solamente hace más eficiente el tiempo de aprendizaje, sino que además ayuda a mejorar el código de manera constante y a desarrollar habilidades cruciales en programación.

Figura 7

Beneficios del uso de Inteligencia artificial generativa



Nota. Beneficios pedagógicos más reportados en los estudios analizados.

Por último, en la investigación de Hartley y otros autores (2024) se evidencia que el uso de ChatGPT aporta beneficios relevantes en el aprendizaje de programación, especialmente en contextos de aprendizaje autorregulado. Los resultados muestran que la herramienta es altamente efectiva en cuatro áreas clave: provisión de materiales didácticos, apoyo al contenido, planificación del aprendizaje y retroalimentación.

En particular, ChatGPT facilita el acceso inmediato a recursos educativos, apoya la organización y planificación del aprendizaje independiente también proporciona orientación experta y retroalimentación rápida durante la resolución de problemas de programación en Python.

A continuación, se presenta la Tabla 4 donde se muestran los beneficios del uso de las herramientas de IAG.

Tabla 7

Beneficios sobre el uso de la inteligencia artificial generativa en el aprendizaje de la programación

Descripción	Documentos
La IA generativa reduce el esfuerzo de aprendizaje y fortalece la autoeficacia, pero solo mejora el rendimiento cuando existe un adecuado ajuste del conocimiento y orientación en su uso.	(Jin et al., 2025)

La inteligencia artificial generativa beneficia el aprendizaje de la programación al facilitar la resolución de problemas y acelerar la producción de soluciones en programadores novatos, mediante ejemplos, fragmentos de código y sugerencias que incrementan la productividad y permiten avanzar con mayor rapidez en tareas prácticas. (Prather et al., 2024)

El uso de GitHub Copilot en el aprendizaje de programación incrementa la productividad, el compromiso y la retención del conocimiento, al permitir completar más actividades, reducir errores y mejorar el desempeño y la conservación de conceptos a mediano plazo (Velho, 2025)

ChatGPT beneficia el aprendizaje de programación al apoyar el aprendizaje autorregulado mediante el acceso a recursos educativos, la planificación del estudio y la retroalimentación inmediata, reduciendo el esfuerzo y facilitando la resolución de problemas en Python. (Hartley et al., 2024)

Las herramientas de IA generativa aportan beneficios significativos al aprendizaje de programación, principalmente al resolver dudas puntuales, proporcionar ejemplos prácticos, acelerar la escritura de código y corregir errores. (Silva et al., 2025)

Nota. Beneficios medibles y pedagógicos del uso de IAG.

c) ¿Qué riesgos se identifican al utilizar inteligencia artificial generativa en los ámbitos educativos?

La investigación de Prather y otros autores (2024), evidencia que el uso de herramientas de inteligencia artificial generativa en el aprendizaje de programación puede generar nuevas dificultades metacognitivas, especialmente en estudiantes con menor experiencia, ya que la IA puede proporcionar soluciones completas que disminuyen la activación de procesos de pensamiento crítico y de análisis propio, lo que puede llevar a una ilusión de competencia y a una comprensión superficial del contenido cuando no se evalúa ni se reflexiona sobre lo generado por la IA.

En el estudio de Heloise Acco y otros autores (2025) se revela que, a pesar de los beneficios potenciales de la inteligencia artificial generativa en el aprendizaje de programación, también se evidencia variabilidad en la calidad de las respuestas (errores, omisiones y sesgos) y un riesgo medible de dependencia por parte del alumnado cuando la herramienta se usa sin mediación.

En las pruebas comparativas, las herramientas funcionan mejor en tareas de apoyo (andamiaje, sugerencias y correcciones parciales) que, en evaluaciones de comprensión

profunda, como su efectividad aumenta notablemente cuando se combinan con actividades que exigen verificación, pruebas y justificación por parte del estudiante.

Por otro lado, en la investigación de Silva y otros autores (2025) se evidencia que el uso de herramientas de inteligencia artificial generativa en el aprendizaje de programación, aunque es percibido por los estudiantes como un recurso eficaz para la resolución de problemas y la comprensión de conceptos, también plantea riesgos importantes como se presenta en la Figura 8.

Figura 8

Riesgos sobre el uso de la Inteligencia Artificial Generativa



Nota. Riesgos sobre el uso y su importancia en contextos educativos.

Los investigadores destacan que depender de estas tecnologías puede limitar la autonomía, el desarrollo del pensamiento crítico y generar una comprensión superficial del contenido cuando no son evaluadas por un docente ni reflexionadas por el estudiante.

También señalan riesgos como la sobre confianza en las respuestas generadas por IA, la posible disminución de esfuerzo cognitivo y la necesidad de marcos pedagógicos sólidos para que la IA no comprometa la calidad del aprendizaje.

A continuación, se presenta la Tabla 5 donde se muestra un resumen de los estudios relacionados con los riesgos de la IAG.

Tabla 8

Resumen de los estudios relacionados con los riesgos al utilizar Inteligencia Artificial Generativa

Descripción	Documentos
Riesgos del uso de IA generativa en programadores principiantes	(Prather et al., 2024)

Riesgos del uso de IA generativa en educación: dependencia cognitiva, comprensión superficial y reducción del pensamiento crítico (Heloise Tives et al., 2025)

Riesgos del uso de IA generativa en programación: dependencia moderada, problemas de evaluación y la difusión de respuestas erróneas o sesgadas. (Silva et al., 2025)

Nota. Resumen de riesgos y limitaciones encontrados en la literatura.

d) ¿Qué metodologías pedagógicas se aplican en combinación con inteligencia artificial generativa para la enseñanza de la programación?

La revisión de la literatura identifica que la metodología predominante para la integración de la inteligencia artificial generativa en la enseñanza de la programación es el Plan de Aprendizaje Activo (ALP) articulado mediante el modelo de aula invertida (flipped classroom).

Según los hallazgos de Portella & Rodríguez (2024), aplicar esta metodología en cursos de ingeniería de sistemas reveló una notable eficacia en los estudiantes logrando una aprobación del 91% y asegurando que los estudiantes obtuvieran resultados óptimos.

Esta metodología se caracteriza por su innovación radical permitiendo enfocarse en la resolución de problemas, la colaboración y la personalización del aprendizaje a través de retroalimentación adaptativa y analítica de datos. Al final esta combinación metodológica no solo mejora la comprensión técnica de la programación también nos ayuda a preparar al estudiante para las exigencias del mercado laboral al integrar de manera efectiva las herramientas de IA en su formación profesional.

La implementación de la inteligencia artificial generativa en la enseñanza de la programación se entrelaza principalmente con metodologías de aprendizaje colaborativo, donde estas herramientas sirven como apoyo, aliados para la resolución de problemas técnicos y la generación de ideas dentro de los equipos de trabajo. La implementación de la IAG fortalece las dinámicas colaborativas provocando la productividad y la creatividad cuando se utiliza como complemento.

Además, se destaca la incorporación de metodologías de aprendizaje activo donde la inteligencia artificial generativa emerge como un aliado que complementa el aprendizaje sin necesidad de sustituir el razonamiento lógico y el razonamiento crítico de los estudiantes. Sin embargo, los descubrimientos destacan la necesidad de un uso controlado y supervisado con el fin de evitar la dependencia excesiva y garantizar el desarrollo de competencias esenciales.

Además, Quispe et al. (2025), enfatizan en la importancia de incorporar estrategias pedagógicas orientadas a la capacitación y accesibilidad que van a estar destinados a reducir las brechas en el uso de herramientas como ChatGPT o GitHub Copilot, causadas por el desconocimiento, limitaciones de acceso y la oposición por las nuevas herramientas.

Finalmente, ese estudio resalta la relevancia de una implementación ética y guiada, que permita evaluar de manera equitativa las contribuciones individuales y minimizar riesgos asociados a errores o sesgos, consolidando a la inteligencia artificial generativa como un recurso complementario dentro de un modelo pedagógico centrado en el aprendizaje colaborativo, activo y consciente.

La implementación del modelo de aula invertida articulada con ChatGPT en la enseñanza de los fundamentos de programación no necesariamente produce un incremento en la media general de las calificaciones, pero genera un impacto positivo significativo en la distribución del rendimiento académico. De acuerdo con los resultados del estudio comparativo (2019-2023), esta integración metodológica logró reducir la tasa de abandono estudiantil del 14.64% al 6.97%, al tiempo que incrementó la proporción de estudiantes en el nivel de excelencia, alcanzando un 20.93%.

Asimismo, en la investigación de Calsin y otros autores (2023), la percepción estudiantil sobre la metodología es mayoritariamente positiva, ya que los estudiantes reportan comprensión del contenido y resolución de dudas mediante un número reducido de consultas. No obstante, se identifican limitaciones relacionadas con la complejidad o falta de claridad de algunas respuestas, así como con el desconocimiento inicial del tema y restricciones de tiempo.

En consecuencia, el estudio destaca la necesidad de capacitación previa en el enfoque de aula invertida y en la formulación adecuada de consultas, señalando que el uso efectivo de ChatGPT, como apoyo pedagógico, depende del conocimiento de cómo interactuar con la herramienta sin comprometer los objetivos de aprendizaje.

A continuación, se presenta la Tabla 6 donde se muestran los estudios relacionados con las metodologías pedagógicas aplicadas junto a las herramientas IAG para el aprendizaje de la programación.

Uyoip Tabla 9

Metodologías pedagógicas que se aplican en combinación con las herramientas IAG para el aprendizaje de la programación.

Descripción	Documentos
Analiza enfoques pedagógicos para la integración de la inteligencia artificial generativa en la educación, destacando la necesidad de estrategias didácticas guiadas que apoyen el aprendizaje activo sin sustituir el rol del docente.	(Portella & Rodriguez, 2024)
Evalúa la efectividad del uso de ChatGPT con un enfoque de aula invertida en fundamentos de programación, mostrando que no mejora el promedio general de calificaciones, pero sí favorece la distribución del rendimiento académico y presenta una	(Calsin et al., 2023)

percepción estudiantil mayormente positiva, destacando la necesidad de capacitación previa en el uso de la herramienta.

Analiza el impacto de la inteligencia artificial generativa en la programación colaborativa, destacando su aporte al trabajo en equipo y al apoyo en la resolución de problemas, así como la necesidad de un uso equilibrado para no afectar el pensamiento crítico. (Quishpe et al., 2025)

Nota. Comparación de metodologías que integran IAG en programación.

CAPÍTULO V.

5. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- El establecer un protocolo de búsqueda bien definido, dentro de la metodología para la Revisión Sistemática de Literatura (SLR), permitió disminuir sesgos al seleccionar solamente investigaciones relevantes y reforzar así la validez del análisis acerca del uso de la Inteligencia Artificial Generativa en el aprendizaje de la programación.
- Al realizar las recopilaciones de publicaciones académicas bajo criterios de inclusión y calidad, se evidencia que la producción científica relacionada con la inteligencia artificial generativa en educación ha experimentado un crecimiento sostenido en los últimos años, lo que refleja un interés creciente de la comunidad académica por su aplicación en la enseñanza de la programación.
- Al sintetizar los resultados basados en las preguntas de investigación de la SLR, se pudo determinar que la inteligencia artificial generativa facilita el entendimiento de conceptos relacionados con la programación, fomenta el aprendizaje independiente y brinda soporte individualizado, sobre todo en alumnos con diferentes grados de conocimiento previo.

5.2 RECOMENDACIONES

- Se recomienda que las investigaciones futuras progresen desde enfoques de revisión documental hacia diseños experimentales o cuasi-experimentales, planteando nuevas cuestiones investigativas enfocadas en medir el rendimiento académico, la motivación y el aprendizaje independiente de los alumnos, al incorporar instrumentos de inteligencia artificial generativa en el aprendizaje de la programación.
- Se sugiere poner en práctica los resultados evidenciados en las investigaciones examinadas sobre el uso de la IAG en el aprendizaje de la programación, en ambientes universitarios del Ecuador o considerando sus propias características institucionales, tecnológicas y pedagógicas.
- Finalmente, se sugiere una elaboración e implementación de guías metodológicas o planes de aprendizaje activo que integren la IAG como una herramienta pedagógica, acorde a los objetivos curriculares y a las necesidades de los estudiantes.

BIBLIOGRAFÍA

- Alberto, C., & Sebastián, K. (s. f.). Integración de la Inteligencia Artificial Generativa en la elaboración de evaluaciones formativas en el proceso de aprendizaje para la etapa de Diseño de Software.
- Bolaño-García, M., & Duarte-Acosta, N. (2024). Una revisión sistemática del uso de la inteligencia artificial en la educación. *Revista Colombiana de Cirugía*, 39(1), 51-63.
- Calsin, M. A., Aedo, M., Castro, E., Calsin, M. A., Aedo, M., & Castro, E. (2023). Impacto de ChatGPT en la enseñanza: Un enfoque de aula invertida para fundamentos de programación. *RISTI - Revista Ibérica de Sistemas e Tecnologias de Informação*, (52), 97-112. <https://doi.org/10.17013/risti.52.97112>
- Chávez Boza, B. M., & Erazo Moreta, O. (2024). Integración de la inteligencia artificial generativa para el aprendizaje de fundamentos de programación: Una revisión sistemática de la literatura. *Revista Mexicana de Investigación e Intervención Educativa*, 3(2), 5-17.
- Dominguez, M., & Puga, M. (s. f.). Integración de la Inteligencia Artificial Generativa en la elaboración de evaluaciones formativas en el proceso de aprendizaje en la etapa de la implementación del Software.
- Freire-Avilés, R. M., Muzzio-Arguello, J. M., Freire-Avilés, V. A., & Vélez-San-Martín, B. O. (2024). Inteligencia artificial en la educación: Una revisión sistemática de la transformación de la enseñanza de Python mediante ChatGPT en la educación superior. *MQRInvestigar*, 8(4), 4712-4738. <https://doi.org/10.56048/MQR20225.8.4.2024.4712-4738>
- Hartley, K., Hayak, M., & Ko, U. H. (2024). Artificial Intelligence Supporting Independent Student Learning: An Evaluative Case Study of ChatGPT and Learning to Code. *Education Sciences*, 14(2), 120. <https://doi.org/10.3390/educsci14020120>
- Heloise Tives, Madianita Marioti, & Patricia Jaques. (2025). Vista do Generative AI Assisting Programming Learning: Functional Roles, Comparative Analysis, and Emerging Challenges. <https://sol.sbc.org.br/index.php/sbie/article/view/38440/38214>
- Jin, Y., He, W., Shen, J., & Hu, J. (2025). Mechanisms of enhancing learning with unequal preparation: An experimental study on generative artificial intelligence use and proficiency in programming learning. *Computers and Education: Artificial Intelligence*, 9, 100467. <https://doi.org/10.1016/j.caeai.2025.100467>
- Lepp, M., & Kaimre, J. (2025). Does generative AI help in learning programming: Students' perceptions, reported use and relation to performance. *Computers in Human Behavior Reports*, 18, 100642. <https://doi.org/10.1016/j.chbr.2025.100642>
- López-Regalado, Ó., Núñez-Rojas, N., López-Gil, Ó. R., & Sánchez-Rodríguez, J. (2024). Análisis del uso de la inteligencia artificial en la educación universitaria: Una revisión sistemática. <https://doi.org/10.12795/pixelbit.106336>
- Mellado, R., Cubillos, C., & Ahumada, G. (2024). Effectiveness of Generative Artificial Intelligence in learning programming to higher education students. 2024 IEEE International Conference on Automation/XXVI Congress of the Chilean Association of Automatic Control (ICA-ACCA), 1-7. <https://doi.org/10.1109/ICA-ACCA62622.2024.10766746>

- Nora, Dunder, Saga, Lundborg, Jacqueline Wong, & Olga Viberg. (2024, marzo). Kattis vs ChatGPT: Evaluación de tareas de programación en la era de la inteligencia artificial. <https://doi.org/10.1145/3636555.3636882>
- Okoli, C. (2015). A Guide to Conducting a Standalone Systematic Literature Review. *Communications of the Association for Information Systems*, 37. <https://doi.org/10.17705/1CAIS.03743>
- Pillpe, R., Inca, M., Soto, G. E., Franco, A., Ore, A. M., Universidad Nacional San Luis Gonzaga, Ore, M. K., & Universidad Nacional San Luis Gonzaga. (2025). Revisión sistemática de inteligencia artificial generativa (GenIA) para el diseño de experiencias de aprendizaje, 2020-2025. *Espaci*, 46(03), 13-27. <https://doi.org/10.48082/espacios-a25v46n03p02>
- Portella, J., & Rodriguez, A. (2024, enero 23). Enhancing Programming Education with an Active Learning Plan and Artificial Intelligence Integration. https://www.researchgate.net/publication/384175218_Enhancing_Programming_Education_with_an_Active_Learning_Plan_and_Artificial_Intelligence_Integration
- Prather, J., Reeves, B. N., Leinonen, J., MacNeil, S., Randrianasolo, A. S., Becker, B. A., Kimmel, B., Wright, J., & Briggs, B. (2024). The Widening Gap: The Benefits and Harms of Generative AI for Novice Programmers. *Proceedings of the 2024 ACM Conference on International Computing Education Research - Volume 1, ICER '24*, 1, 469-486. <https://doi.org/10.1145/3632620.3671116>
- Quishpe, M. J. C., Borroso, J. L. C., & Jaime, D. J. F. (2025). Inteligencia artificial generativa y su impacto en la programación para fortalecer el aprendizaje colaborativo. *Revista de Investigación Multidisciplinaria para la Ciencia Abierta*, 1(2), 1-21.
- Salazar, M., Fernadez, J., Sobenis, F., & Navarro. (s. f.). La inteligencia artificial generativa como herramienta de apoyo en la personalización del aprendizaje: Implicaciones y desafíos éticos en el aula para estudiantes de EGB | Reincisol. Recuperado 17 de marzo de 2026, de https://www.reincisol.com/ojs/index.php/reincisol/article/view/536?utm_source=chatgpt.com
- Silva, T. L. da, Vidotto, K. N. S., Tarouco, L. M. R., & Silva, P. F. da. (2025). Análise do Uso de Ferramentas de IA Generativa no Ensino de Programação: Perspectivas de Estudantes do Curso de Sistemas de Informação. *Workshop sobre Educação em Computação (WEI)*, 1277-1288. <https://doi.org/10.5753/wei.2025.9236>
- Stryker, C., & Kavlakoglu. (2024, agosto 9). ¿Qué es la Inteligencia Artificial (IA)? | IBM. <https://www.ibm.com/es-es/think/topics/artificial-intelligence>
- Velho, M. A. de B. (2025). Investigando o impacto da inteligência artificial generativa no aprendizado em programação. <https://lume.ufrgs.br/handle/10183/294939>

ANEXOS

ANEXO 1: Pantalla de práctica del proceso de selección de la literatura como se

menciona en la figura 3

No	Año	Título	Autores	Tipo	Revista / Fuente	País	Institución	Enfoque	Alcance	Tipo	Diseño	Objetivo	Resultado / Conclusión	Cita	URL
1	2023	Mechanisms of enhancing learning with unequal preparation: An experimental study on generative artificial intelligence use and proficiency in programming learning (Mecanismos para mejorar el aprendizaje con una preparación desigual: Un estudio experimental sobre el uso y la competencia de la inteligencia artificial generativa en el aprendizaje de programación)	Zin, Yuan; Li, wei; Shen, Junbo; Hu, Shengyan	Artículo	Computadoras y educación: Inteligencia artificial	USA	U-Tech Univ	Cuantitativo	Explícitivo	De Campo	Diseño Experimental	Identificar los efectos del uso de herramientas GenAI en los resultados de aprendizaje a través de su impacto en el beneficio informativo percibido, el costo del aprendizaje y la adecuación del conocimiento.	RESULTADOS: Los resultados muestran que, aunque se esperaba que las herramientas aumentadas de GenAI ayudarían a reducir la brecha educativa, aún no se ha comprobado que beneficien de manera efectiva a los estudiantes desfavorecidos; para aprovechar su potencial es indispensable que los alumnos dominen su uso. En el ámbito educativo general, los hallazgos subrayan que los docentes deben enseñar el uso adecuado de las herramientas de IA, adaptar las orientaciones según el nivel de preparación de los estudiantes y brindar apoyo especial a los más desfavorecidos; combinando el uso de GenAI para aliviar cargas cognitivas excesivas pero promoviendo al mismo tiempo el esfuerzo y el compromiso, con el fin de evitar una dependencia excesiva que termine perjudicando la eficacia del aprendizaje. CONCLUSIÓN: Los autores concluyen que las herramientas GenAI, como ChatGPT, aportan un panorama complejo en educación. Su principal aporte es reducir el costo y esfuerzo del aprendizaje y hacer que este parezca más fácil, pero por sí solas no mejoran.	Zin et al., (2023)	https://www.mdpi.com/journal/ai/article/2023/04/04/743#
2	2024	Artificial Intelligence Supporting Independent Student Learning: An Evaluative Case Study of ChatGPT and Learning to Code (Inteligencia artificial que apoya el aprendizaje independiente de los estudiantes: un estudio de caso evaluativo de ChatGPT y Learning to Code)	Kendall Hartley,ORCID iD,Merav Hayek, Yan Iyeyok Ko	Artículo	Educación Secundaria (MDPI)	Estados Unidos (autores en EE. UU.) y colaborador en Israel (afiliaciones de autores)	University of Nevada, Las Vegas; Ben-Gurion University (afiliaciones de autores)	Cuantitativo y cualitativo (estudio evaluativo)	Investigación de caso evaluativo	Investigación aplicada/Evaluativa	Diseño de caso evaluativo	Este estudio investigó el potencial de ChatGPT para apoyar el aprendizaje autoregulado y el aprendizaje de un lenguaje de programación.	Conclusiones Este estudio analizó el potencial de ChatGPT para apoyar el aprendizaje autoregulado y la programación en Python en cuatro áreas: materiales didácticos, herramientas de contenido, planificación y evaluación. Los resultados muestran que ChatGPT es muy efectivo para proporcionar recursos educativos y ayudar a clarificar el aprendizaje independiente, así como para dar retroalimentación inmediata y orientación experta en programación; sin embargo, presenta limitaciones importantes para proporcionar recursos educativos y ayudar a experiencias prácticas e interactivas compatibles con IDE, e individualizar de forma consistente la enseñanza. Su utilidad depende en gran medida de las habilidades de autoregulación y metacognición del estudiante, pues el sistema apoya principalmente a quienes formulan buenas preguntas y reflejan sus indicaciones. Aunque la IA puede facilitar la planificación y el acceso a información, sigue siendo poco adecuada para aspectos como la motivación, el monitoreo profundo de la comprensión y la personalización de la enseñanza.	Hartley et al., (2024)	https://www.mdpi.com/2227-7092/14/2/326
3	2024	The Widening Gap: The Benefits and Harms of Generative AI for Novice Programmers (La brecha cada vez mayor: beneficios y perjuicios de la IA generativa para programadores novatos)	James Prather; Brent N. Reeves; Lulu Lincon; Stephen Machell; Arissa S. Kandravanadu; Brett A. Becker; Bailey Kinmeel; Jared Wright; Ben Briggs;	Artículo de investigación	ICER 2024 (Procedimiento)	Internacional (al Autores en EE. UU., Irlanda, Islandia)	Adelphi Christian University; Alcorn State University; Temple University; University College Dublin	Mixto	Experimentación (Observacional) y Cuantitativo	Investigación aplicada/Empírica	Comparativo experimental/observacional entre grupos	Evaluar beneficios y perjuicios del uso de la generativa (Generative Artificial o AI) para programadores novatos	Resultados: El estudio muestra que la IA generativa puede facilitar la resolución de problemas y acelerar la producción de soluciones entre programadores novatos, proporcionando ejemplos, fragmentos de código y sugerencias que aumentan la productividad y permiten avanzar más rápido en tareas prácticas; sin embargo, las observaciones comparativas entre grupos revelan efectos adversos importantes, como deficiencias metodológicas (novatos que no detectan errores conceptuales), una falsa sensación de progreso tras obtener soluciones plausibles y una falta de reflexión crítica que impide evaluar objetivamente las salidas de la IA y quienes no. Conclusión. Los autores concluyen que, aunque la IA generativa tiene potencial pedagógico para apoyar el aprendizaje de programación, su adopción sin mediación puede empeorar resultados educativos para ciertos novatos; por ello recomiendan diseñar intervenciones educativas específicas —por ejemplo, actividades que fomenten la verificación crítica, el diagnóstico metacognitivo y el análisis de El estudio muestra que la IA generativa mejora aspectos concretos del aprendizaje de programación: acelera la resolución de problemas, ofrece retroalimentación inmediata y facilita la generación de ejemplos y fragmentos de código adaptados al nivel del estudiante. Sin embargo, los resultados también evidencian variabilidad en la calidad de las respuestas (errores, omisiones y sesgo) y un riesgo medible de dependencia por parte del alumno cuando la herramienta se usa sin mediación. En las pruebas comparativas, las herramientas funcionan mejor en tareas de apoyo (ejemplares, sugerencias y correcciones parciales) que en evaluaciones de comprensión profunda, y su efectividad aumenta notablemente cuando se combinan con actividades que exigen verificación, pruebas y justificación por parte del estudiante. Conclusiones: Para la práctica educativa, se recomienda lo siguiente: (1) formación docente específica para la mediación en entornos de aprendizaje híbridos; (2) desarrollo de protocolos estructurados para la evaluación crítica de sugerencias automatizadas; y (3) implementación de modelos pedagógicos que eleven sistemáticamente la colaboración docente-alumno y la autonomía del aprendiz.	(Pashler et al., 2024)	https://doi.org/10.3389/fpsyg.2024.1262294
4	2025	Generative AI Assisting Programming Learning: Functional Roles, Comparative Analysis, and Emerging Challenges (IA generativa que asiste en el aprendizaje de la programación: Roles funcionales, análisis comparativo y desafíos emergentes)	Nelisse Accio Tiesi Bello, Madeline Rago Mariotti, Patricia Jaques	Artículo de conferencia	XIV Congreso Brasileiro de Informática em Educação (CBIE 2025)	Brazil	Federal University of Paraná (UFPR) – Curitiba – PR - Brazil Zuercher University Center of Palmas (CUZUPALMS) – PA – TO - Brazil	Mixto	Exploratorio y comparativo	Investigación aplicada/Evaluativa	Estudio de caso evaluativo con recolección de datos cualitativos y cuantitativos	Identificar roles funcionales, comparar herramientas y señalar desafíos.	Los autores concluyen que, aunque la IA generativa tiene potencial pedagógico para apoyar el aprendizaje de programación, su adopción sin mediación puede empeorar resultados educativos para ciertos novatos; por ello recomiendan diseñar intervenciones educativas específicas —por ejemplo, actividades que fomenten la verificación crítica, el diagnóstico metacognitivo y el análisis de conclusiones: La investigación concluye que la IA potencia el aprendizaje colaborativo en programación cuando se integra de forma guiada y pedagógica, aprendiendo beneficios claros en eficiencia y creatividad; no obstante, su implementación sin supervisión adecuada puede comprometer la calidad del aprendizaje.	Nelisse Accio,Mariotti P,Jaques P, (2025)	https://doi.org/10.3389/fpsyg.2024.1262294
5	2025	GENERATIVE ARTIFICIAL INTELLIGENCE AND ITS IMPACT ON PROGRAMMING: TO STRENGTHEN COLLABORATIVE LEARNING (INTELIGENCIA ARTIFICIAL GENERATIVA Y SU IMPACTO EN LA PROGRAMACIÓN PARA FORTALECER EL APRENDIZAJE COLABORATIVO)	Cepeda Quiroga Michailsoner ,Cruzeta Borrero Joyalisette, Figueroa Jaime DouglasRiver	Artículo de investigación	Revista BIMCA Vol 1 Issue 2 (2025)	Ecuador	Universidad Estatal de Guayaquil	Cuantitativo	Exploratorio/descriptivo	Investigación aplicada y empírica	Cuantitativo descriptivo (encuesta transversal)	Análisis del impacto de la inteligencia Artificial Generativa en el aprendizaje colaborativo en programación	El estudio, aplicado a una muestra de 23 estudiantes, mostró que el uso de la inteligencia Artificial Generativa (IAG) en actividades de programación incrementó la eficiencia en la resolución de tareas y fortaleció la creatividad en la generación de soluciones y alternativas de diseño. Las respuestas cuantitativas del cuestionario indicaron una percepción mayoritariamente positiva sobre la rapidez y la facilidad para prototipar ideas, así como una mejora en la colaboración al compartir y revisar experiencias generadas por la IAG. Sin embargo, se registraron evidencias de dependencia tecnológica que exigen pasos críticos a la IAG; actualizarse para evaluar aportes individuales, trabajar jornadas y episodios de resultados inconsistentes o sesgados provenientes de las herramientas (p. ej., soluciones incompletas o con errores lógicos). CONCLUSIONES: La investigación concluye que la IAG potencia el aprendizaje colaborativo en programación cuando se integra de forma guiada y pedagógica, aprendiendo beneficios claros en eficiencia y creatividad; no obstante, su implementación sin supervisión adecuada puede comprometer la calidad del aprendizaje.	(Quiroga et al., 2025)	https://revistabimca.com/index.php/bimca/article/view/916