

Recursos instruccionales y aprendizaje de la programación en Java: una experiencia basada en el diseño

Instructional Resources and Java Programming Learning: A Design-Based Experience

Darly José Juárez Lara

Facultad de Contaduría y Administración
zS21016023@estudiantes.uv.mx
<https://orcid.org/0009-0009-2498-1935>

Patricia Arieta Melgarejo

Facultad de Contaduría y Administración
parieta@uv.mx
<https://orcid.org/0009-0000-9721-7350>

Martha Elba Ruíz Libreros

Facultad de Pedagogía
maruiz@uv.mx
<https://orcid.org/0000-0003-1529-9159>

Suzel Gómez Jiménez

Facultad de Contaduría y Administración
suzgomez@uv.mx
<https://orcid.org/0009-0001-9308-2441>

Universidad Veracruzana
Xalapa, Veracruz, México.

Resumen

El artículo examina el impacto de los recursos instruccionales digitales en el aprendizaje de los estudiantes sobre los Fundamentos de Programación, misma que representa el mayor índice de reprobación en la Licenciatura en Sistemas Computacionales Administrativos. El objetivo fue analizar los recursos instruccionales que favorecen el rendimiento académico y desarrollo de habilidades prácticas. La investigación por tipo de datos recolectados es cuantitativa con diseño cuasi-experimental, su alcance es descriptivo, con una fuente de recolección de datos observacional y un plan de recolección de datos experimental y recolección transversal. La población estuvo integrada por 129 estudiantes, con una muestra de 98 participantes. Se aplicaron cuestionarios pretest y posttest, validados mediante juicio de expertos y prueba piloto. Los datos se analizaron mediante estadística descriptiva e inferencial, donde los resultados muestran una mejora en la dimensión heurística del rendimiento académico. Se concluye que los recursos instruccionales digitales fortalecen el aprendizaje práctico en programación.

Palabras clave: autoaprendizaje; computación; recursos; tecnología.

Abstract

The article examines the impact of digital instructional resources on students' learning in the course Fundamentals of Programming, which presents the highest failure rate in the Bachelor's Degree in Administrative Computer Systems. The objective was to analyze the instructional resources that enhance academic performance and the development of practical skills. The study follows a quantitative approach with a quasi-experimental design and a descriptive scope, using an observational data collection source and a cross-sectional data collection plan. The population consisted of 129 students, with a sample of 98 participants. Pretest and posttest questionnaires were applied, validated through expert judgment and a pilot test. Data were analyzed using descriptive and inferential statistics, and the results show an improvement in the heuristic dimension of academic performance. It is concluded that digital instructional resources strengthen practical learning in programming.

Keywords: self-learning; computing; resources; technology.

Introducción

Problema de investigación

En la actualidad, es un desafío importante la enseñanza de la programación en los programas universitarios relacionados con las tecnologías de la información, para estudiantes que viven en la inmediatez y en un mundo de grandes distractores. La complejidad de los lenguajes de programación, junto con las limitadas habilidades de razonamiento lógico de los estudiantes, se traduce en elevados índices de reprobación en asignaturas básicas (Pires et al., 2024).

En este contexto, la experiencia educativa (EE) de Fundamentos de Programación de la Licenciatura en Sistemas Computacionales Administrativos de la Universidad Veracruzana, presenta el mayor índice de reprobación, lo que evidencia la necesidad de implementar estrategias instruccionales innovadoras que favorezcan la comprensión y la práctica de los conceptos básicos de la programación. Lo que llevo a plantear la interrogante de: ¿Los recursos instruccionales digitales favorecen el rendimiento académico relacionado con las dimensiones heurística, teórica y axiológica de los estudiantes de la EE de Fundamentos de Programación de la Licenciatura en Sistemas Computacionales Administrativos de la Universidad Veracruzana, durante el periodo agosto 2024–enero 2025?

Muchas investigaciones han demostrado que los recursos instruccionales digitales acompañan y brindan las facilidades a los estudiantes, para un aprendizaje autónomo y fortalecido con ambientes intuitivos que permitan una interacción constante con contenidos significativos (Cabero-Almenara et al., 2021). Estos recursos integran elementos multimedia, simuladores y entornos de práctica que promueven la aplicación de la lógica algorítmica y el razonamiento computacional (Imbaquingo & Romero, 2025). Sin embargo, la evidencia empírica sobre su impacto real en el rendimiento académico dentro de asignaturas de programación, sigue siendo parcial y requiere análisis contextualizados.

La falta de estrategias instruccionales efectivas limita el desarrollo de competencias cognitivas y heurísticas necesarias para la solución de problemas reales, mediante algoritmos y estructuras lógicas trasladadas a un lenguaje de programación (Sánchez-Rodríguez et al., 2023; Velasco, 2020). Lo que provoca que muchos estudiantes adquieran conocimientos fragmentados y dependan totalmente de la guía del docente para avanzar, en lugar de construir aprendizajes significativos de manera autónoma. Frente a esta problemática, la integración de recursos

instruccionales digitales se plantea como una alternativa viable para optimizar los procesos de enseñanza y aprendizaje.

De acuerdo con estudios recientes, el uso planificado de recursos instruccionales contribuye a reducir las brechas entre teoría y práctica, fomenta la autogestión del aprendizaje y fortalece la dimensión práctica del conocimiento (Romero et al., 2021). Sin embargo, es necesario analizar en qué medida dichos recursos inciden de forma diferenciada en las dimensiones teórica, heurística y axiológica del aprendizaje. Esta investigación parte, por tanto, de la necesidad de evaluar empíricamente el impacto de los recursos instruccionales digitales en el desempeño académico de los estudiantes de Fundamentos de Programación, para determinar su efectividad real y las áreas que aún requieren fortalecimiento.

Referente teórico

El modelo TPACK (Technological Pedagogical Content Knowledge) constituye el marco integrador de esta investigación, al articular los conocimientos tecnológico, pedagógico y disciplinar que el docente requiere, para diseñar experiencias de aprendizaje efectivas mediante el uso de tecnología. Flores (2024) analiza el modelo desde una perspectiva narrativa y contextual de la enseñanza en la era digital, misma que demanda la intersección equilibrada entre el dominio del contenido, la aplicación de estrategias pedagógicas y el uso pertinente de herramientas tecnológicas. Según UNIR (2020), el modelo TPACK permite comprender cómo la tecnología puede favorecer la práctica educativa al incorporarse de manera contextualizada en los procesos de enseñanza-aprendizaje.

La tecnología educativa, como *componente del conocimiento tecnológico*, se define como el conjunto de herramientas, recursos y sistemas digitales orientados a mejorar la enseñanza y el aprendizaje. La UNESCO (2021) la describe como cualquier tecnología, hardware o software diseñado con fines educativos, destacando su carácter diverso y adaptable. Esta incluye desde plataformas virtuales y recursos multimedia hasta entornos inmersivos o aplicaciones interactivas, cuyo objetivo es fomentar la inclusión, la accesibilidad y la innovación pedagógica. En este contexto, los Learning Management Systems (LMS) como Moodle (2022) y los entornos en la nube facilitan la organización, seguimiento y evaluación del aprendizaje en la educación superior, consolidando un ecosistema digital enfocado en el estudiante.

Desde el *componente pedagógico* del modelo, las teorías del aprendizaje proporcionan la base conceptual que guía la creación de estrategias instruccionales. El cognitivismo explica el aprendizaje como un proceso mental activo en el que el individuo adquiere y organiza información (Tapia, 2022). El constructivismo sostiene que el aprendizaje se concibe de un proceso activo en el cual los estudiantes construyen significado de forma interna, a partir de la experiencia y la interacción con su entorno (Giannoukos, 2024), en tanto que el aprendizaje autodirigido enfatiza la autonomía y la autorregulación del estudiante en la planificación y evaluación de su propio proceso formativo (Karataş y Arpacı 2021). Estas perspectivas convergen en la necesidad de propiciar aprendizajes significativos, reflexivos y centrados en el estudiante.

El *componente de contenido* del modelo se relaciona con el pensamiento computacional y algorítmico, habilidades esenciales en la formación en programación. De acuerdo a Rojas-López y García-Peñalvo (2020), el pensamiento computacional se considera como la capacidad de formular problemas y diseñar soluciones de modo que puedan ser ejecutadas por un procesador de información, mientras que Pires et al. (2024) consideran el pensamiento algorítmico como el proceso de analizar, estructurar y resolver problemas mediante secuencias lógicas y precisas. Estas competencias son fundamentales para el desarrollo del razonamiento lógico, la abstracción y la resolución de problemas en contextos tecnológicos y educativos.

La tecnología instruccional representa la aplicación práctica de los principios del modelo TPACK, al incorporar la tecnología educativa, las teorías del aprendizaje y el conocimiento disciplinar en el diseño de experiencias formativas. Seels & Richey (1994) la definen como la teoría y práctica del diseño, desarrollo y uso de recursos que optimizan el aprendizaje. En este marco, los recursos instruccionales digitales constituyen los medios concretos que guían la instrucción, promueven la práctica autónoma y fortalecen la transferencia del conocimiento (Kammerer-Rojas, 2023; Molenda, 2022). Su diseño requiere una planificación pedagógica sistemática que garantice la pertinencia, interactividad y coherencia del material educativo.

Por último, los modelos de diseño instruccional proporcionan el soporte metodológico para la creación de materiales educativos. El modelo ADDIE (Análisis, Diseño, Desarrollo, Implementación y Evaluación) ofrece una estructura secuencial que permite planificar, aplicar y retroalimentar los procesos de enseñanza (Fadila et al., 2020). Por su parte, el modelo ASSURE (Eliana et al., 2024) enfatiza la participación activa del estudiante y el uso eficaz de los medios y materiales educativos. Ambos modelos garantizan la alineación entre objetivos, contenidos y

evaluación, asegurando que los recursos instruccionales digitales cumplan su propósito de mejorar el aprendizaje y el rendimiento académico en la educación superior.

El presente estudio se sustenta teóricamente en la metodología de la Investigación Basada en el Diseño (Design-Based Research, DBR), orientada a la innovación educativa mediante la incorporación de elementos nuevos que transformen situaciones reales. Esta metodología busca mejorar los procesos de aprendizaje al vincular de manera dinámica la teoría y la práctica en contextos auténticos. Su propósito es generar conocimiento útil, aplicable y transferible, derivado de la reflexión sobre la intervención educativa y su impacto (Rolf, 2021)

De acuerdo a lo planteado por Hoadley y Campos (2022), la Design-Based Research (DBR) integra un proceso de diseño educativo que se perfecciona de manera continua con una evaluación sistemática de sus resultados, lo que permite ajustar y mejorar las soluciones durante su implementación. Este enfoque resulta particularmente valioso, cuando se busca comprender y transformar al mismo tiempo la práctica pedagógica, ya que favorece la creación de innovaciones situadas, que respondan a necesidades educativas reales. Bajo este modelo, la presente investigación desarrolló una solución innovadora para la enseñanza de la programación.

Objetivo general

Analizar los recursos instruccionales digitales que favorecen el rendimiento académico relacionado con las dimensiones heurística, teórica y axiológica de los estudiantes de la EE de *Fundamentos de Programación* de la Licenciatura en Sistemas Computacionales Administrativos de la Universidad Veracruzana, durante el periodo agosto 2024–enero 2025.

Hipótesis

Partiendo de la pregunta de

Se plantean las siguientes hipótesis:

H_0 = Los recursos instruccionales digitales no favorecen el rendimiento académico de Fundamentos de Programación.

H_i = Los recursos instruccionales digitales favorecen el rendimiento académico de Fundamentos de Programación.

Método

La investigación se desarrolló bajo un enfoque cuantitativo con diseño cuasiexperimental, orientado a analizar los recursos instruccionales digitales que favorecen el aprendizaje de los estudiantes de la EE de Fundamentos de Programación. De acuerdo con Hernández-Sampieri y Mendoza (2018), este tipo de enfoque permite recolectar datos medibles y analizar vínculos entre variables mediante procedimientos estadísticos.

El estudio se basó en la manipulación parcial de la variable independiente (los recursos instruccionales digitales) para observar sus efectos sobre la variable dependiente (el aprendizaje de los estudiantes), sin controlar todos los factores del entorno. Tal como señalan Hernández - Sampieri y Mendoza (2018) los diseños cuasiexperimentales manipulan deliberadamente al menos una variable independiente para observar su efecto en una o más variables dependientes, mientras que Tamayo y Tamayo (2019) precisan que estos estudios examinan relaciones causa-efecto sin las condiciones de control propias de un experimento puro.

La propuesta innovadora se materializó en la creación de recursos instruccionales digitales elaborados conforme a los principios de la DBR, incorporando elementos visuales, auditivos y prácticos que facilitan la comprensión de conceptos complejos en programación. Estos materiales fueron alojados en una plataforma web diseñada para el proyecto, llamada “Código para todos”. El sitio reúne diversos apoyos didácticos, entre ellos videos animados, guías ilustradas, ejercicios prácticos y actividades interactivas, todos orientados a promover un aprendizaje activo, participativo y con sentido.

El propósito central de esta intervención fue proporcionarles a los estudiantes un espacio de aprendizaje autónomo, accesible y visualmente estimulante, que facilitara el fortalecimiento de los contenidos teóricos y prácticos fuera del aula. Los materiales digitales se incorporaron como un apoyo complementario a las sesiones presenciales, funcionando como un recurso permanente dentro del proceso de evaluación formativa. Las actividades relacionadas con los videos y las guías se asignaron para su realización autónoma, favoreciendo la práctica continua, el desarrollo de habilidades heurísticas y la consolidación del conocimiento a través de la aplicación gradual de los conceptos revisados en clase.

Para elaborar los recursos instruccionales se recurrió a diversas herramientas tecnológicas de diseño y edición digital, que facilitaron tanto su creación como su difusión. La

plataforma Google Sites funcionó como el repositorio principal donde se alojaron los materiales (Google, 2024), y YouTube permitió compartir los videos educativos de manera abierta (YouTube, 2024). Además, se empleó Canva para la elaboración de gráficos y elementos visuales (Canva, 2024), Clipchamp para la edición de contenidos audiovisuales (Microsoft, 2024) y Adobe Enhance para mejorar la calidad del audio y la imagen (Adobe, 2024).

El uso articulado de estas herramientas posibilitó la creación de recursos interactivos y accesibles, integrables en entornos virtuales y presenciales de aprendizaje. Para evaluar su efectividad, se aplicaron pruebas pretest y postest, antes y después del uso de los recursos, con el fin de medir los conocimientos y la mejora en el rendimiento académico. Este tipo de instrumentos es común en los diseños cuasiexperimentales, al permitir comparar los efectos de una intervención educativa (Creswell & Creswell, 2022).

Así mismo, se llevó a cabo un seguimiento de las calificaciones obtenidas en las actividades asociadas a los recursos digitales, con el fin de identificar si existía una relación entre el uso y la mejora del desempeño académico. La exposición de los resultados, sustentada en el análisis de los datos obtenidos en el pretest y el postest, muestran de manera clara el efecto favorable que tuvieron los recursos instruccionales, tanto en el rendimiento académico, como en la percepción de adquisición de conocimientos de los estudiantes.

Estos resultados subrayan el papel fundamental del aprendizaje autónomo en el uso de recursos instruccionales digitales, ya que favorece la autorregulación y la responsabilidad del estudiante en su propio proceso formativo. Como señalan Romero et al (2021), la autonomía en el aprendizaje se potencia cuando los materiales educativos digitales ofrecen retroalimentación inmediata, permiten que cada quien marque su propio ritmo de estudio y brindan espacios flexibles de práctica que acompañan el desarrollo gradual de las competencias.

En este sentido, el diseño y la implementación de recursos instruccionales en Fundamentos de Programación, se orienta a impulsar aprendizajes significativos y sostenibles dentro de la educación superior. De acuerdo con Biggs y Tang (2022), el aprendizaje significativo acontece cuando los estudiantes logran vincular los nuevos conocimientos con sus experiencias previas y aplicarlos en situaciones reales, lo que favorece una comprensión más profunda y duradera.

Asimismo, Sterling (2021) señala que un aprendizaje sostenible implica formar estudiantes capaces de adaptarse, pensar críticamente y asumir un compromiso activo con el

cambio social. Por su parte, la UNESCO (2023) destaca que la sostenibilidad educativa requiere integrar prácticas pedagógicas que fomenten la autonomía, la innovación y la responsabilidad frente a los desafíos globales.

Clasificación y diseño de la investigación

La investigación se clasificó, por el tipo de datos recolectados, como cuantitativa, dado que se obtuvieron datos numéricos mediante la aplicación de cuestionarios estructurados pretest y posttest a los estudiantes de la EE de Fundamentos de Programación. El alcance fue explicativo, al analizar el efecto de la implementación de recursos instruccionales digitales sobre el aprendizaje de los estudiantes.

Según la fuente de recolección de datos, fue experimental, al manipular la variable independiente y observar su impacto en la variable dependiente. De acuerdo con el plan de recolección de datos, correspondió a un diseño cuasiexperimental, pues no se contó con control total de las variables externas. Este tipo de diseño permite evaluar relaciones causa-efecto en contextos reales de aula, combinando la medición empírica con la observación directa (Hernández-Sampieri y Mendoza, 2018; Tamayo & Tamayo, 2019).

Población de estudio

La investigación se enfocó en 129 estudiantes que cursaron la experiencia educativa de Fundamentos de Programación, pertenecientes al programa educativo de Licenciatura en Sistemas Computacionales Administrativos de la Universidad Veracruzana, región Xalapa, durante el periodo agosto 2024–enero 2025. Dicha asignatura es una de las materias con mayor índice de reprobación en el programa y se imparte únicamente en un semestre, lo que justifica su relevancia como objeto de estudio.

El muestreo fue de tipo no probabilístico intencional considerando a los estudiantes que participaron voluntariamente, quienes completaron los instrumentos (pretest y posttest), quedando la muestra integrada por 98 estudiantes.

Variables de investigación

La investigación contempló dos variables principales: recursos instruccionales digitales y rendimiento académico.

Recursos instruccionales digitales: conjunto de materiales, medios y herramientas tecnológicas diseñadas con fines educativos para facilitar la comprensión, práctica y aplicación del conocimiento en entornos virtuales o presenciales, mediante la plataforma Código para todos. Dicha plataforma incluye tutoriales, videos, simuladores y guías interactivas que promueven el aprendizaje autónomo y significativo del estudiante (Cabero-Almenara & Llorente-Cejudo, 2021).

Rendimiento académico: nivel de logro alcanzado por el estudiante en el proceso de aprendizaje, expresado a través las dimensiones teórica, heurística y axiológica, que reflejan el grado de comprensión, ejecución y valoración del aprendizaje en la asignatura Fundamentos de Programación (Rojas-López y García-Peñalvo, 2020).

Operacionalización de variables

La operacionalización de variables permitió vincular los conceptos teóricos con los elementos observables y medibles del estudio. Se definieron las variables principales (recursos instruccionales digitales y rendimiento académico) junto con sus subvariables, indicadores e instrumento de recolección de datos, como se muestra en la tabla 1.

Tabla 1. Operacionalización de variables.

Variable	Subvariables	Indicadores	Instrumento de recolección
Recursos instruccionales digitales (Variable independiente)	Plataforma Código para todos	Accesibilidad, facilidad de uso, pertinencia con los contenidos de programación, frecuencia de uso, percepción de utilidad	Cuestionario postest (escala Likert de 5 puntos)
Rendimiento académico (Variable dependiente)	Dimensión teórica	Comprensión de conceptos, estructuras algorítmicas, fundamentos de Java	Cuestionario pretest y postest
	Dimensión heurística	Habilidades prácticas en resolución de problemas y aplicación de algoritmos	Cuestionario pretest y postest
	Dimensión axiológica	Actitudes, motivación e interés hacia la programación	Cuestionario pretest y postest

Fuente: elaboración propia.

Validación de instrumentos

El cuestionario fue validado mediante juicio de expertos, quienes evaluaron la claridad, congruencia, pertinencia y coherencia interna de los ítems con las dimensiones teórica, heurística y axiológica. Se realizaron ajustes léxicos y semánticos con base en sus observaciones. Posteriormente, se aplicó una prueba piloto a un grupo de estudiantes que no formó parte de la muestra final, con el propósito de verificar la comprensión y consistencia del instrumento.

Las observaciones obtenidas permitieron realizar ajustes menores en la redacción de los ítems y confirmar la validez y confiabilidad del cuestionario, conforme a las recomendaciones metodológicas de Hernández-Sampieri y Mendoza (2018) y Cabero-Almenara et al. (2021).

Recolección y análisis de datos

La recolección de datos se realizó entre agosto 2023 y enero 2024, tras la autorización institucional y el consentimiento de los participantes. Se aplicaron cuestionarios pretest y postest a 98 estudiantes de Fundamentos de Programación, con secciones sobre datos sociodemográficos, actitudes y competencias en programación. Las respuestas fueron codificadas y analizadas mediante los programas Minitab Statistical Software (v.17) y Statistica (v.7), empleados para el procesamiento descriptivo e inferencial de los datos (Minitab LLC, 2017; StatSoft Inc., 2004).

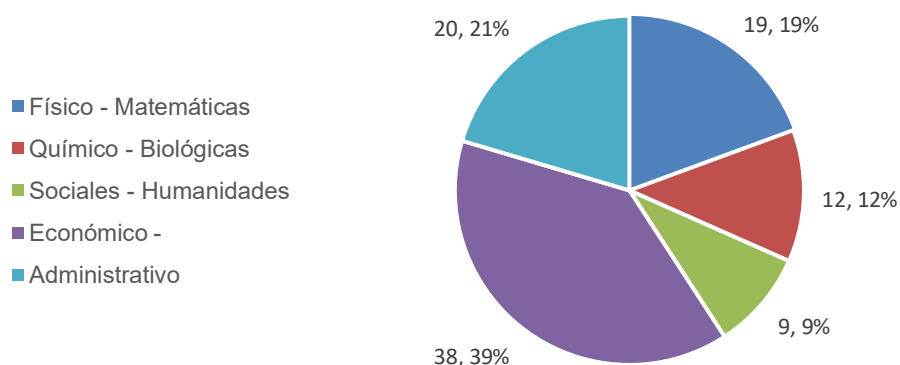
El análisis se realizó en tres momentos: primero a través de la análisis descriptivo con el cálculo de frecuencias, porcentajes y gráficos; un segundo momento para el análisis inferencial mediante la prueba de normalidad para determinar el tipo distribución de los datos, una vez que se identificó que los datos no cumplieron con el supuesto de normalidad se procedió a realizar la prueba de hipótesis no paramétrica de Wilcoxon por pares, la cual evalúa si hay una diferencia significativa entre dos condiciones relacionadas, es decir entre el Pretest y Postet, comparando los rangos de las diferencias entre las muestras. Este análisis permitió contrastar las hipótesis y establecer el impacto de los recursos instruccionales sobre el aprendizaje, conforme a las recomendaciones de George & Mallery (2021); finalmente para complementar dicho análisis se presentan diagramas de caja y alambre a fin de visualizar los cambios entre pretest y postest.

Resultados y Discusiones

Resultados

Caracterización sociodemográfica

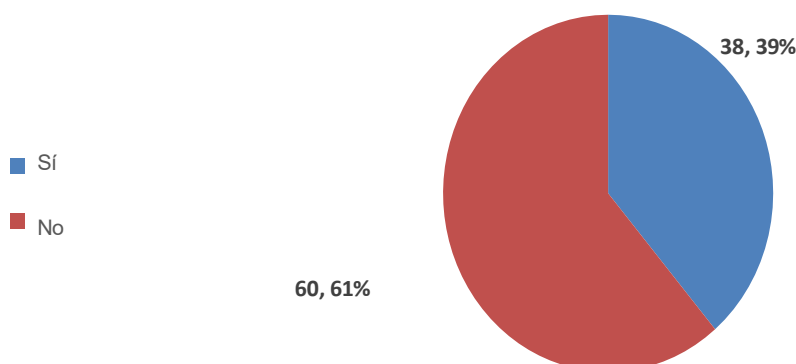
Figura 1. Distribución de estudiantes según área de conocimiento del bachillerato.



Fuente: elaboración propia (2024).

Para este indicador en la figura 1, se observan los resultados iniciales del pretest, los cuales permiten describir el perfil de los 98 estudiantes participantes. La mayoría proviene del área Económico-Administrativa del bachillerato, seguida de áreas técnicas y Físico-Matemáticas.

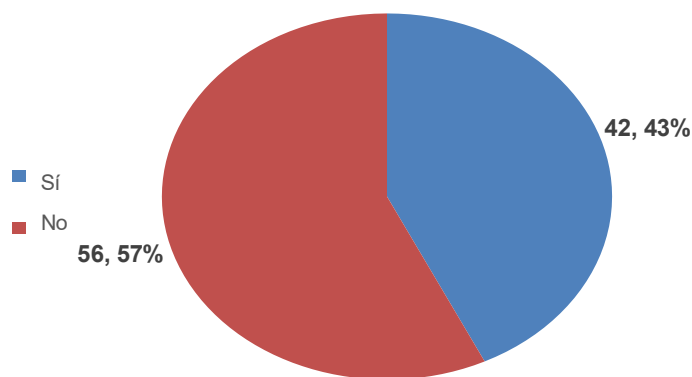
Figura 2. Distribución de estudiantes que cursaron asignaturas relativas a la programación.



Fuente: elaboración propia (2024).

Asimismo, en la figura 2, se identifica que el 39 % había cursado materias relacionadas con programación fuera del programa educativo, mientras que el 57 % reportó no tener conocimientos previos en esta disciplina.

Figura 3. Distribución de estudiantes con conocimientos en programación.



Fuente: elaboración propia (2024).

Finalmente, se muestra en la figura 3 que solo el 8 % cursaba la EE por segunda vez y un 16 % había aplazado su inscripción anterior, lo que refleja las dificultades que históricamente presenta la asignatura Fundamentos de Programación (Vinueza-Morales et al., 2025).

Resultados del pretest

En la medición inicial, el 64.3 % de los estudiantes se declaró muy interesado y el 46.9 % muy motivado para cursar la asignatura. El 55.1 % manifestó entusiasmo alto por iniciar clases y el 51 % un gusto medianamente alto por la programación. No obstante, el 49 % consideró la materia medianamente difícil y el 27.6 % muy difícil, evidenciando la percepción de reto que caracteriza a la asignatura. Aun así, el 85.7 % reconoció la importancia de adquirir habilidades en programación y el 72.4 % valoró positivamente disponer de una guía práctica complementaria, como se muestra en la tabla 2.

Tabla 2. Resultados del pretest en frecuencias y porcentajes.

Pregunta	Escala	Frecuencia	Porcentaje
Interés por cursar la EE	Muy interesado	63	64.3
Motivación inicial	Muy motivado	46	46.9
Dificultad percibida	Medianamente difícil	48	49
Importancia de desarrollar habilidades	Medianamente importante	84	85.7
Valoración de guía práctica	Muy importante	71	72.4

Fuente: elaboración propia (2024).

Resultados de Postest

Tras la implementación de los recursos instruccionales digitales, se observaron variaciones positivas en las percepciones estudiantiles. El 64.3% reportó mantenerse muy interesado en la asignatura y el 44.9 % muy motivado para continuar con cursos de programación. Un 58.2 % reconoció haber adquirido conocimientos y desarrollado habilidades en la materia, mientras que el 59.2 % consideró que la programación representa un reto formativo valioso para su desarrollo profesional, como se muestra en la tabla 3.

Tabla 3. Resultados del postest en frecuencias y porcentajes.

Pregunta	Escala	Frecuencia	Porcentaje
Interés posterior	Muy interesado	63	64.3
Motivación para continuar	Muy motivado	44	44.9
Adquisición de habilidades	Mucho	57	58.2
Utilidad de la guía	Muy útil	50	51
Satisfacción con los recursos	Totalmente de acuerdo	48	49

Fuente: elaboración propia (2024).

El 83.7 % utilizó la guía de la estudiante propuesta, de los cuales el 51 % la calificó como muy útil. Asimismo, el 54.1 % afirmó usar frecuentemente los materiales digitales (videos, guías, actividades) y el 37.8 % lo hizo siempre. El 49 % de los estudiantes coincidió en que los recursos

fueron fáciles de acceder, y un 50 % afirmó que promovieron la práctica y consolidación de habilidades algorítmicas.

Plataforma Código para todos

Los registros de Google Analytics evidencian una participación activa de los estudiantes. El sitio recibió 444 usuarios, de los cuales 345 accedieron desde computadoras y 101 desde teléfonos móviles. La página más visitada fue Operadores (912 vistas), seguida por Introducción a Java (689 vistas) y Programación Orientada a Objetos (POO) (399 vistas), como se muestra en la tabla 4. Es importante aclarar que, predominaron los accesos desde el sistema operativo Windows (73.4 %) y el navegador Chrome (84 %), lo que demuestra la accesibilidad y estabilidad técnica de la plataforma digital (Google Analytics, 2023), como se muestra en la tabla 4.

Tabla 4. Actividad por página en la plataforma Código para todos.

Página	Vistas	Usuarios activos	Tiempo promedio (s)
Operadores	912	322	66.4
Introducción a Java	689	203	62.9
POO	399	168	163.5
Arreglos	175	85	151.1
Bucles	148	79	163.2

Fuente: elaboración propia (2024).

Análisis inferencial

Para comprobar la normalidad de los datos mediante la prueba de Shapiro-Wilk en Minitab 17, se empleó la prueba de rangos con signo de Wilcoxon para contrastar los valores del pretest y posttest (Hernández-Sampieri y Mendoza, 2018). Para determinar si hay una diferencia significativa en la prueba por pares de Wilcoxon, se fijó en el valor de P que arrojará Statistica, con el valor de la prueba de normalidad, de $P > 0.05$, la cual se muestra en la tabla 5.

Tabla 5. Resultados de la prueba de Wilcoxon por pares.

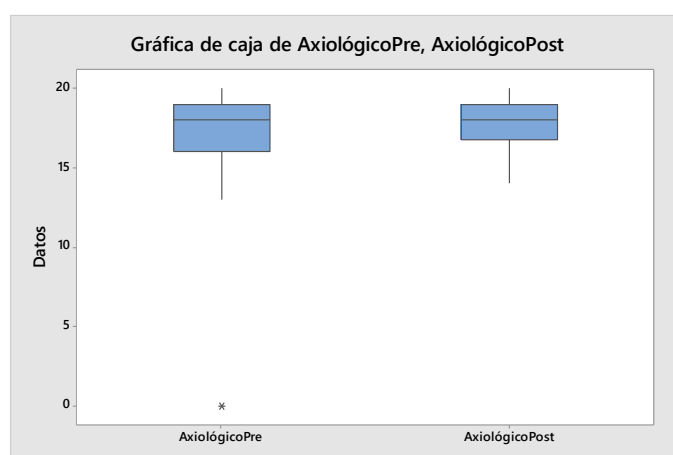
Dimensión	Z	p	Interpretación
Axiológica	-0.668	0.504	No significativa
Teórica	-5.469	0	Diferencia significativa
Heurística	-4.54	0	Diferencia significativa positiva

Fuente: elaboración propia (2024).

En consecuencia, se rechaza la hipótesis nula (H_0) y se acepta la hipótesis alterna (H_a), concluyendo que *los recursos instruccionales digitales favorecen significativamente el aprendizaje de la programación*, especialmente en la dimensión heurística, al fortalecer las habilidades prácticas y la aplicación de conceptos (Imbaquingo & Romero, 2025).

Los análisis estadísticos se realizaron mediante diagramas de cajas y alambres, así como a través de la prueba de hipótesis de Wilcoxon por pares. Este análisis se estructuró en torno a tres ejes fundamentales: el *axiológico*, orientado a los valores y actitudes hacia la programación; el *teórico*, centrado en la comprensión de conceptos y fundamentos; y el *heurístico*, enfocado en el desarrollo de habilidades para aplicar e implementar algoritmos. Estos ejes permiten evaluar de manera integral el conocimiento adquirido (Beltrán, 2005), cuyos resultados se presentan a continuación.

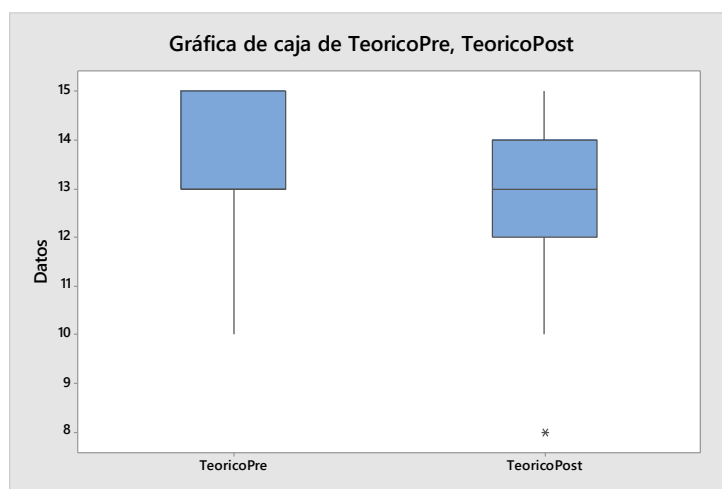
Figura 4. Comparación Axiológica Pre y Post Intervención.



Fuente: elaboración propia (2024).

La figura 4 de cajas del AxiológicoPre y AxiológicoPost, muestra una ligera diferencia entre ambos momentos. En el AxiológicoPre la dispersión es mayor y aparece un valor atípico muy bajo, mientras que en el AxiológicoPost los datos se presentan más concentrados y con un rango intercuartil menor. Aunque el cambio no es tan marcado, los valores se mantienen altos en ambas mediciones, lo que evidencia que la actitud de los estudiantes hacia la programación se conserva estable y favorable.

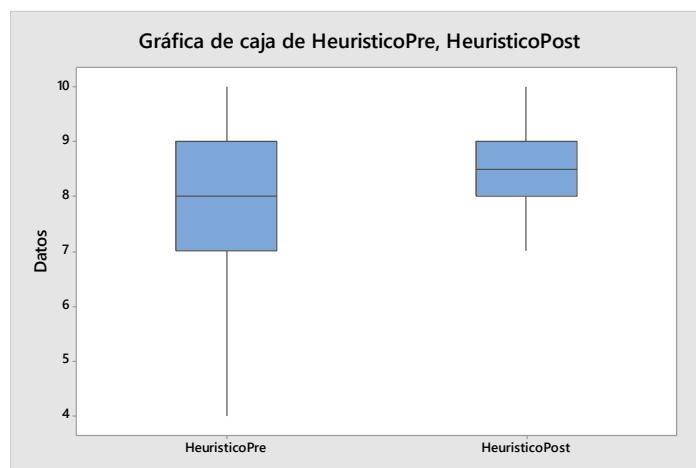
Figura 5. Comparación Teórica Pre y Post Intervención.



Fuente: elaboración propia (2024).

La figura 5 de cajas del TeóricoPre y TeóricoPost, evidencia una diferencia más marcada entre ambas mediciones. En el TeóricoPre los valores se concentran en rangos superiores, mientras que en el TeóricoPost la distribución desciende y aparece un valor atípico bajo. Esto refleja una ligera disminución en el dominio teórico sobre diseño e implementación de algoritmos, experiencia educativa previa.

Figura 6. Comparación Heurística Pre y Post Intervención.



Fuente: elaboración propia (2024).

La figura 6 de cajas del HeurísticoPre y HeurísticoPost, muestra una diferencia clara entre ambos momentos. En el HeurísticoPre la dispersión es más amplia, con valores que se extienden hacia rangos bajos, mientras que en el HeurísticoPost los datos se concentran en niveles superiores y con menor variabilidad. Este comportamiento indica una percepción más homogénea respecto a la dificultad de la programación. Aunque el área sigue considerándose desafiante, los estudiantes reconocen una mejora en sus habilidades para aplicar e implementar algoritmos.

Los resultados muestran que las dimensiones actitudes hacia la programación y percepción de dificultad no presentan discrepancias significativas ($p = 0.504$ y $p = 0.570$, respectivamente). Sin embargo, se identificaron diferencias estadísticamente significativas ($p < 0.05$) en las dimensiones adquisición de conocimientos y habilidades, uso y percepción de los recursos instruccionales.

El análisis de cajas indica una concentración mayor de los valores en la dimensión heurística tras la intervención, reflejando un progreso en la capacidad práctica y algorítmica de los estudiantes. En cambio, la dimensión teórica mostró una ligera reducción, lo que sugiere la necesidad de reforzar la comprensión conceptual en futuras implementaciones.

Los datos de la tabla 6 confirman el progreso en la dimensión heurística, donde el 80% de los estudiantes reportó sentirse más confiado en programación y el 71% utilizó los recursos regularmente. El análisis estadístico (Wilcoxon, $p=0.000003$) valida que las mejoras en

habilidades prácticas son significativas, con incrementos consistentes del 23-26% en competencias algorítmicas. En contraste, la dimensión teórica mostró descenso significativo ($p=0.039858$), con reducción del primer cuartil de 13 a 12 puntos, indicando que algunos estudiantes privilegiaron la práctica sobre la reflexión conceptual. La dimensión axiológica mantuvo estabilidad ($p=0.051178$, mediana=18), evidenciando que los recursos preservaron la motivación inicial.

Tabla 6. Comparación de competencias de programación pre y post intervención.

Competencias	Pre (%)	Post (%)	Incremento	Wilcoxon p-value
<i>Dimensión Heurística</i>				$p=0.000003^*$
Habilidades prácticas en resolución de problemas y aplicación de algoritmos	42%	65%	+23%	
	35%	58%	+23%	
	31%	54%	+23%	
	45%	71%	+26%	
<i>Dimensión Teórica</i>				$p=0.039858^*$
Comprensión conceptual (mediana)	13	13	Q1: 13→12	
<i>Dimensión Axiológica</i>				$p=0.051178$
Motivación (mediana)	18	18	Estable	
Confianza en habilidades	46.9%	80%	+33.1%	

Fuente: elaboración propia (2025). Análisis realizado con una muestra de $n = 98$ estudiantes. Los porcentajes representan a quienes alcanzaron un dominio $\geq 70\%$ en los ejercicios. El valor p corresponde a la prueba de Wilcoxon; $p < 0.05$ indica diferencias estadísticamente significativas.

Discusión

Los resultados confirman que los recursos instruccionales digitales (variable independiente) impactaron diferencialmente las tres dimensiones del rendimiento académico (variable dependiente). La dimensión heurística registró mejora significativa ($p=0.000003$) gracias a estrategias pedagógicas específicas implementadas en "Código para todos": videos explicativos paso a paso (utilizados por el 75.5%), diagramas de flujo y elementos visuales (73.5%), actividades prácticas graduales (69.4%), y retroalimentación contextualizada disponible 24/7. Esta combinación de secuenciación progresiva de contenidos, representaciones visuales de algoritmos y ejercicios de complejidad creciente fortaleció el pensamiento computacional y la capacidad de resolución de problemas, permitiendo que el 65% dominara condicionales, el 58% ciclos, y el 54% estructuras de datos, comparado con 42%, 35% y 31% respectivamente en el pretest. Sin embargo, la dimensión teórica descendió ($p=0.039858$), sugiriendo que los

estudiantes aplicaron conceptos mecánicamente sin profundizar en su fundamentación, coincidiendo con Rojas López y García-Peñalvo (2020), sobre riesgos de aprendizajes superficiales cuando la tecnología no se acompaña de reflexión conceptual. La dimensión axiológica mantuvo estabilidad ($p=0.051178$), evitando el declive motivacional frecuente en programación.

La mejora observada en la dimensión heurística respalda lo planteado por Labanda et al. (2024), quienes sostienen que la combinación de recursos multimedia y ejercicios prácticos fortalece el pensamiento lógico y algorítmico. Los resultados postintervención muestran que los estudiantes logran mayor dominio en la aplicación de conceptos y resolución de problemas, lo que evidencia la efectividad del enfoque instruccional. De igual forma, la retroalimentación inmediata y la disponibilidad permanente de los materiales están en consonancia con los principios del aprendizaje autodirigido descritos por Romero et al (2021).

Desde la perspectiva del modelo TPACK, los hallazgos corroboran la importancia de integrar de manera armónica los conocimientos tecnológicos, pedagógicos y disciplinares. Según Mishra y Koehler (2006), la intersección de estos tres componentes posibilita experiencias de enseñanza contextualizadas y efectivas. En el caso analizado, la implementación del sitio “*Código para todos*”, ejemplifica una aplicación práctica de este modelo, al combinar contenidos disciplinares con estrategias didácticas y herramientas digitales que facilitan la comprensión y práctica de la programación en entornos mixtos.

Con el propósito de fortalecer el desarrollo del pensamiento computacional y la resolución de problemas, las estrategias didácticas utilizadas como recurso de apoyo y complemento a través del aprendizaje basado en problemas, mediante la resolución de situaciones reales, que requirieron procesos de análisis, abstracción y diseño de soluciones algorítmicas. De manera complementaria, se fomentó el aprendizaje colaborativo a través del trabajo en equipo y lluvia de ideas, para llevar a discusión posibles soluciones, fortaleciendo habilidades de comunicación, pensamiento crítico y toma de decisiones en el proceso de programación.

Por otro lado, los resultados también evidencian la necesidad de fortalecer la dimensión teórica, en la que se observaron leves descensos tras la intervención. Este fenómeno coincide con lo señalado por Rojas-López y García-Peñalvo (2020), quienes advierten que el uso de recursos tecnológicos debe acompañarse de estrategias que promuevan la reflexión conceptual para evitar aprendizajes superficiales. En concordancia, Biggs y Tang (2022) subrayan que el

aprendizaje significativo se alcanza cuando los estudiantes integran la teoría con la práctica y son capaces de transferir el conocimiento a nuevos contextos.

Finalmente, los hallazgos coinciden con la visión de Sterling (2021) y la UNESCO (2023), al evidenciar que la sostenibilidad del aprendizaje depende de la capacidad del estudiante para mantener la motivación, la autorregulación y la adaptación ante entornos tecnológicos cambiantes. La permanencia de actitudes positivas hacia la programación sugiere que los recursos digitales no solo promueven resultados académicos inmediatos, sino también un aprendizaje sostenible orientado a la formación continua y al desarrollo de competencias para el futuro.

Conclusiones

La presente investigación analiza el impacto de los recursos instruccionales digitales en el aprendizaje de los estudiantes de la EE de Fundamentos de Programación de la Licenciatura en Sistemas Computacionales Administrativos de la Universidad Veracruzana. El estudio parte del problema del bajo rendimiento y la alta tasa de reprobación en asignaturas de programación, con el objetivo de evaluar si la incorporación de materiales digitales, diseñados bajo el enfoque de la Design-Based Research (DBR), favorece el desempeño académico y la motivación estudiantil. Los resultados obtenidos evidencian un progreso importante en la dimensión heurística y una percepción positiva hacia el uso de los recursos digitales, tal como se muestra en la figura 7.

Figura 7. Síntesis de investigación.



Fuente: elaboración propia (2025).

Los hallazgos confirman que la integración de recursos instruccionales digitales fortalece la comprensión práctica de los contenidos, incrementa la motivación y promueve la autonomía en el aprendizaje. Los estudiantes muestran mayor disposición para continuar con asignaturas afines a la programación, lo que coincide con lo planteado por Cabero-Almenara et al. (2021) y Labanda et al. (2024), quienes destacan que los materiales digitales interactivos potencian el aprendizaje activo y la autorregulación. Asimismo, se observa que los recursos implementados bajo el modelo TPACK posibilitan una relación coherente entre la pedagogía, la tecnología y el contenido disciplinar, mejorando la experiencia de enseñanza-aprendizaje.

Entre las principales limitaciones se identifica la naturaleza cuasiexperimental del estudio y el tiempo de exposición a los recursos, lo cual restringe la posibilidad de analizar efectos a largo plazo. Además, la evaluación se centró en una sola EE, por lo que los resultados no pueden generalizarse a otros contextos sin estudios complementarios. Futuras investigaciones podrían incorporar métodos mixtos que incluyan la observación del desempeño en entornos reales de programación o el uso de analítica de aprendizaje para monitorear el progreso individual y grupal.

En conclusión, los recursos instruccionales digitales constituyen una estrategia pedagógica eficaz para mejorar el aprendizaje en programación, al propiciar experiencias significativas, motivadoras y sostenibles. Su implementación permite reducir la brecha entre la teoría y la práctica, optimizar el tiempo dedicado al estudio autónomo y fomentar competencias esenciales en el ámbito tecnológico. Se recomienda su adopción institucional como parte de los programas formativos, priorizando su actualización constante y el fortalecimiento de la dimensión teórica mediante estrategias reflexivas que consoliden la comprensión conceptual.

Estos hallazgos, respaldados estadísticamente (dimensión heurística: $p=0.000003$; teórica: $p=0.039858$; axiológica: $p=0.051178$), confirman que los recursos instruccionales digitales fortalecen competencias prácticas, pero requieren complementarse con estrategias reflexivas para consolidar la comprensión conceptual. Como trabajo futuro, se propone:

1. Integrar actividades metacognitivas (cuestionarios reflexivos, mapas conceptuales interactivos) para fortalecer la dimensión teórica;
2. Incorporar gamificación (sistemas de puntos, retos progresivos) para incrementar participación;
3. Evolucionar "Código para todos" hacia un LMS con IA que ofrezca retroalimentación personalizada y asistente virtual;

4. Implementar analítica del aprendizaje para monitorear progreso individual y detectar estudiantes en riesgo; y
5. Realizar estudios longitudinales que evalúen impacto a mediano plazo en asignaturas subsecuentes.

La implementación progresiva de estas mejoras consolidará la plataforma como herramienta integral y escalable que promueva competencias digitales esenciales para la formación continua en el ámbito tecnológico.

Referencias

- Adobe. (2024). *Adobe Enhance: Enhance speech and audio quality online*. Adobe Inc. <https://www.adobe.com/express/feature/audio/enhance>
- Biggs, J., & Tang, C. (2022). *Teaching for quality learning at university* (5th ed.). Open University Press.
- Beltrán, J., (2005). El Modelo Educativo Integral y Flexible de la Universidad Veracruzana. *CPU-e, Revista de Investigación Educativa*, (1), 1-10. <https://www.redalyc.org/articulo.oa?id=283121715005>
- Cabero-Almenara, J., Guillén-Gámez, F. D., Ruiz-Palmero, J., & Palacios-Rodríguez, A. (2021). Digital competence of higher education faculty according to DigCompEdu: Statistical research methods with ANOVA among areas of knowledge across different age ranges. *Education and Information Technologies*, 26, 4691–4708. <https://doi.org/10.1007/s10639-021-10476-5>
- Canva. (2024). *Canva for education*. Canva Pty Ltd. <https://www.canva.com/education>
- Creswell, J. W., & Creswell, J. D. (2022). *Research design: Qualitative, quantitative, and mixed methods approaches* (6th ed.). SAGE.
- Eliana, N., Wati, U. A., & Rahmadona, S. (2024). Leveraging the ASSURE model for optimized information technology-based learning media. *Al-Ishlah: Jurnal Pendidikan*, 16(3), 1614-3626. <https://doi.org/10.35445/alishlah.v16i3.5639>
- Fadila, S. N., Ahmad, D., & Nurpahmi, S. (2020). Desarrollo de materiales didácticos basados en el modelo de aprendizaje por investigación. *ELTIES*, 2(1), 54-68. <https://doi.org/10.24252/elties.v2i1.11275>
- Flores, F. (2024). Modelo TPACK: su aplicación en el análisis de la integración de las TIC a la enseñanza universitaria. Narrativa, contexto y praxis docente. *Praxis Educativa*, 28(3), 1-20. 134–153. <https://dx.doi.org/10.19137/praxiseducativa-2024-280308>
- George, D., & Mallery, P. (2021). *IBM SPSS Statistics 27 step by step: A simple guide and reference*. Routledge.
- Giannoukos, G. (2024). Main learning theories in education. *European Journal of Contemporary Education and E-Learning*, 2(5), 93–100. [https://doi.org/10.59324/ejceel.2024.2\(5\).06](https://doi.org/10.59324/ejceel.2024.2(5).06)
- Google. (2024). *Google Sites: Create a website*. Google LLC. <https://sites.google.com>
- Google Analytics. (2023). *Google Analytics* [Plataforma de análisis de datos]. Google LLC. <https://analytics.google.com/>
- Hernández-Sampieri, R., & Mendoza Torres, C. P. (2018). *Metodología de la investigación: Las rutas cuantitativa, cualitativa y mixta* (7ª ed.). McGraw-Hill.
- Hoadley, C., & Campos, F. C. (2022). Design-based research: What it is and why it matters to studying online learning. *Theory Into Practice*, 57(3), 207–220. <https://doi.org/10.1080/00461520.2022.2079128>

- Imbaquingo, S. M., & Romero, J. R. (2025). Recursos educativos digitales en la enseñanza de las ciencias experimentales. *Revista Científica y Académica*, 5(3), 4782-4796. <https://doi.org/10.61384/r.c.a..v5i3.1502>
- Karataş, K., & Arpacı, I. (2021). The role of self-directed learning, metacognition, and 21st century skills predicting the readiness for online learning. *Contemporary Educational Technology*, 13(3), 1-13. <https://doi.org/10.30935/cedtech/10786>
- Kammerer-Rojas, M. (2023). Estrategias instruccionales para desarrollar del pensamiento lógico-matemático. *Revista Docentes 2.0*, 16(1), 77–82. <https://doi.org/10.37843/rtd.v16i1.355>
- Labanda, M., Coloma, M., & Michay, G. (2024). *Herramientas digitales para la enseñanza de la programación en el ámbito educativo*. Editorial Universitaria, Universidad Nacional de Loja. https://unl.edu.ec/sites/default/files/archivo/2024-12/Herramientas%20digitales%20para%20la%20ense%C3%B1anza%20de%20la%20programaci%C3%B3n%20en%20el%20%C3%A1mbito%20educativo_1.pdf
- Microsoft. (2024). *Clipchamp video editor*. Microsoft Corp. <https://www.clipchamp.com>
- Minitab LLC. (2017). Minitab Statistical Software (versión 17) [software]. Minitab Inc. <https://www.minitab.com>
- Molenda, M. H. (2022). Historia y desarrollo del diseño y la tecnología instruccional. *En Manual de educación abierta, a distancia y digital*. Springer. https://doi.org/10.1007/978-981-19-0351-9_4-1
- Moodle. (2022). *What is a learning management system (LMS)?* Moodle Pty Ltd. <https://moodle.com>
- Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. (2021). *Technology in education: a tool on whose terms?* UNESCO Publishing. <https://unesdoc.unesco.org/ark:/48223/pf0000386165>
- Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura. (2021). *Reimagining our futures together: A new social contract for education*. UNESCO Publishing. <https://unesdoc.unesco.org/ark:/48223/pf0000379707.locale=en>
- Pires, J. P. J., Brito, F., Gomes, A., Borges, A. R., & Bernardino, J. (2024). *Predicting student performance in introductory programming courses*. *Computers*, 13(9), 1-19. <https://doi.org/10.3390/computers13090219>
- Rojas-López, A., & García-Peñalvo, F. J. (2020). Evaluación de habilidades del pensamiento computacional para predecir el aprendizaje y retención de estudiantes en la asignatura de programación de computadoras en educación superior. *Revista de Educación a Distancia*, 20(63), 1-39. <https://doi.org/10.6018/red.409991>
- Rolf, E. (2021). *Teachers as designers: Analyses of pedagogical patterns and their use* [Doctoral thesis, Department of Computer and Systems Sciences, Stockholm University]. <https://urn.kb.se/resolve?urn=urn:nbn:se:su:diva-197748>

- Romero-Tena, R., Llorente Cejudo, C., & Palacios Rodríguez, A. (2021). Competencias digitales docentes desarrolladas por el alumnado del Grado en Educación Infantil: Presencialidad vs. virtualidad. *EduTec, Revista Electrónica de Tecnología Educativa*, (76), 109–125. DOI <https://doi.org/10.21556/edutec.2021.76.2071>
- Seels, B., & Richey, R. (1994). *Instructional technology: The definition and domains of the field*. Association for Educational Communications and Technology.
- StatSoft Inc. (2004). Statistica (versión 7) [software]. StatSoft Inc. <https://www.statsoft.com>
- Sterling, S. (2021). *Sustainable education: Revisioning learning and change*. Green Books.
- Tapia, H. (2022). Aprendizaje cognoscitivo impulsor de la autorregulación en la construcción del conocimiento. *Revista De Ciencias Sociales*, 28(5), 172-183. <https://doi.org/10.31876/rsc.v28i.38154>
- Tamayo y Tamayo, M. (2019). *El proceso de la investigación científica* (5.ª ed.). Limusa.
- La Universidad en Internet. (2020, 17 diciembre). *Modelo TPACK: Integración de la tecnología en la enseñanza*. Universidad Internacional de La Rioja. <https://www.unir.net/revista/educacion/tpack-que-es/>
- Velasco, M. L. (2020). Resolución de problemas algorítmicos y objetos de aprendizaje: una revisión de la literatura. *Revista Iberoamericana para la Investigación y el Desarrollo Educativo*, 10(20), 1-26. <https://doi.org/10.23913/ride.v10i20.630>
- Vinueza-Morales, M., Rodas-Silva, J., Vidal-Silva, C., Córdova-Morán, J., & Cevallos-Ayón, E. (2025). Teaching programming in higher education: A bibliometric analysis of trends, technologies, and pedagogical approaches. *Frontiers in Education*, 10, 1-11. <https://doi.org/10.3389/feduc.2025.1525917>
- YouTube. (2024). *YouTube for education*. Google LLC. <https://www.youtube.com/@codigoparatodosdag>