



DEL PROMPT AL PENSAMIENTO COMPUTACIONAL: EVIDENCIAS DE LA INTEGRACIÓN DE LA INTELIGENCIA ARTIFICIAL GENERATIVA EN LA ENSEÑANZA DE LA PROGRAMACIÓN EN EDUCACIÓN MEDIA TÉCNICA

Sandra Esperanza Sánchez Barragán¹
ORCID: 0009-0000-7117-6853

¹ Doctoranda, Universidad Americana de Europa (UNAE). Correo: sandritaes.barragan@gmail.com

RESUMEN

La integración de la inteligencia artificial generativa (IAG) en la enseñanza de la programación representa una oportunidad para fortalecer el pensamiento computacional y las competencias digitales, siempre que su uso esté mediado por estrategias pedagógicas que promuevan el análisis crítico y la validación técnica del código. El objetivo de esta investigación fue analizar las evidencias derivadas de la integración de la inteligencia artificial generativa en la enseñanza de la programación mediante la formulación de prompts, la comparación de respuestas generadas por diferentes herramientas de IA y la validación del código para fortalecer el pensamiento computacional en estudiantes de educación media técnica. Se desarrolló un estudio con enfoque cualitativo y diseño de intervención educativa con 28 estudiantes del curso 1001 del Colegio Técnico Palermo IED. La estrategia didáctica incluyó la elaboración de prompts, la generación de código con ChatGPT, Gemini y Copilot, la comparación de las respuestas obtenidas, la validación técnica mediante SQLite Online y la reflexión sobre la calidad del código y las buenas prácticas de programación. Los resultados evidenciaron mejoras en la calidad de los prompts, mayor capacidad para identificar errores y validar soluciones, así como el fortalecimiento del pensamiento computacional, la argumentación técnica y la resolución de problemas. Se concluye que la integración pedagógica de la inteligencia artificial generativa favorece el aprendizaje activo cuando se articula con procesos de mediación docente, validación técnica y reflexión crítica.

Palabras clave: Educación Media Técnica; Inteligencia Artificial Generativa; Pensamiento Computacional.

ABSTRACT

Generative Artificial Intelligence (GenAI) is transforming programming education by creating new opportunities to strengthen computational thinking and digital competencies. This study analyzed the evidence derived from integrating generative artificial intelligence into programming education through prompt engineering, comparison of AI-generated solutions, and code validation to enhance computational thinking among technical secondary education students. A qualitative educational intervention was conducted with 28 students from Colegio Técnico Palermo IED. The instructional sequence included prompt formulation, code generation using ChatGPT, Gemini and Copilot, comparison of AI-generated responses, technical validation through SQLite Online, and critical reflection on programming best practices. The findings revealed improvements in prompt quality, code analysis, problem-solving, technical reasoning, and computational thinking. The study concludes that the pedagogical integration of generative artificial intelligence promotes active learning when combined with prompt engineering, technical validation, and teacher mediation.

Keywords: Computational Thinking; Generative Artificial Intelligence; Technical Secondary Education.

INTRODUCCIÓN

La transformación digital ha redefinido los procesos de enseñanza y aprendizaje en todos los niveles educativos, impulsando la incorporación de tecnologías emergentes que modifican las prácticas pedagógicas y las competencias requeridas por estudiantes y docentes. Entre estas tecnologías, la inteligencia artificial generativa (IAG) ha adquirido un protagonismo creciente debido a su capacidad para producir textos, código fuente, imágenes y otros recursos digitales a partir de instrucciones formuladas en lenguaje natural. Herramientas como ChatGPT, Gemini y Copilot están siendo incorporadas progresivamente en escenarios educativos, particularmente en áreas relacionadas con la programación, donde facilitan la generación automática de algoritmos, consultas SQL y estructuras de bases de datos (Dwivedi et al., 2023; Walter, 2024).

En el ámbito de la enseñanza de la programación, la inteligencia artificial generativa representa una oportunidad para apoyar la comprensión de conceptos complejos, proporcionar ejemplos de código y ofrecer retroalimentación inmediata. Sin embargo, también plantea desafíos pedagógicos importantes. La facilidad con la que estas herramientas generan soluciones funcionales puede conducir a que los estudiantes acepten las respuestas de manera acrítica, sin comprender la lógica subyacente, validar su funcionamiento o verificar el cumplimiento de estándares de calidad del software. En consecuencia, el desafío para las instituciones educativas no consiste únicamente en incorporar la inteligencia artificial al aula, sino en diseñar estrategias didácticas que promuevan su uso crítico, reflexivo y responsable.

En este contexto, el pensamiento computacional constituye una competencia fundamental para la formación de estudiantes en áreas de tecnología e informática. Desde la propuesta de Wing (2006), este concepto ha sido entendido como la capacidad para analizar problemas, diseñar soluciones, reconocer patrones, abstraer información relevante y desarrollar procedimientos sistemáticos susceptibles de ser implementados mediante herramientas computacionales. Más recientemente, diversos estudios han señalado que el desarrollo del pensamiento computacional debe trascender el aprendizaje de lenguajes de programación y orientarse hacia la construcción de habilidades analíticas, de resolución de problemas y de evaluación crítica de soluciones tecnológicas (Holmes et al., 2019; Hsu, 2025).

La aparición de la inteligencia artificial generativa introduce un nuevo escenario para el desarrollo de estas competencias. En lugar de limitarse a escribir código desde cero, los estudiantes deben aprender a formular instrucciones precisas para interactuar con modelos de lenguaje, interpretar las respuestas obtenidas, contrastarlas con diferentes fuentes y validar técnicamente los resultados antes de aceptarlos como correctos. Este proceso ha dado origen al denominado Prompt Engineering, entendido como la capacidad para diseñar, estructurar y optimizar instrucciones que permitan obtener respuestas pertinentes, completas y contextualizadas por parte de los sistemas de inteligencia artificial (Lo, 2023). Más que una habilidad técnica aislada, el Prompt Engineering constituye una competencia emergente asociada a la alfabetización digital y al uso responsable de la inteligencia artificial en contextos educativos y profesionales.

Desde esta perspectiva, la inteligencia artificial no debe concebirse como un sustituto del razonamiento humano, sino como un recurso de mediación pedagógica que favorece procesos de exploración, análisis y construcción del conocimiento. Su incorporación al aula requiere que los estudiantes desarrollen competencias para evaluar la calidad de las respuestas generadas, identificar posibles errores, reconocer limitaciones y aplicar criterios técnicos que permitan mejorar el código propuesto. En el caso de la programación de bases de datos, este proceso implica validar aspectos como la sintaxis SQL, la definición de claves primarias y foráneas, la integridad referencial, las convenciones de nomenclatura y la compatibilidad con diferentes motores de bases de datos.

Aunque la literatura reciente reconoce el potencial educativo de la inteligencia artificial generativa, gran parte de las investigaciones se han concentrado en describir sus posibilidades de uso o en analizar las percepciones de docentes y estudiantes frente a estas tecnologías (Dwivedi et al., 2023; Zawacki-Richter et

al., 2019). En contraste, son menos frecuentes los estudios que presentan evidencias empíricas derivadas de intervenciones didácticas, particularmente en educación media técnica, donde los estudiantes comparan diferentes herramientas de inteligencia artificial, validan técnicamente el código generado y reflexionan sobre la aplicación de buenas prácticas de programación. Asimismo, existe una limitada producción científica que analice cómo la formulación de prompts y la validación crítica del código pueden contribuir al fortalecimiento del pensamiento computacional en contextos reales de aula.

Con el propósito de aportar evidencia en este campo, la presente investigación analiza una experiencia de intervención educativa desarrollada con estudiantes de educación media técnica en el área de Tecnología e Informática, centrada en la formulación de prompts para programación de bases de datos, la comparación entre respuestas generadas por ChatGPT, Gemini y Copilot, y la validación del código mediante SQLite Online. La estrategia didáctica buscó transformar el uso de la inteligencia artificial desde una herramienta de generación automática de respuestas hacia un recurso para promover el análisis crítico, la argumentación técnica y la comprensión profunda de los principios de programación.

En este sentido, el estudio tiene como objetivo analizar las evidencias derivadas de la integración de la inteligencia artificial generativa en la enseñanza de la programación, mediante la formulación de prompts, la comparación entre diferentes herramientas de IA y la validación técnica del código, con el fin de fortalecer el pensamiento computacional en estudiantes de educación media técnica.

A partir de este propósito se plantea la siguiente pregunta de investigación:

¿Qué evidencias demuestran que la integración de la inteligencia artificial generativa, mediada por la formulación de prompts, la comparación entre herramientas de IA y la validación técnica del código, contribuye al fortalecimiento del pensamiento computacional en estudiantes de educación media técnica?

La integración de la inteligencia artificial generativa mediante estrategias de Prompt Engineering, comparación entre diferentes modelos de IA y validación técnica del código favorece el desarrollo del pensamiento computacional y fortalece las competencias en programación de bases de datos en estudiantes de educación media técnica.

Finalmente, este artículo aporta una perspectiva diferente a la presentada en el capítulo del libro, al centrarse en el análisis de los hallazgos empíricos obtenidos durante la intervención educativa. La contribución principal consiste en demostrar que el aprendizaje significativo en programación no depende exclusivamente de la capacidad de la inteligencia artificial para generar código, sino de la capacidad del estudiante para cuestionarlo, validarlo, corregirlo y mejorarlo mediante procesos de reflexión crítica y aplicación de buenas prácticas de programación.

El Objetivo General es analizar las evidencias derivadas de la integración de la inteligencia artificial generativa en la enseñanza de la programación, mediante la formulación de prompts, la comparación entre herramientas de IA y la validación técnica del código, para fortalecer el pensamiento computacional en estudiantes de educación media técnica.

Los Objetivos específicos son:

- Analizar la calidad de los prompts elaborados por los estudiantes durante la intervención didáctica.
- Comparar las respuestas generadas por ChatGPT, Gemini y Copilot para identificar fortalezas y limitaciones en la producción de código.
- Examinar el proceso de validación del código mediante SQLite Online como estrategia para fortalecer el aprendizaje de la programación.
- Identificar evidencias del desarrollo del pensamiento computacional derivadas de la integración de la inteligencia artificial generativa en el aula.

ESTADO DEL ARTE

La incorporación de la inteligencia artificial generativa en los procesos educativos ha generado un creciente interés en la comunidad científica debido a su potencial para transformar las estrategias de enseñanza y aprendizaje. Investigaciones recientes destacan que herramientas como ChatGPT, Gemini y Copilot favorecen la generación de contenidos, la resolución de problemas y el apoyo a procesos de programación; sin embargo, también advierten la necesidad de desarrollar competencias críticas que permitan validar las respuestas generadas por estos sistemas (Dwivedi et al., 2023; Walter, 2024).

En el ámbito de la enseñanza de la programación, diversos estudios evidencian que la inteligencia artificial puede facilitar la comprensión de conceptos, acelerar la generación de código y proporcionar retroalimentación inmediata. No obstante, Hsu (2025) señala que el verdadero valor educativo no radica únicamente en la producción automática de soluciones, sino en la capacidad del estudiante para formular instrucciones precisas (Prompt Engineering), analizar críticamente las respuestas obtenidas y comprender los fundamentos computacionales involucrados.

Por otra parte, Lo (2023) plantea que el diseño de prompts constituye una nueva competencia digital que requiere habilidades de comunicación, pensamiento crítico y estructuración lógica de problemas. Desde esta perspectiva, la alfabetización en inteligencia artificial implica formar estudiantes capaces de interactuar estratégicamente con modelos de lenguaje y evaluar la calidad de la información producida.

Las revisiones sistemáticas desarrolladas por Zawacki-Richter et al. (2019) muestran un incremento sostenido de investigaciones sobre inteligencia artificial en educación superior; sin embargo, identifican una limitada producción científica relacionada con la educación media y con experiencias que integren procesos de validación técnica del código generado por inteligencia artificial.

En consecuencia, se identifica un vacío de investigación relacionado con la escasa evidencia empírica sobre estrategias didácticas que articulen formulación de prompts, comparación entre diferentes herramientas de inteligencia artificial y validación técnica del código como mecanismos para fortalecer el pensamiento computacional en estudiantes de educación media técnica. La presente investigación busca contribuir a este campo mediante el análisis de una intervención educativa desarrollada en un contexto escolar colombiano.

Tabla 1

Antecedentes recientes sobre inteligencia artificial generativa, Prompt Engineering y pensamiento computacional.

Autor	Contexto	Hallazgo principal	Vacío identificado
Dwivedi et al. (2023)	IA generativa	Oportunidades y riesgos	No estudia programación escolar
Lo (2023)	Prompt Engineering	Calidad de los <i>prompts</i>	No valida código
Walter (2024)	IA y alfabetización	IA responsable	No analiza SQL
Hsu (2025)	Pensamiento computacional	Prompting como competencia	No compara modelos de IA

Nota. Elaboración Propia

Como se observa en la Tabla 1, las investigaciones recientes coinciden en reconocer el potencial de la inteligencia artificial generativa para apoyar los procesos de enseñanza y aprendizaje. Sin embargo, persisten vacíos relacionados con la validación técnica del código, la comparación sistemática entre diferentes modelos de inteligencia artificial y la generación de evidencias empíricas en educación media técnica. En este contexto, la presente investigación contribuye a la literatura al integrar la formulación de

prompts, la comparación entre ChatGPT, Gemini y Copilot, y la validación del código mediante SQLite Online como estrategia para fortalecer el pensamiento computacional en estudiantes de educación media técnica.

El análisis de los estudios revisados evidencia un consenso respecto al potencial de la inteligencia artificial generativa para transformar los procesos de enseñanza y aprendizaje. Dwivedi et al. (2023) destacan las oportunidades y desafíos derivados de la incorporación de modelos de lenguaje en diferentes contextos educativos y profesionales, mientras que Walter (2024) enfatiza la necesidad de desarrollar competencias relacionadas con la alfabetización en inteligencia artificial y el pensamiento crítico como condiciones indispensables para un uso responsable de estas tecnologías.

Por su parte, Lo (2023) centra su investigación en el Prompt Engineering, planteándolo como una competencia emergente para mejorar la interacción con modelos de lenguaje. En una línea complementaria, Hsu (2025) demuestra que la formulación de prompts favorece el desarrollo del pensamiento computacional cuando forma parte de estrategias pedagógicas estructuradas.

Aunque estos estudios coinciden en reconocer el valor educativo de la inteligencia artificial generativa, presentan diferencias metodológicas importantes. La mayoría corresponde a revisiones teóricas, estudios conceptuales o investigaciones desarrolladas en educación superior, con escasa evidencia empírica en educación media técnica. Asimismo, pocos trabajos integran procesos sistemáticos de comparación entre diferentes modelos de inteligencia artificial y validación técnica del código generado.

En este contexto, la presente investigación amplía la literatura existente al aportar evidencias obtenidas mediante una intervención educativa desarrollada en educación media técnica, incorporando una secuencia didáctica que articula Prompt Engineering, comparación entre ChatGPT, Gemini y Copilot y validación mediante SQLite Online para fortalecer el pensamiento computacional.

MARCO TEÓRICO

Inteligencia Artificial Generativa

La inteligencia artificial generativa (IAG) constituye uno de los desarrollos tecnológicos más relevantes de la última década debido a su capacidad para producir contenido original en múltiples formatos, incluyendo texto, imágenes, código fuente y recursos multimedia, a partir de instrucciones expresadas en lenguaje natural. Su funcionamiento se basa en modelos de lenguaje de gran escala (Large Language Models, LLM), entrenados con extensos volúmenes de información mediante técnicas de aprendizaje profundo, lo que les permite reconocer patrones lingüísticos y generar respuestas contextualizadas (Dwivedi et al., 2023).

En el ámbito educativo, la IAG ha transformado la interacción entre estudiantes, docentes y recursos digitales. Herramientas como ChatGPT, Gemini y Copilot facilitan la generación de ejemplos, explicaciones, algoritmos y soluciones de programación, reduciendo el tiempo necesario para desarrollar determinadas tareas y favoreciendo procesos de aprendizaje personalizados (Walter, 2024). No obstante, diversos autores advierten que estas tecnologías no garantizan por sí mismas aprendizajes significativos, ya que la calidad de las respuestas depende tanto de la precisión de las instrucciones suministradas como de la capacidad del usuario para interpretar, validar y contextualizar la información obtenida (Domínguez Figaredo & Stoyanovich, 2023).

En consecuencia, la incorporación de la inteligencia artificial generativa en educación requiere estrategias pedagógicas que trasciendan el uso instrumental de estas herramientas y promuevan el análisis crítico, la toma de decisiones fundamentadas y la construcción activa del conocimiento. En el contexto de la enseñanza de la programación, esto implica que el estudiante no solo genere código mediante IA, sino que también sea capaz de verificar su funcionamiento, identificar limitaciones y mejorarlo de acuerdo con principios de calidad del software.

Prompt Engineering

El Prompt Engineering se ha consolidado como una competencia emergente para interactuar eficazmente con los modelos de inteligencia artificial generativa. Consiste en el diseño intencional de instrucciones claras, específicas y contextualizadas que orientan el comportamiento del modelo y mejoran la calidad de las respuestas obtenidas (Lo, 2023). Más allá de formular preguntas, esta competencia implica estructurar solicitudes que definan objetivos, restricciones, contexto y criterios de calidad, favoreciendo respuestas más pertinentes y técnicamente consistentes.

En el ámbito educativo, el Prompt Engineering representa una oportunidad para fortalecer habilidades cognitivas superiores, ya que obliga a los estudiantes a analizar el problema, organizar la información relevante y anticipar el tipo de respuesta requerida. Desde esta perspectiva, la elaboración de un buen prompt constituye un proceso de razonamiento previo a la generación del código y no una simple interacción con una herramienta tecnológica.

Investigaciones recientes destacan que la enseñanza explícita del Prompt Engineering favorece el aprendizaje autónomo, el pensamiento crítico y la alfabetización digital, al permitir que los estudiantes comprendan las capacidades y limitaciones de los modelos de IA y adopten una postura crítica frente a las respuestas obtenidas (Walter, 2024; Hsu, 2025). En consecuencia, esta competencia comienza a ser considerada un componente esencial dentro de los procesos formativos asociados a la programación y al desarrollo de software.

Pensamiento Computacional

El pensamiento computacional ha sido reconocido como una competencia fundamental para la formación en ciencias de la computación y para la resolución de problemas en diversos contextos disciplinares. Wing (2006) lo define como un proceso intelectual que permite formular problemas y diseñar soluciones de manera que puedan ser ejecutadas por un ser humano o por una máquina. Este enfoque involucra habilidades como la descomposición de problemas, el reconocimiento de patrones, la abstracción y el diseño de algoritmos.

Posteriormente, diversos investigadores ampliaron esta perspectiva al considerar que el pensamiento computacional no se limita a aprender un lenguaje de programación, sino que comprende la capacidad para analizar situaciones complejas, evaluar alternativas y construir soluciones fundamentadas (Holmes et al., 2019). En este sentido, el desarrollo del pensamiento computacional requiere experiencias de aprendizaje donde los estudiantes argumenten decisiones, validen procedimientos y reflexionen sobre la calidad de las soluciones propuestas.

La incorporación de herramientas de inteligencia artificial generativa introduce nuevas posibilidades para fortalecer esta competencia. La generación automática de código constituye únicamente una etapa inicial del proceso de aprendizaje; el verdadero desarrollo del pensamiento computacional ocurre cuando el estudiante analiza críticamente las respuestas generadas, identifica errores, realiza ajustes y verifica su funcionamiento mediante procesos de validación técnica.

Alfabetización Digital e IA Responsable

La rápida expansión de la inteligencia artificial ha puesto de manifiesto la necesidad de fortalecer la alfabetización digital y promover un uso responsable de estas tecnologías. La alfabetización digital trasciende el dominio instrumental de herramientas informáticas e incorpora competencias relacionadas con la evaluación crítica de la información, la ética digital, la protección de datos y la toma de decisiones fundamentadas en contextos tecnológicos (Domínguez Figaredo & Stoyanovich, 2023).

En este escenario, la IA responsable implica comprender las limitaciones de los modelos generativos, reconocer posibles sesgos, verificar la veracidad de las respuestas y asumir una actitud ética frente a su utilización en procesos académicos y profesionales. Más que prohibir el uso de la inteligencia

artificial, la educación debe orientar a los estudiantes hacia prácticas que favorezcan la transparencia, la responsabilidad y la validación permanente de la información generada.

Desde esta perspectiva, la alfabetización digital y la IA responsable constituyen competencias transversales que complementan el desarrollo del pensamiento computacional y fortalecen la capacidad de los estudiantes para utilizar la inteligencia artificial como una herramienta de apoyo al aprendizaje y no como un sustituto del razonamiento humano.

Enseñanza de la programación mediada por inteligencia artificial

La enseñanza de la programación ha experimentado importantes transformaciones con la incorporación de herramientas de inteligencia artificial generativa. Tradicionalmente, el aprendizaje de la programación se centraba en la escritura manual de algoritmos y código fuente; sin embargo, la disponibilidad de asistentes inteligentes ha desplazado el énfasis hacia la comprensión, evaluación y mejora de soluciones generadas automáticamente.

Diversos estudios señalan que la inteligencia artificial puede facilitar la comprensión de conceptos, proporcionar retroalimentación inmediata y apoyar la resolución de problemas, siempre que exista una mediación pedagógica adecuada (Dwivedi et al., 2023; Walter, 2024). No obstante, la generación automática de código no garantiza que las soluciones sean correctas, eficientes o acordes con las buenas prácticas de desarrollo de software. Por ello, resulta indispensable que los estudiantes aprendan a comparar diferentes respuestas, identificar inconsistencias, validar el código mediante entornos de ejecución y aplicar criterios técnicos relacionados con la calidad del software.

En este contexto, la presente investigación propone una estrategia didáctica basada en la formulación de prompts, la comparación entre distintas herramientas de inteligencia artificial generativa y la validación del código en SQLite Online. Este enfoque busca transformar la interacción con la IA en un proceso de aprendizaje activo que favorezca la comprensión del código, el desarrollo del pensamiento computacional y la adopción de buenas prácticas de programación.

En conjunto, los referentes teóricos analizados sustentan la necesidad de integrar la inteligencia artificial generativa en la enseñanza de la programación desde una perspectiva pedagógica centrada en el análisis crítico y la validación técnica. Asimismo, proporcionan el marco conceptual para interpretar los resultados de la intervención educativa presentada en este estudio y comprender cómo la combinación de Prompt Engineering, pensamiento computacional, alfabetización digital e IA responsable puede contribuir al fortalecimiento de las competencias en programación en estudiantes de educación media técnica.

METODOLOGÍA

Enfoque de investigación

La presente investigación se desarrolló bajo un enfoque cualitativo con alcance descriptivo e interpretativo, complementado con elementos de análisis documental derivados de las evidencias producidas durante la intervención educativa. Este enfoque permitió comprender cómo la integración de herramientas de inteligencia artificial generativa influyó en el desarrollo del pensamiento computacional y en las competencias de programación de los estudiantes, a partir del análisis de sus producciones académicas, procesos de validación técnica y reflexiones sobre el uso de estas tecnologías.

El estudio se fundamentó en el análisis de evidencias generadas durante actividades auténticas de aula, privilegiando la interpretación de los procesos de aprendizaje sobre la medición de variables cuantitativas. La información fue obtenida mediante observación participante, revisión de documentos elaborados por los estudiantes, análisis comparativo del código generado por herramientas de inteligencia artificial y validación técnica utilizando SQLite Online.

Diseño de la investigación

Se adoptó un diseño de investigación aplicada con intervención educativa, orientado a integrar la inteligencia artificial generativa como recurso de mediación pedagógica en la enseñanza de la programación de bases de datos.

La intervención se estructuró alrededor de una secuencia didáctica que promovió el aprendizaje activo mediante cinco momentos: formulación de prompts, generación automática de código, comparación entre diferentes herramientas de inteligencia artificial, validación técnica mediante SQLite Online y reflexión crítica sobre la calidad de las soluciones obtenidas. Esta secuencia permitió observar cómo los estudiantes evolucionaron desde el uso instrumental de la IA hacia procesos de análisis, argumentación y mejora del código generado.

A diferencia de estudios centrados únicamente en las capacidades de las herramientas de inteligencia artificial, esta investigación situó el proceso de validación y reflexión como eje central del aprendizaje, permitiendo analizar la construcción del pensamiento computacional a partir de la interacción entre el estudiante, la IA y el entorno de programación.

Participantes y contexto

La investigación se desarrolló en el Colegio Técnico Palermo IED, institución oficial ubicada en Bogotá (Colombia), en el marco del área de Tecnología e Informática, específicamente en la asignatura Programación de Bases de Datos.

Participaron 28 estudiantes del curso 1001, pertenecientes al nivel de educación media técnica. Los participantes contaban con conocimientos previos sobre modelado entidad-relación, lenguaje SQL, diseño de bases de datos relacionales y principios básicos de programación, adquiridos durante las actividades desarrolladas en el currículo institucional.

La intervención se realizó durante las sesiones regulares de clase, utilizando equipos de cómputo con acceso a internet y herramientas de inteligencia artificial generativa de libre acceso. La participación de los estudiantes tuvo fines exclusivamente académicos y las evidencias utilizadas en la investigación fueron tratadas preservando la confidencialidad de la información personal. La Tabla 2 presenta las características generales de los participantes que hicieron parte de la intervención educativa.

Tabla 2

Caracterización de los participantes

Característica	Descripción
Institución educativa	Colegio Técnico Palermo IED
Nivel educativo	Educación Media Técnica
Curso	1001
Número de participantes	28 estudiantes
Área	Tecnología e Informática
Asignatura	Programación de Bases de Datos
Modalidad	Presencial
Herramientas utilizadas	ChatGPT, Gemini, Copilot y SQLite Online

Como se observa en la Tabla 2, la totalidad de los participantes pertenecía al mismo curso de educación media técnica, condición que permitió desarrollar una experiencia homogénea de aprendizaje y analizar en profundidad las evidencias derivadas de la integración de la inteligencia artificial generativa.

Instrumentos de recolección de información

Para obtener las evidencias de la investigación se emplearon diversos instrumentos que permitieron analizar tanto el proceso de interacción con la inteligencia artificial como la calidad de los productos elaborados por los estudiantes. Los principales instrumentos fueron:

Guía didáctica de la actividad, orientada a la formulación de prompts para programación de bases de datos mediante inteligencia artificial generativa.

Producciones académicas de los estudiantes, que incluyeron documentos comparativos, análisis de respuestas generadas por ChatGPT, Gemini y Copilot, así como reflexiones escritas sobre el proceso desarrollado.

Código SQL generado y validado, ejecutado en SQLite Online para verificar su funcionamiento, identificar errores y realizar ajustes.

Rúbrica de evaluación, diseñada para valorar aspectos relacionados con la calidad de los prompts, la aplicación de buenas prácticas de programación, la validación del código y la argumentación técnica.

Observación participante, realizada durante el desarrollo de las sesiones, registrando comportamientos, dificultades y estrategias utilizadas por los estudiantes.

La triangulación de estos instrumentos permitió fortalecer la confiabilidad de los resultados y contrastar la información obtenida desde diferentes fuentes de evidencia.

Actividad de intervención

La actividad central de la investigación consistió en la formulación de prompts para programación de bases de datos utilizando inteligencia artificial generativa, desarrollada durante las clases de Programación de Bases de Datos. Los estudiantes recibieron el siguiente reto:

"Diseñar una base de datos para el Colegio Técnico Palermo que incluyera las tablas Estudiante, Curso, Profesor, Asignatura, Grado, Acudiente, Tipo de Documento, Género y Calificaciones."

Cada estudiante formuló un prompt inicial y lo ejecutó en diferentes herramientas de inteligencia artificial generativa (ChatGPT, Gemini y Copilot). Posteriormente comparó las respuestas obtenidas, identificó diferencias en la estructura de las tablas, las relaciones entre entidades y las convenciones de programación utilizadas.

Como parte de la actividad, el código SQL generado fue ejecutado en SQLite Online, permitiendo validar su funcionamiento, detectar errores sintácticos y verificar aspectos relacionados con la integridad referencial, las claves primarias y foráneas, así como el cumplimiento de buenas prácticas de programación.

Finalmente, los estudiantes elaboraron un análisis crítico en el que argumentaron cuál de las respuestas resultó más adecuada y justificaron las modificaciones realizadas antes de aceptar el código como solución válida.

Procedimiento

La intervención educativa se desarrolló en cinco fases secuenciales, diseñadas para favorecer la construcción progresiva del pensamiento computacional.

Formulación del prompt. Los estudiantes diseñaron instrucciones orientadas a resolver un problema de programación de bases de datos, definiendo el contexto, las entidades requeridas y los criterios esperados para la respuesta.

Generación de respuestas mediante inteligencia artificial. El mismo prompt fue utilizado en ChatGPT, Gemini y Copilot con el propósito de obtener diferentes propuestas de solución.

Comparación y análisis crítico. Los estudiantes identificaron similitudes y diferencias entre las respuestas generadas por las herramientas, evaluando aspectos relacionados con la estructura del código, la organización del modelo de datos y el uso de convenciones de programación.

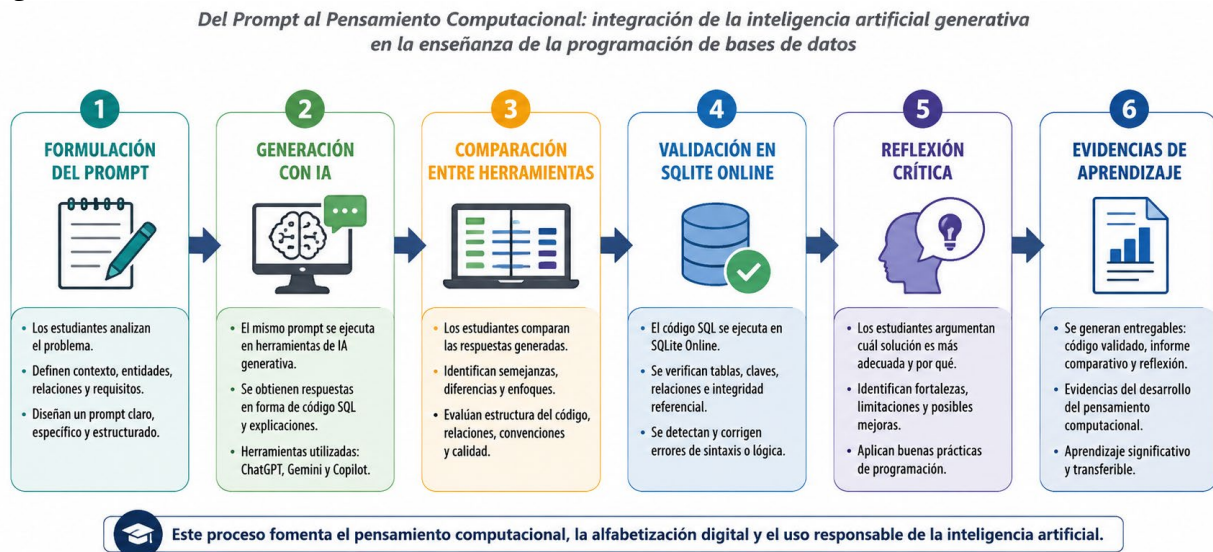
Validación técnica en SQLite Online. El código fue ejecutado en un entorno de base de datos para verificar su funcionamiento, corregir errores y comprobar la consistencia de las relaciones entre tablas.

Reflexión y argumentación. Finalmente, los estudiantes elaboraron un informe en el que justificaron las decisiones adoptadas durante el proceso, identificaron fortalezas y limitaciones de cada herramienta de inteligencia artificial y reflexionaron sobre la importancia de validar el código antes de su implementación.

Con el propósito de sintetizar la secuencia metodológica implementada durante la intervención educativa, la Figura 1 presenta las etapas desarrolladas desde la formulación de prompts hasta la validación del código generado mediante inteligencia artificial.

Figura 1

Secuencia metodológica de la integración de la inteligencia artificial generativa en la enseñanza de la programación.



Nota. Elaboración propia a partir de los resultados de la investigación. Diseño gráfico asistido mediante herramientas de inteligencia artificial.

Como se observa en la Figura 1, la estrategia didáctica se desarrolló en cinco etapas articuladas que permitieron integrar la inteligencia artificial generativa con actividades de análisis, validación técnica y reflexión crítica. Esta secuencia favoreció un aprendizaje progresivo y permitió que los estudiantes comprendieran la importancia de verificar las respuestas producidas por los modelos de lenguaje antes de utilizarlas en sus proyectos de programación.

Proceso de análisis de la información

El análisis se desarrolló mediante un proceso de codificación temática inspirado en Braun y Clarke (2006). Inicialmente se efectuó una lectura exhaustiva de todas las evidencias recolectadas, identificando

unidades de significado relacionadas con la formulación de prompts, la comparación entre herramientas de inteligencia artificial, la validación del código y el desarrollo del pensamiento computacional.

Posteriormente estas unidades fueron agrupadas en categorías emergentes que permitieron interpretar las relaciones entre las diferentes evidencias. Con el propósito de fortalecer la credibilidad de los resultados se aplicó un proceso de triangulación utilizando tres fuentes de información: productos desarrollados por los estudiantes; observaciones registradas durante la intervención; registros audiovisuales y validaciones realizadas mediante SQLite Online.

La convergencia entre estas fuentes permitió corroborar la consistencia de los hallazgos.

Como parte del proceso de documentación de la intervención educativa se recopilaron evidencias digitales mediante registros audiovisuales del desarrollo de las actividades en el aula. Estos registros permitieron documentar la interacción de los estudiantes con herramientas de inteligencia artificial generativa, el proceso de formulación de prompts, la comparación entre las respuestas obtenidas y la validación del código en SQLite Online. Uno de estos registros se encuentra disponible como material suplementario del estudio (Sánchez Barragán, 2026).

Construcción y validación de la rúbrica

La rúbrica utilizada para valorar las producciones de los estudiantes fue diseñada a partir de los resultados de aprendizaje establecidos para la asignatura y de las buenas prácticas de programación abordadas durante la intervención. Los criterios de evaluación incluyeron: calidad del prompt, estructura del código generado, aplicación de buenas prácticas de programación, validación técnica mediante SQLite Online y argumentación de las decisiones adoptadas durante el proceso de análisis.

Cada criterio fue valorado en cuatro niveles de desempeño (Inicial, Básico, Alto y Superior), permitiendo realizar una evaluación cualitativa del proceso de aprendizaje más allá de la funcionalidad del código.

Previamente a su aplicación, la rúbrica fue revisada por docentes con experiencia en programación y en integración educativa de tecnologías digitales para verificar la claridad de los criterios y su correspondencia con los objetivos de la investigación. Las observaciones realizadas permitieron ajustar la redacción de algunos indicadores antes de su utilización en el aula.

La construcción de la rúbrica se fundamentó en los resultados de aprendizaje de la asignatura y en referentes sobre evaluación auténtica y enseñanza de la programación. Los criterios fueron definidos previamente a la intervención y posteriormente revisados por docentes con experiencia en programación e integración educativa de tecnologías digitales, fortaleciendo su validez de contenido.

RESULTADOS

Los resultados que se presentan a continuación corresponden a la implementación de la estrategia didáctica desarrollada con estudiantes del curso 1001 del Colegio Técnico Palermo IED, orientada a integrar herramientas de inteligencia artificial generativa en la enseñanza de la programación de bases de datos. Las evidencias fueron obtenidas a partir del análisis de los prompts elaborados por los estudiantes, la comparación entre las respuestas generadas por ChatGPT, Gemini y Copilot, la validación técnica del código mediante SQLite Online y las reflexiones consignadas en los productos académicos desarrollados durante la intervención.

En conjunto, los resultados muestran que la inteligencia artificial generativa constituyó un recurso de apoyo para el aprendizaje cuando estuvo acompañada por procesos de mediación pedagógica, validación técnica y análisis crítico. Asimismo, evidencian que la generación automática de código no sustituyó el razonamiento de los estudiantes, sino que promovió procesos de revisión, argumentación y mejora de las soluciones propuestas.

Calidad de los prompts

Uno de los principales hallazgos de la investigación fue la evolución observada en la formulación de prompts. En una primera aproximación, la mayoría de los estudiantes elaboró instrucciones generales, limitándose a solicitar la creación de una base de datos sin especificar el contexto, las entidades o las relaciones requeridas. Estas solicitudes produjeron respuestas incompletas o poco adaptadas a las necesidades del ejercicio.

A medida que avanzó la intervención, los estudiantes incorporaron elementos propios del Prompt Engineering, definiendo con mayor precisión el contexto institucional, las tablas necesarias, las relaciones entre entidades, las restricciones de integridad y el motor de base de datos utilizado. Esta evolución permitió obtener respuestas más completas y técnicamente consistentes.

Los trabajos elaborados por estudiantes como Isabella, Paula y Kawakami evidenciaron una mejora progresiva en la calidad de los prompts, reflejada en instrucciones más estructuradas y contextualizadas, lo que facilitó la generación de modelos de datos más cercanos a los requerimientos planteados. Para analizar la evolución de la formulación de prompts, se sistematizaron las producciones realizadas por los estudiantes durante la intervención.

Tabla 3

Evolución de la calidad de los prompts

Aspecto	Prompt inicial	Prompt mejorado
Contexto	General	Específico (Colegio Técnico Palermo)
Entidades	Parciales	Completas
Relaciones	No especificadas	Definidas explícitamente
Restricciones	Ausentes	Incluidas
Resultado obtenido	Incompleto	Más preciso y contextualizado

La información presentada en la Tabla 3 evidencia una mejora progresiva en la calidad de los prompts, caracterizada por instrucciones más específicas, mayor contextualización y una mejor definición de los requerimientos técnicos. Estos resultados evidencian que la formulación de prompts constituye una competencia que puede desarrollarse mediante procesos de mediación pedagógica y práctica sistemática.

Comparación entre ChatGPT, Gemini y Copilot

La comparación entre las respuestas generadas por diferentes herramientas de inteligencia artificial permitió a los estudiantes identificar fortalezas, limitaciones y diferencias en la calidad del código propuesto.

En términos generales, las tres herramientas generaron estructuras funcionales para el diseño de la base de datos; sin embargo, se observaron diferencias en la organización de las tablas, la definición de relaciones y el nivel de detalle de las explicaciones. Asimismo, ninguna de las herramientas aplicó de manera consistente las convenciones de nomenclatura trabajadas durante las clases, como PascalCase o UpperCamelCase, lo que obligó a los estudiantes a revisar y ajustar el código antes de considerarlo una solución válida.

La comparación favoreció el desarrollo del pensamiento crítico, ya que los estudiantes dejaron de asumir que la primera respuesta generada era necesariamente la mejor opción y comenzaron a justificar técnicamente la selección de una propuesta sobre otra.

Tabla 4

Comparación entre herramientas de inteligencia artificial

Criterio	ChatGPT	Gemini	Copilot
Organización de tablas	Alta	Muy alta	Alta
Explicación del modelo	Buena	Excelente	Buena
Compatibilidad con SQLite	Parcial	Parcial	Parcial
Uso consistente de PascalCase	No	No	No
Claridad de la respuesta	Alta	Muy alta	Alta

Los resultados muestran diferencias en la estructura y precisión de las respuestas generadas por ChatGPT, Gemini y Copilot, lo que permitió a los estudiantes comparar las ventajas y limitaciones de cada herramienta antes de validar técnicamente el código. Los resultados indican que el valor pedagógico de la inteligencia artificial no reside únicamente en la calidad de la respuesta generada, sino en el proceso de comparación, análisis y argumentación realizado por los estudiantes.

Las evidencias obtenidas durante la intervención muestran la participación activa de los estudiantes en la comparación de las respuestas generadas por ChatGPT, Gemini y Copilot, así como en la validación crítica del código. Este proceso puede observarse en el registro audiovisual de la experiencia (Sánchez Barragán, 2026).

Validación en SQLite Online

La validación técnica del código constituyó uno de los momentos más relevantes de la intervención. Después de obtener las respuestas generadas por las herramientas de inteligencia artificial, los estudiantes ejecutaron el código SQL en SQLite Online con el propósito de verificar su funcionamiento.

Durante este proceso se identificaron errores relacionados con la sintaxis, la definición de claves foráneas, la integridad referencial y la compatibilidad con el motor de base de datos utilizado. La detección de estas inconsistencias permitió a los estudiantes comprender que una respuesta aparentemente correcta puede requerir ajustes antes de ser implementada.

Las capturas de pantalla obtenidas durante la ejecución del código muestran que los estudiantes realizaron procesos de depuración y corrección antes de alcanzar una versión funcional de la base de datos.

Con el propósito de analizar las diferencias entre las respuestas generadas por los distintos modelos de inteligencia artificial, se realizó una comparación utilizando un mismo prompt relacionado con la creación de bases de datos.

Figura 2
Flujo de validación del código generado por inteligencia artificial



Nota. Elaboración propia a partir de los resultados de la investigación. Diseño gráfico asistido mediante herramientas de inteligencia artificial.

La Figura 2 evidencia que, aunque las tres herramientas produjeron soluciones funcionales, ninguna aplicó de manera consistente todas las buenas prácticas trabajadas durante la intervención, especialmente en aspectos relacionados con la nomenclatura, las claves primarias y la organización del código. Este resultado reforzó la importancia de la validación crítica por parte de los estudiantes. Este proceso fortaleció la comprensión del lenguaje SQL y promovió una actitud crítica frente a las respuestas generadas por la inteligencia artificial.

Aplicación de buenas prácticas de programación

Otro hallazgo relevante fue la aplicación consciente de buenas prácticas de programación por parte de los estudiantes. Durante la intervención se reforzó el uso de convenciones de nomenclatura como PascalCase y UpperCamelCase, así como la definición adecuada de claves primarias, claves foráneas e integridad referencial.

Las evidencias muestran que los estudiantes identificaron diferencias entre el código generado por la inteligencia artificial y las convenciones enseñadas en clase. En consecuencia, realizaron modificaciones orientadas a mejorar la legibilidad, consistencia y mantenimiento del código.

Este resultado evidencia que el conocimiento disciplinar adquirido previamente permitió a los estudiantes evaluar críticamente las respuestas generadas por la IA y no limitarse a reproducirlas de forma automática.

Evidencias del pensamiento computacional

Las producciones académicas elaboradas por los estudiantes permiten evidenciar el desarrollo de habilidades propias del pensamiento computacional. En los documentos analizados se identificaron procesos de descomposición del problema, reconocimiento de patrones, análisis de alternativas, formulación de argumentos y toma de decisiones fundamentadas.

Asimismo, las reflexiones finales muestran que los estudiantes comprendieron la importancia de validar el código generado por inteligencia artificial antes de utilizarlo, reconociendo que las herramientas

de IA constituyen un apoyo para el aprendizaje, pero no reemplazan la necesidad de comprender la lógica de programación ni de aplicar criterios técnicos para garantizar la calidad del software.

Las evidencias audiovisuales obtenidas durante las sesiones de clase complementan estos hallazgos al mostrar la interacción de los estudiantes con las herramientas de inteligencia artificial, las discusiones generadas durante la comparación de respuestas y el proceso colaborativo de validación del código.

Síntesis de hallazgos

Los resultados obtenidos permiten afirmar que la estrategia didáctica favoreció el fortalecimiento del pensamiento computacional mediante la integración de inteligencia artificial generativa y procesos de validación técnica.

Tabla 5

Síntesis de los principales hallazgos

Hallazgo	Evidencia
Mejora en la formulación de <i>prompts</i>	Prompts progresivamente más específicos y contextualizados.
Comparación crítica entre herramientas	Identificación de fortalezas y limitaciones de ChatGPT, Gemini y Copilot.
Validación técnica del código	Corrección de errores mediante SQLite Online.
Aplicación de buenas prácticas	Uso de PascalCase, claves primarias y foráneas e integridad referencial.
Desarrollo del pensamiento computacional	Argumentación, análisis crítico y mejora del código generado.

La Tabla 5 resume las principales evidencias observadas durante el proceso de validación mediante SQLite Online, destacando la identificación de errores de sintaxis, inconsistencias en las relaciones entre tablas y oportunidades de mejora en las consultas SQL.

En conjunto, estos resultados muestran que el aprendizaje no se produjo por la utilización aislada de herramientas de inteligencia artificial, sino por la combinación de Prompt Engineering, comparación entre diferentes modelos, validación técnica del código y reflexión crítica. Esta secuencia permitió transformar la inteligencia artificial generativa en un recurso de mediación pedagógica que favoreció el desarrollo de competencias en programación y pensamiento computacional en estudiantes de educación media técnica.

DISCUSIÓN

Los resultados obtenidos durante esta investigación permiten afirmar que la incorporación de la inteligencia artificial generativa en la enseñanza de la programación va mucho más allá de utilizar una herramienta para obtener respuestas rápidas. La experiencia desarrollada en el aula mostró que el verdadero aprendizaje se produjo cuando los estudiantes comenzaron a cuestionar, comparar y validar las respuestas generadas por la inteligencia artificial, comprendiendo que estas herramientas constituyen un apoyo para el aprendizaje, pero no sustituyen el razonamiento propio del proceso de programación.

Uno de los aspectos que más llamó la atención durante la intervención fue que, en la mayoría de las actividades, ChatGPT generó respuestas más completas y estructuradas que Gemini y Copilot. Sin embargo, este hallazgo no significa que sus respuestas fueran correctas en todos los casos. Al validar el código mediante SQLite Online, los estudiantes identificaron errores relacionados con la definición de claves, la

organización de las tablas, la nomenclatura de los atributos y la aplicación de buenas prácticas de programación. Esta situación permitió comprender que una respuesta aparentemente correcta debe ser analizada desde criterios técnicos antes de ser aceptada, aspecto que coincide con lo planteado por Dwivedi et al. (2023) sobre la necesidad de utilizar la inteligencia artificial desde una perspectiva crítica y responsable.

Por otra parte, Gemini presentó respuestas más variables, especialmente cuando los estudiantes formularon prompts poco específicos. En varias oportunidades las soluciones resultaron incompletas o requerían ajustes importantes para poder ejecutarse correctamente. Lejos de considerar esto como una desventaja de la herramienta, la experiencia permitió evidenciar que la calidad de la respuesta depende, en gran medida, de la capacidad del estudiante para comunicar con claridad lo que necesita. En este sentido, los resultados respaldan lo planteado por Lo (2023), quien señala que el Prompt Engineering constituye una competencia emergente que debe desarrollarse de manera intencionada en los procesos educativos.

Otro hallazgo significativo fue comprobar que ninguna de las herramientas evaluadas respondió exactamente como se esperaba desde el punto de vista académico. Aunque las tres ofrecieron propuestas funcionales, ninguna aplicó de forma consistente las convenciones de programación trabajadas en clase, como el uso de PascalCase, la correcta definición de claves primarias y foráneas o la organización adecuada de las relaciones entre tablas. Precisamente esta situación convirtió la inteligencia artificial en una oportunidad de aprendizaje, ya que los estudiantes dejaron de limitarse a copiar código y comenzaron a analizar, justificar y corregir las respuestas obtenidas. Desde mi perspectiva, este fue uno de los logros más importantes de la intervención, porque el centro del proceso dejó de ser la herramienta y pasó a ser el razonamiento del estudiante.

Estos resultados también permiten reflexionar sobre el desarrollo del pensamiento computacional. A lo largo de la experiencia fue evidente que los estudiantes no solo mejoraron en la formulación de prompts, sino también en la forma de abordar los problemas de programación. La necesidad de comparar diferentes respuestas, identificar errores y validar el código favoreció procesos de análisis, descomposición de problemas y toma de decisiones que son propios del pensamiento computacional descrito por Wing (2006). En consecuencia, la inteligencia artificial no reemplazó estos procesos cognitivos, sino que generó nuevas oportunidades para fortalecerlos mediante actividades de reflexión y validación.

Las evidencias obtenidas plantean igualmente desafíos para la práctica docente. La incorporación de la inteligencia artificial en el aula exige un cambio en el rol del profesor, quien deja de ser únicamente transmisor de contenidos para convertirse en un mediador que orienta a los estudiantes en la formulación de preguntas, el análisis crítico de las respuestas y la validación de las soluciones propuestas por la inteligencia artificial. En este contexto, el dominio técnico de las herramientas resulta importante, pero aún más relevante es la capacidad de diseñar experiencias de aprendizaje que promuevan el pensamiento crítico, la autonomía y el uso ético de estas tecnologías, tal como lo plantea Walter (2024).

Desde la enseñanza de la programación, los resultados invitan a replantear algunas prácticas tradicionales de evaluación. Durante muchos años el énfasis ha estado en comprobar si un programa funciona correctamente; sin embargo, la disponibilidad de herramientas capaces de generar código en pocos segundos hace necesario valorar otras competencias, como la capacidad para interpretar, validar, argumentar y mejorar las soluciones propuestas por la inteligencia artificial. En otras palabras, hoy resulta tan importante comprender por qué un código funciona como saber escribirlo.

Si bien los resultados obtenidos muestran evidencias favorables sobre la integración de la inteligencia artificial generativa en la enseñanza de la programación, es importante interpretar estos hallazgos considerando el alcance del estudio. La investigación se desarrolló con un único grupo de estudiantes y desde un enfoque cualitativo, por lo que las conclusiones no pretenden establecer generalizaciones estadísticas. No obstante, la riqueza de las evidencias recopiladas y la triangulación de la

información permiten ofrecer una comprensión profunda del fenómeno estudiado y constituyen una base sólida para futuras investigaciones en otros contextos educativos.

Finalmente, considero que uno de los principales aportes de esta investigación consiste en mostrar que la inteligencia artificial generativa puede convertirse en un recurso pedagógico valioso cuando se integra dentro de una estrategia didáctica claramente planificada. La combinación de Prompt Engineering, comparación entre ChatGPT, Gemini y Copilot y validación técnica mediante SQLite Online permitió que los estudiantes desarrollaran una actitud crítica frente a las respuestas generadas por la inteligencia artificial y fortalecieran sus competencias en programación desde una perspectiva reflexiva. Más que identificar cuál herramienta produce mejores respuestas, la experiencia demuestra que el verdadero potencial educativo de la inteligencia artificial reside en la capacidad del docente para convertir esas respuestas en oportunidades de aprendizaje, análisis y construcción de conocimiento.

Tabla 6

Relación entre los hallazgos del estudio y la literatura científica

Hallazgo del estudio	Evidencia empírica	Referente teórico	Contribución del estudio
Mejora en la formulación de prompts	Producciones de los estudiantes	Lo (2023); Hsu (2025)	El Prompt Engineering puede enseñarse y mejorar mediante mediación pedagógica.
Comparación entre IA	Análisis de ChatGPT, Gemini y Copilot	Dwivedi et al. (2023)	La comparación favorece el pensamiento crítico frente a las respuestas de la IA.
Validación del código	Ejecución en SQLite Online	Walter (2024)	La validación técnica fortalece la autonomía y la comprensión del código.
Desarrollo del pensamiento computacional	Análisis de corrección del código	Wing (2006); Holmes et al. (2019)	La IA puede potenciar el pensamiento computacional cuando existe mediación docente.
IA responsable	Reflexiones de los estudiantes	Domínguez Figaredo y Stoyanovich (2023); Zawacki-Richter et al. (2019)	La alfabetización en IA requiere procesos de evaluación crítica y uso ético.

La Tabla 6 sintetiza la relación entre los hallazgos obtenidos y los antecedentes científicos revisados, mostrando cómo la presente investigación amplía la evidencia existente al incorporar la comparación entre herramientas de inteligencia artificial y la validación técnica del código como estrategias para fortalecer el pensamiento computacional.

Desde una perspectiva científica, esta investigación aporta evidencia empírica sobre la integración de la inteligencia artificial generativa en educación media técnica mediante una estrategia que combina Prompt Engineering, comparación entre diferentes modelos de IA y validación técnica del código. A diferencia de estudios centrados únicamente en el potencial de estas herramientas, el presente trabajo demuestra que el aprendizaje significativo se fortalece cuando la inteligencia artificial se incorpora como objeto de análisis y reflexión dentro del proceso pedagógico, ampliando así el conocimiento disponible sobre el desarrollo del pensamiento computacional en contextos escolares.

CONCLUSIONES

Los resultados obtenidos permiten concluir que la integración de la inteligencia artificial generativa en la enseñanza de la programación constituye una estrategia pedagógica efectiva cuando está acompañada por procesos de mediación docente, formulación de prompts, comparación entre herramientas de inteligencia artificial y validación técnica del código. Las evidencias recopiladas durante la intervención demuestran que el aprendizaje significativo no depende de la capacidad de la IA para generar respuestas, sino de la capacidad del estudiante para analizarlas, cuestionarlas y mejorarlas a partir de criterios técnicos y pedagógicos.

En primer lugar, la investigación evidenció que el Prompt Engineering puede desarrollarse como una competencia educativa. La evolución observada en la formulación de prompts permitió a los estudiantes generar instrucciones progresivamente más precisas y contextualizadas, obteniendo respuestas de mayor calidad. Este hallazgo confirma que la interacción con modelos de inteligencia artificial requiere habilidades de análisis, planificación y comunicación que pueden fortalecerse mediante estrategias didácticas intencionales.

En segundo lugar, la comparación entre ChatGPT, Gemini y Copilot mostró que ninguna herramienta produjo respuestas completamente ajustadas a las buenas prácticas de programación trabajadas en el aula. La necesidad de revisar aspectos relacionados con la nomenclatura mediante PascalCase o UpperCamelCase, la definición de claves primarias y foráneas, la integridad referencial y la compatibilidad con SQLite Online puso de manifiesto que la inteligencia artificial debe entenderse como un recurso de apoyo al aprendizaje y no como un sustituto del conocimiento disciplinar ni del juicio técnico del estudiante.

En tercer lugar, la validación del código mediante SQLite Online constituyó un elemento decisivo para el fortalecimiento del pensamiento computacional. La ejecución del código permitió identificar errores, realizar procesos de depuración y justificar técnicamente las modificaciones efectuadas antes de obtener una solución funcional. Estas actividades favorecieron el desarrollo de competencias relacionadas con el análisis, la resolución de problemas, la argumentación y la toma de decisiones fundamentadas, aspectos esenciales para la formación en programación.

Finalmente, la principal contribución de esta investigación consiste en aportar evidencia empírica sobre la manera en que la integración de la inteligencia artificial generativa puede transformar los procesos de enseñanza y aprendizaje cuando se articula con estrategias pedagógicas que promueven la reflexión crítica y la validación técnica. Más que demostrar la eficacia de una herramienta tecnológica específica, el estudio evidencia que la combinación de Prompt Engineering, comparación entre modelos de IA, validación del código y aplicación de buenas prácticas favorece el desarrollo del pensamiento computacional y fortalece las competencias en programación de bases de datos en estudiantes de educación media técnica.

En conjunto, estos hallazgos respaldan la necesidad de replantear la enseñanza de la programación desde un enfoque en el que la inteligencia artificial generativa se incorpore como un recurso para potenciar el aprendizaje activo, la alfabetización digital y la construcción de conocimiento, reafirmando el papel del docente como mediador del proceso formativo y orientador del uso crítico, ético y responsable de estas tecnologías. Esta investigación contribuye así a ampliar la evidencia disponible sobre la integración pedagógica de la inteligencia artificial en educación y ofrece elementos que pueden orientar futuras experiencias de innovación curricular e investigación aplicada en programación y tecnologías educativas.

LIMITACIONES DEL ESTUDIO

La presente investigación presenta algunas limitaciones que deben considerarse al interpretar sus resultados. En primer lugar, la intervención se desarrolló con un único grupo de estudiantes del curso 1001 del Colegio Técnico Palermo IED, perteneciente a la educación media técnica, por lo que los hallazgos

corresponden a un contexto educativo específico y no pueden generalizarse de manera directa a otros niveles de formación o instituciones con características diferentes.

En segundo lugar, la validación técnica del código se realizó exclusivamente mediante SQLite Online, herramienta seleccionada por su facilidad de acceso y pertinencia para el proceso de enseñanza. Aunque este entorno permitió verificar la sintaxis y el funcionamiento del código SQL, futuras investigaciones podrían ampliar el análisis incorporando otros sistemas gestores de bases de datos utilizados en el ámbito profesional, como MySQL, PostgreSQL, SQL Server u Oracle, con el fin de comparar el comportamiento de las soluciones generadas por la inteligencia artificial en diferentes entornos tecnológicos.

Otra limitación corresponde al carácter cualitativo de la investigación. Si bien las evidencias obtenidas —documentos elaborados por los estudiantes, comparación entre herramientas de inteligencia artificial, validación técnica del código, observaciones de aula y producciones académicas— permitieron comprender en profundidad el proceso de aprendizaje, no se aplicaron instrumentos estandarizados para medir cuantitativamente el desarrollo del pensamiento computacional antes y después de la intervención.

Finalmente, la investigación se centró en actividades relacionadas con la programación de bases de datos y el lenguaje SQL, por lo que sus resultados deben interpretarse dentro de este dominio disciplinar. No obstante, los principios didácticos identificados podrían servir como referencia para futuras experiencias en otros lenguajes de programación y áreas de la informática.

LÍNEAS FUTURAS DE INVESTIGACIÓN

Los resultados obtenidos abren diversas oportunidades para ampliar el conocimiento sobre la integración de la inteligencia artificial generativa en la enseñanza de la programación. En primer lugar, se propone replicar esta investigación en diferentes contextos educativos, incluyendo programas de educación superior, formación técnica y tecnológica, con el propósito de comparar el comportamiento del modelo didáctico en poblaciones con distintos niveles de experiencia en programación.

Asimismo, resulta pertinente analizar la aplicación de la estrategia en otros lenguajes y paradigmas de programación, incorporando tecnologías como Python, Java, C#, desarrollo web y programación orientada a objetos, así como distintos sistemas gestores de bases de datos. Esto permitiría establecer si los hallazgos observados son consistentes en escenarios tecnológicos diversos.

Otra línea de investigación consiste en evaluar el impacto de asistentes inteligentes especializados en programación y agentes de inteligencia artificial capaces de ofrecer retroalimentación personalizada durante el proceso de desarrollo de software. La comparación entre diferentes modelos de lenguaje y nuevas generaciones de herramientas de IA permitirá comprender cómo evoluciona su aporte al aprendizaje y cuáles son las competencias docentes necesarias para integrarlas de manera efectiva en el aula.

De igual manera, se recomienda desarrollar estudios longitudinales que permitan analizar la evolución del pensamiento computacional y de las competencias en programación a lo largo del tiempo, identificando si los aprendizajes promovidos mediante la formulación de prompts, la validación crítica del código y la aplicación de buenas prácticas se mantienen y transfieren a contextos académicos y profesionales posteriores.

Finalmente, futuras investigaciones podrían incorporar metodologías mixtas que combinen análisis cualitativos y cuantitativos, integrando instrumentos estandarizados para evaluar el pensamiento computacional, la alfabetización en inteligencia artificial y el desarrollo de competencias digitales. Este enfoque permitiría fortalecer la evidencia empírica disponible y contribuir a la consolidación de modelos pedagógicos fundamentados en el uso crítico, ético y responsable de la inteligencia artificial generativa.

REFERENCIAS

- Domínguez Figaredo, D., & Stoyanovich, J. (2023). Responsible AI literacy: A stakeholder-first approach. *Big Data & Society*, 10(2). <https://doi.org/10.1177/20539517231219958>
- Dwivedi, Y. K., Hughes, L., Baabdullah, A. M., Ribeiro-Navarrete, S., Giannakis, M., Al-Debei, M. M., Dennehy, D., Metri, B., Buhalis, D., Cheung, C. M. K., Conboy, K., Doyle, R., Dubey, R., Dutot, V., Felix, R., Goyal, D. P., Gustafsson, A., Hinsch, C., Jebabli, I., ... Wright, R. (2023). *International Journal of Information Management*, 71, 102642. <https://doi.org/10.1016/j.ijinfomgt.2023.102642>
- Holmes, W., Bialik, M., & Fadel, C. (2019). *Artificial intelligence in education: Promises and implications for teaching and learning*. Center for Curriculum Redesign.
- Hsu, H.-P. (2025). From programming to prompting: Developing computational thinking through large language model-based generative artificial intelligence. *TechTrends*, 69, 485–506. <https://doi.org/10.1007/s11528-025-01052-6>
- Lo, L. S. (2023). The CLEAR path: A framework for enhancing information literacy through prompt engineering. *The Journal of Academic Librarianship*, 49(4), 102720. <https://doi.org/10.1016/j.acalib.2023.102720>
- Luckin, R., Holmes, W., Griffiths, M., & Forcier, L. B. (2016). *Intelligence unleashed: An argument for AI in education*. Pearson.
- Walter, Y. (2024). Embracing the future of artificial intelligence in the classroom: The relevance of AI literacy, prompt engineering, and critical thinking in modern education. *International Journal of Educational Technology in Higher Education*, 21, Article 15. <https://doi.org/10.1186/s41239-024-00448-3>
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35. <https://doi.org/10.1145/1118178.1118215>
- Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education: Where are the educators? *International Journal of Educational Technology in Higher Education*, 16, Article 39. <https://doi.org/10.1186/s41239-019-0171-0>

MATERIAL SUPLEMENTARIO

- Sánchez Barragán, S. E. (2026, julio 15). Integración de la inteligencia artificial generativa en la enseñanza de la programación: Evidencia de la intervención educativa [Video]. YouTube. https://youtu.be/v5P-jqtS_1M