

Gamificación con Python e Inteligencia Artificial: Prácticas de Programación en Juegos Integradas con Herramientas Digitales Colaborativas

Gamification with Python and Artificial Intelligence: Game Programming Practices Integrated with Collaborative Digital Tools

Arturo Guerrero Monroy^a, Zen O. Robles Montero^b, Mercedes G. García Reyes^c

Abstract:

This article aims to enhance university students' digital skills through organized practice sessions in games powered by intelligent algorithms, thus fostering creativity, motivation, and collaborative work. Constructivism and project-based learning (PBL) methodologies encourage active participation and the practical application of knowledge; therefore, they constitute the theoretical basis for this approach. The methodological proposal is structured in four stages: practice planning, Python programming with artificial intelligence libraries, integration with digital platforms, and participatory assessment.

Keywords:

Gamification, Python, Artificial Intelligence, Digital Tools, Higher Education, Industry 4.0

Resumen:

El presente artículo tiene como objetivo potenciar las habilidades digitales de los alumnos universitarios mediante prácticas organizadas en juegos con algoritmos inteligentes, impulsando así la creatividad, la motivación y el trabajo colaborativo. Las metodologías del constructivismo y del aprendizaje basado en proyectos (ABP) fomentan la participación activa y la utilización práctica de los saberes; por ello, constituyen la base teórica de esta propuesta. La propuesta metodológica se estructura en cuatro etapas: la planificación de prácticas, la programación en Python con bibliotecas de inteligencia artificial, la integración con plataformas digitales y la valoración participativa.

Palabras Clave:

Gamificación, Python, Inteligencia Artificial, Herramientas digitales, Educación Superior, Industria 4.0

Introducción

En la educación superior, el desarrollo de habilidades digitales es un eje clave para formar profesionales capacitados para la Industria 4.0, en la cual la ciencia de datos, la inteligencia artificial (IA) y la automatización están configurando los nuevos ambientes laborales [1, 2].

En estas circunstancias, es imprescindible desarrollar tácticas pedagógicas que combinen la práctica con la teoría

y que promuevan la creatividad y la motivación del alumnado. La gamificación, que se refiere a la integración de componentes de juego en los procesos educativos, ha sido establecida como una técnica innovadora para fomentar el compromiso y la participación activa de los alumnos [3, 4].

En diversos estudios, el juego en línea promueve la conservación del conocimiento y el desarrollo de habilidades sociales y cognitivas [5].

^{a,b,c} Tecnológico de Estudios Superiores de Ecatepec | División de Informática | Ecatepec-Estado de México | México,

<https://orcid.org/0009-0003-7078-1527>, Email: aguerreromonroy@tese.edu.mx; <https://orcid.org/0000-0001-9643-8426>, Email: roblesm@tese.edu.mx; <https://orcid.org/0009-0001-8815-3188>, Email: mguadalupe@tese.edu.mx

Además, al implementar específicamente a Python como lenguaje de programación se brinda un ambiente accesible para la incorporación de prácticas de inteligencia artificial en juegos; en este contexto, los alumnos tienen la posibilidad de experimentar con simulaciones interactivas, agentes inteligentes y algoritmos de aprendizaje automático [6, 7].

La integración de plataformas digitales colaborativas como, Quizziz, Microsoft Teams, Padlet y Mentimeter se sugiere para maximizar el impacto pedagógico de estos métodos.

La comunicación, la retroalimentación y el trabajo en equipo son componentes esenciales para el progreso de habilidades transversales y para establecer comunidades de aprendizaje digital [8, 9], y estas plataformas facilitan su mejora.

Por lo tanto, la propuesta actual tiene como objetivo combinar la gamificación con Python e IA (Inteligencia Artificial) y el empleo de herramientas colaborativas, con el fin de implementarla como una estrategia pedagógica integral que dinamice la instrucción en programación promueva la innovación y fortalezca las habilidades digitales acordes a los desafíos educativos que se presentan en el siglo XXI.

2. Fundamentación Pedagógica y principales retos educativos.

Para optimizar el impacto pedagógico de estos métodos, se recomienda incorporar plataformas digitales colaborativas como Padlet, Microsoft Teams, Mentimeter y Quizziz. Para el avance de las competencias transversales y para crear comunidades de aprendizaje digital [8, 9], son elementos fundamentales la comunicación, la retroalimentación y el trabajo en equipo, y estas plataformas favorecen su mejora.

Desde una perspectiva sociocultural y constructivista, fomenta la reflexión colectiva y la co-creación del conocimiento porque posibilita que el alumno genere significados a través de la interacción. Sin embargo, según Laurillard [10] y Cabero & Llorente [11], los desafíos educativos más importantes son la capacitación de los docentes en habilidades digitales, la evaluación genuina del aprendizaje colaborativo y la sostenibilidad tecnológica.

En términos generales, estas herramientas, combinadas con métodos activos como el aprendizaje basado en proyectos y la gamificación, ayudan a generar ambientes de aprendizaje que son inclusivos, estimulantes y enfocados en los estudiantes.

2.1. La Gamificación en la educación superior

La gamificación se ha desarrollado en los últimos diez años como un recurso educativo que incorpora componentes y dinámicas lúdicas en entornos de enseñanza-aprendizaje [1]. Su potencial no se restringe al entretenimiento, sino que tiene como objetivo aumentar la autorregulación, la persistencia y la motivación intrínseca del alumno [13, 14].

La gamificación, según estudios recientes en el contexto universitario, promueve la atención sostenida, la

colaboración en entornos digitales y el aprendizaje profundo [1].

Asimismo, investigaciones actuales indican que la gamificación favorece el fortalecimiento de habilidades socioemocionales como el manejo de la frustración y la resiliencia, elementos esenciales para una educación completa en la era digital [2]. Este método facilita la conversión de tareas abstractas en desafíos interactivos, como solucionar algoritmos en un ambiente recreativo o crear agentes inteligentes en contextos simulados, lo que mejora la percepción de la utilidad educativa [3].

2.2. Python e Inteligencia Artificial en la enseñanza de la programación

Python se ha establecido como el lenguaje de referencia para la instrucción en inteligencia artificial y programación debido a su sintaxis clara y su extenso ecosistema de bibliotecas [4]. Su fortaleza en términos educativos se encuentra en la capacidad de combinar visualización, teoría y experimentación a través de herramientas asequibles como Scikit-learn, NumPy y TensorFlow; estas se utilizarán en la Unidad 5 del curso.

NumPy posibilita que los alumnos entiendan la lógica matemática detrás de los algoritmos de inteligencia artificial [5], ya que simplifica el manejo de estructuras de datos numéricos y operaciones vectorizadas. Scikit-learn presenta un conjunto de algoritmos de aprendizaje automático que posibilita la práctica de ideas de predicción y clasificación por parte del alumnado, entre los que sobresalen:

- Árbol de decisión: estructura la información en jerarquías fundamentadas en condiciones lógicas, lo que facilita entender cómo se toman las decisiones en contextos inciertos [6].
- K-Nearest Neighbors (KNN): clasifica a un nuevo dato en función de su semejanza con los "k" vecinos más próximos, lo que fortalece la adquisición de métricas de agrupamiento y distancia entre datos, esenciales para el aprendizaje supervisado [7].

Adicionalmente, Python posibilita la implementación de algoritmos de IA simbólica clásica, por ejemplo, Minimax. Esta herramienta investiga todas las decisiones posibles para hallar el mejor movimiento en situaciones de juego, lo que vincula la programación con la toma de decisiones lógica y la simulación de agentes inteligentes [8].

TensorFlow facilita la creación de modelos de aprendizaje profundo y redes neuronales artificiales, proporcionando visualizaciones del entrenamiento, ajuste de parámetros y retropropagación, lo cual hace más fácil entender los procesos de aprendizaje automático [9].

La integración de estas bibliotecas se llevará a cabo en la Unidad 5: Servicios Web con XML. A través de prácticas de laboratorio, los alumnos diseñarán, evaluarán y probarán modelos de inteligencia artificial (IA) utilizando datos de la base XML. Este tipo de actividades fomenta el aprendizaje

activo e impulsa el desarrollo del pensamiento computacional, el razonamiento lógico y la comprensión de los principios del aprendizaje automático. Estas habilidades son fundamentales para la Industria 4.0 [10]

2.3. Constructivismo y Aprendizaje Basado en Proyectos (ABP)

La teoría constructivista, planteada por Piaget, [19] y Vygotsky, [27] defiende que el saber se genera de manera activa a través de la colaboración social y la interacción con el medio. Desde este punto de vista, la programación de juegos con IA (Inteligencia Artificial) y Python es un ambiente constructivista, ya que los alumnos aprenden a través de la práctica, la experimentación y el análisis de sus errores.

El Aprendizaje Basado en Proyectos (ABP), que Bell [3] y Thomas [25], han retomado, complementa esta perspectiva al poner a los alumnos ante problemas reales que necesitan soluciones cooperativas y creativas. Los alumnos, al crear un videojuego con elementos de inteligencia artificial, integran teoría y práctica, administran los tiempos, toman decisiones en conjunto y presentan un producto final, lo que produce un aprendizaje relevante.

Diversas investigaciones recientes resaltan que el Aprendizaje Basado en Proyectos (ABP), enfocado en la programación, potencia las capacidades para resolver problemas y para pensar de forma computacional, habilidades fundamentales en la Industria 4.0 [11].

2.4. Herramientas digitales colaborativas y aprendizaje activo

Incorporar plataformas digitales al modelo pedagógico refuerza el aprendizaje de los alumnos, ya que, permite la extensión del aula a espacios virtuales en los que se promueve la co-creación y el feedback continuo. La participación activa, el trabajo en equipo para aprender y la formación de comunidades académicas se han visto potenciados por herramientas como Microsoft Teams, Quizziz, Mentimeter y Padlet [12].

Las investigaciones más recientes demuestran que, al combinar plataformas colaborativas con métodos activos se fortalece no solo la dimensión técnica, sino también las habilidades blandas, lo cual fomenta la creatividad, el manejo de proyectos y la comunicación eficaz [13].

Así, la integración de la tecnología no solo simplifica el aprendizaje técnico de Inteligencia Artificial (IA) y programación, sino que además favorece el desarrollo de habilidades transversales, las cuales son esenciales para conseguir empleo en esta época.

2.5. Aprendizaje automático en el juego del gato

El objetivo de esta propuesta es crear un entorno educativo en el que la inteligencia artificial (IA) pueda aprender de sus propias vivencias dentro de un contexto de juego. Para lograr esto, el juego del gato (Tic-Tac-Toe) actúa como un laboratorio de programación donde los estudiantes aplican

técnicas de aprendizaje automático utilizando Python, el algoritmo Minimax, y una base de datos en formato XML que guarda las jugadas realizadas en partidas previas.

El algoritmo Minimax permite simular el razonamiento estratégico que un jugador humano ejecuta al prever los movimientos de su oponente. Su principio básico es minimizar las posibles pérdidas máximas, evaluando todos los movimientos disponibles en el tablero y asignando valores de victoria, empate o derrota a cada uno. Sin embargo, el modelo se potencia al incorporar una base de conocimiento dinámica en formato XML, que documenta cada jugada, el resultado, y las configuraciones del tablero.

De esta forma, el agente inteligente consulta dicha base para recordar jugadas exitosas y ajustar sus estrategias en el futuro, aplicando los principios del aprendizaje por refuerzo simbólico. En lugar de depender únicamente de la búsqueda exhaustiva del Minimax, el sistema integra un tipo de memoria que le permite aprender de la experiencia acumulada, emulando un comportamiento adaptativo.

La estructura XML cumple una doble función:

- Base de datos accesible para almacenar las secuencias de juego de manera estructurada.
- Fuente de entrenamiento para algoritmos de aprendizaje supervisado, como K-Nearest Neighbors (KNN) o Decision Tree, que pueden analizar los patrones en las jugadas exitosas.

Cada vez que se juega una partida, los resultados se añaden al archivo XML. Posteriormente, los modelos en Python (Scikit-learn, NumPy y TensorFlow) procesan los datos, identifican combinaciones de jugadas con mayor probabilidad de éxito y modifican el comportamiento del agente en nuevas sesiones de juego. Así, la IA evoluciona de un modelo determinista (solo Minimax) a uno adaptativo, que ajusta sus decisiones con base en la retroalimentación del entorno.

Este enfoque no solo muestra la aplicabilidad de la inteligencia artificial en un entorno educativo, sino que también promueve el pensamiento computacional, la toma de decisiones, y la experimentación científica entre los estudiantes.

Según Sutton y Barto, los sistemas que pueden aprender a partir de recompensas y castigos simulan uno de los principios más naturales de la inteligencia: aprender del error. En el contexto del gato, esto se traduce en una experiencia pedagógica donde el algoritmo, los datos y el aprendizaje se entrelazan para formar una herramienta didáctica que enseña a aprender aprendiendo.

3. Diseño metodológico de la presente propuesta

Se elaboró la metodología con una perspectiva de Aprendizaje Basado en Proyectos (ABP) y constructivista, incorporando prácticas de programación en Python junto a elementos de Inteligencia Artificial (IA) y el empleo de

24 –30 de octubre de 2025

instrumentos digitales colaborativos. Se implementó este modelo en el sexto semestre de la carrera de Ingeniería en Informática, con la asignatura de Desarrollo de Aplicaciones Web, y se consideró un total de 47 alumnos.

El diseño metodológico se organizó en cuatro etapas sucesivas, lo que posibilitó el progreso desde la conceptualización a nivel teórico hasta la elaboración de prototipos funcionales de videojuegos con inteligencia artificial. Esto promovió el aprendizaje activo y el desarrollo de habilidades transversales.

3.1. Fase 1. Diseño de prácticas

Se crearon retos graduales en Python y guías de aprendizaje que están en línea con los temas de la materia Desarrollo de Aplicaciones Web. Los desafíos se elaboraron con fundamentos de gamificación (niveles, puntos, recompensas, retroalimentación inmediata), lo cual potencia la motivación y la persistencia de los alumnos [14]. Algunos ejemplos de desafíos son:

- Poner en práctica sistemas de puntuación que se adapten.
- Crear el juego del gato utilizando bibliotecas de aprendizaje automático como Scikit-learn, NumPy y TensorFlow para implementar algoritmos elementales de inteligencia artificial.

La razón de la implementación de estos retos es porque es necesario hacer de la programación algo contextualizado en situaciones reales, lo cual es un principio del aprendizaje basado en proyectos [15]. En la siguiente imagen se ilustra un reto que se realiza utilizando Padlet para almacenar la evidencia.

Reto 1: Dibujar el tablero

- Objetivo:** Comprender cómo se utilizan coordenadas y gráficos en Pygame.
- Tarea:** Modificar la función `dibujar_tablero()` para cambiar colores, grosor de los líneas y tamaño del tablero (ejemplo: poser de 3x3 a 4x4).
- Evidencia:** Captura del nuevo tablero modificado con el código en Padlet: <https://padlet.com/omaelmontero/reto1>

Figura 1. Ejemplo del reto implementando Quizziz para dar la instrucción y Padlet para visualizar la evidencia

3.2. Fase 2. Programación en Python e Inteligencia Artificial

Los estudiantes realizaron proyectos en Python, utilizando Pygame para diseñar interfaces gráficas que posibilitan la visualización de los algoritmos de aprendizaje automático en funcionamiento. Por su amplia gama de bibliotecas educativas y su sintaxis comprensible, Python ha sido considerado el lenguaje más conveniente para la enseñanza de inteligencia artificial [4].

La utilización de juegos gamificados posibilita la observación en tiempo real de cómo un algoritmo afecta el comportamiento de un personaje, lo cual mejora tanto la comprensión conceptual como la motivación del alumnado

[16]. Esta metodología tiene como fundamento el principio constructivista de aprender a través de la práctica [17]. La imagen que sigue muestra el juego del gato, el cual fue programado en Visual Studio Code. En esta fase, los estudiantes desarrollaron el juego del Gato (Tic-Tac-Toe) utilizando Python y las bibliotecas Pygame, NumPy, Scikit-learn y TensorFlow, con el propósito de comprender cómo los algoritmos de IA toman decisiones racionales y mejoran a partir de la experiencia [3].

- Se empleó Pygame para la interfaz gráfica, lo que hizo más fácil observar las jugadas y la lógica del agente.

- NumPy posibilitó que el tablero fuera modelado como una matriz de datos, lo cual facilitó la simulación de estrategias y el cálculo de posiciones potenciales.

El algoritmo Minimax, que indaga los movimientos disponibles, asigna valores de victoria, empate o derrota y elige la jugada más conveniente, fue el fundamento del sistema. De esa manera se representa la inteligencia artificial simbólica clásica [8].

Se creó una base de conocimiento en formato XML para suministrar información al agente de aprendizaje adaptativo, en la cual se documentan las jugadas, los resultados y las configuraciones del tablero.

Se creó una base de conocimiento en formato XML para suministrar información al agente de aprendizaje adaptativo, en la cual se documentan las jugadas, los resultados y las configuraciones del tablero. Esto hará posible que se consulten experiencias pasadas, se fortalezcan las estrategias que han tenido éxito y se castigue lo contrario, siguiendo los principios del aprendizaje por refuerzo simbólico [18].

Los alumnos utilizarán TensorFlow, Scikit-learn y NumPy en la Unidad 5 para examinar los datos guardados en XML a través de algoritmos como KNN y Decision Tree. También emplearán estas herramientas para poner en funcionamiento redes neuronales densas con entrenamiento iterativo, fusionando la base de datos con aprendizaje automático y servicios web. Esto posibilitará que el agente cambie de un comportamiento determinista (solo Minimax) a uno que se adapta (Minimax + XML + machine learning).

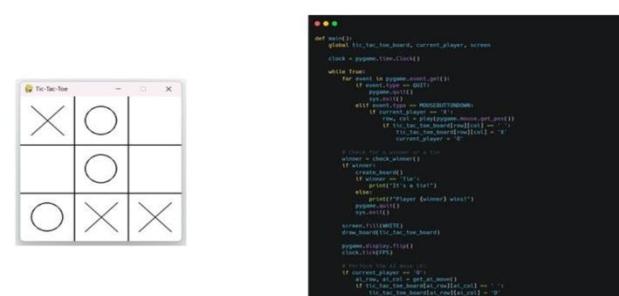


Figura 2. Ejemplo del juego del gato en Visual Code

3.3. Fase 3. Integración digital colaborativa

Se incluyeron plataformas digitales colaborativas para mejorar la comunicación, la retroalimentación y el trabajo en equipo:

- Microsoft Teams, para administrar proyectos y comunicarse de manera sincrónica y asincrónica [19].
- Padlet, en calidad de repositorio digital para el intercambio de prototipos y reflexiones [20].
- Mentimeter, para realizar encuestas y lluvias de ideas en tiempo real [21].
- Quizziz, para valoraciones de tipo gamificado que refuerzan el aprendizaje de forma divertida [23].

La incorporación de estas plataformas se basa en investigaciones que resaltan la eficacia de las herramientas digitales para crear comunidades de aprendizaje activo y colaborativo [10]. En la siguiente ilustración se muestra la valoración obtenida por el alumno al realizar el ejercicio del Gato en Python

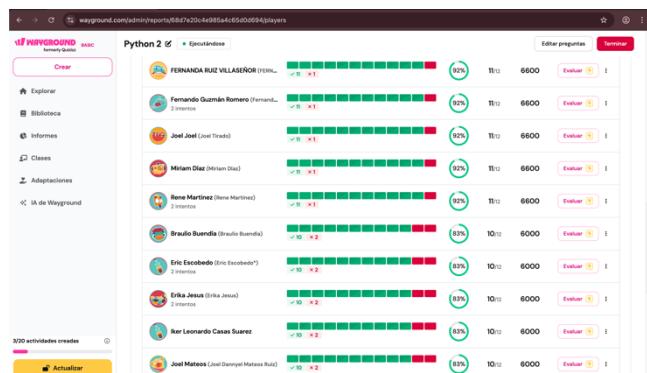


Figura 3. Valoración obtenida por los alumnos en Quizziz al implementar el Gato con Python

3.4. Fase 4. Evaluación y retroalimentación

La evaluación fue desarrollada utilizando un modelo mixto, que junta estrategias de tipo formativo y sumativo.

- Formativa:** autoevaluación y coevaluación a través de rúbricas, *feedback* en tiempo real con Quizziz y reflexión grupal utilizando Mentimeter.
- Sumativa:** entrega de un videojuego que funcione, con inteligencia artificial y una memoria técnica que detalla el procedimiento de desarrollo.

De acuerdo con Biggs y Tang [6], un método de evaluación que esté en consonancia con el aprendizaje significativo hace posible que se incremente la calidad de los productos finales, así como la motivación del alumno. Además, la triangulación de instrumentos como encuestas, rúbricas y análisis de participación digital ayuda a evaluar competencias transversales y técnicas como el trabajo en equipo, la creatividad y la comunicación [23]. En la siguiente ilustración se muestra cómo se realizó la evaluación en Mentimeter.



Figura 4. Resultados obtenidos en Mentimeter de la encuesta de satisfacción

4. Resultados

Al Implementar la gamificación con Python e inteligencia artificial (IA), así como herramientas digitales colaborativas, como Microsoft Teams, Mentimeter, Padlet y Quizziz, se obtendrá un efecto positivo en contraste con los métodos tradicionales de enseñanza que se enfocan en la exposición teórica y en la resolución individual de ejercicios. Tres aspectos mostraron el impacto positivo: la motivación del estudiante, la comprensión de los algoritmos de IA y el rendimiento académico.

Los alumnos, desde la perspectiva del Aprendizaje Basado en Proyectos (ABP), llevaron a cabo el juego del gato por medio de bibliotecas de Python como TensorFlow, Scikit-learn y NumPy. Para que el agente pudiera tomar decisiones racionales y optimizar su rendimiento conforme adquiría experiencia, utilizaron el algoritmo Minimax y modelos de aprendizaje supervisado (KNN y Decision Tree). Esta integración posibilitó ver cómo los conceptos teóricos de IA se convierten en procesos que pueden ser visualizados dentro del juego, lo cual refuerza el entendimiento del modo en que operan los algoritmos.

Del total de 47 participantes, 34 estudiantes (72.3%) terminaron el proyecto durante la primera sesión (2,5 horas); 7 alumnos (14.9%), en una segunda sesión (~4 horas); y únicamente 6 alumnos (12.8%) no lograron cumplir con el objetivo en el tiempo estipulado (Gráfica 1 y Tabla 1). por el contrario, en cohortes anteriores del mismo curso, donde no se utilizaron métodos de programación con inteligencia artificial ni de gamificación, únicamente el 54.8% de los alumnos consiguieron terminar la actividad en la primera sesión. Esto es un incremento del 17.5% en el índice de logro inicial, que se debe a la mejor comprensión y motivación resultantes de la perspectiva inteligente y gamificada.

Tabla 1. Resumen del Porcentaje de logros obtenidos en la sesión de la programación del juego del gato

Indicador	Número de alumnos	Porcentaje
Total de estudiantes	47	100%
Logro en primera sesión (2.5 h)	34	72.3%
Logro en segunda sesión (~4 h)	7	14.9%
Estudiantes sin logro	6	12.8%



Grafica 1. Porcentaje de logros obtenidos en la sesión de la programación del juego del gato

Los descubrimientos indican que la aplicación de IA en la programación de videojuegos no solo favorece una mejor comprensión conceptual, sino que también promueve el razonamiento lógico-computacional y la solución de problemas complejos [24]. Los estudiantes, al observar la evolución del agente dentro del ambiente de juego, construirán el conocimiento a través de una experimentación activa, lo cual está en línea con las teorías constructivistas de Piaget (1970) y con el aprendizaje sociocultural de Vygotsky (1978).

Estos resultados evidencian que la gamificación con Python e IA favorece la adquisición de habilidades técnicas y competencias transversales, aunque la diferencia en tiempos de logro sugiere la necesidad de apoyo adicional para estudiantes con menor experiencia previa. Además, la proporción de alumnos que no completó la actividad indica que el diseño pedagógico y la orientación personalizada son factores clave para maximizar el éxito.

Desde un enfoque crítico, el modelo demuestra avances en la incorporación de métodos activos y tecnologías emergentes, pero requiere evaluación continua y capacitación docente, para asegurar que las TIC generen un impacto real en el aprendizaje [25].

5. Conclusiones

El desarrollo de la experiencia educativa para la Unidad 5: Implementación de la Inteligencia Artificial en ambientes de programación evidenció que aplicar gamificación respaldada por IA es una táctica eficaz para reforzar las habilidades tecnológicas y el pensamiento lógico-computacional en los alumnos de educación superior.

Capaces de observar cómo la inteligencia artificial puede tomar decisiones autónomas en situaciones controladas y lúdicas, al aplicar conceptos teóricos y prácticos de algoritmos inteligentes mediante el diseño del juego del gato con Python.

Los resultados alcanzados muestran un impacto positivo en comparación con los métodos convencionales, lo que se demuestra con un aumento del 17.5% en el índice de logro académico y una percepción positiva por parte del alumno (91% de respuesta afirmativa en encuestas).

Estas mejoras se deben a la gamificación, que fomenta un enfoque activo, colaborativo y reflexivo, apoyado en plataformas digitales como Quizziz, Mentimeter y Padlet.

Estas herramientas permiten una participación colectiva y una retroalimentación instantánea a la gamificación, que fomenta un enfoque activo, colaborativo y reflexivo, apoyado en plataformas digitales como Quizziz, Mentimeter y Padlet. Estas herramientas permiten una participación colectiva y una retroalimentación instantánea.

Desde el punto de vista pedagógico, esta experiencia demuestra que la integración con tecnologías emergentes como la inteligencia artificial (IA) potencia el aprendizaje basado en proyectos para fomentar habilidades cognitivas superiores.

El alumno no solamente se dedica a programar, sino que también genera conocimiento a través de la experimentación, entendiendo los procesos internos del razonamiento algorítmico. En esta línea, la inteligencia artificial funciona como un mediador pedagógico y cognitivo, de acuerdo con las ideas constructivistas de Piaget [19] y el enfoque sociocultural de Vygotsky [27].

En la actualidad, el trabajo está en una segunda etapa que se enfoca en poner en práctica los algoritmos de inteligencia artificial (como el Minimax y los modelos de aprendizaje supervisados como K-Nearest Neighbors y Decision Tree), además de utilizar XML para almacenar datos estructurados y garantizar que sean interoperables. Esta fase posibilitará la evaluación de la eficacia del sistema y la habilidad de la IA para aprender a partir de las acciones del usuario, lo que reforzará al proyecto como un instrumento educativo con un alto valor formativo.

En conclusión, la Unidad 5 es un espacio relevante para combinar la programación, la inteligencia artificial y el manejo de datos, todo ello enmarcado en un modelo de aprendizaje activo que se basa en juegos y está dirigido a proyectos. La experiencia demuestra que la inteligencia artificial aplicada en ambientes educativos no solo potencia el desempeño académico, sino que además fortalece la creatividad, el pensamiento crítico y la autonomía, habilidades fundamentales para preparar a los profesionales del siglo XXI.

6. Referencias

- [1] Abadi, M., et al. (2016). TensorFlow: Large-scale machine learning on heterogeneous systems. Google Research. Recuperado de <https://www.tensorflow.org>
- [2] Al-Emran, M. (2022). Padlet for collaborative learning: A review of literature. *Education and Information Technologies*, 27(5), 6573–6593. <https://doi.org/10.1007/s10639-022-11066-4>
- [3] Alsawaiyer, R. S. (2020). The effect of gamification on motivation and engagement. *International Journal of Information and Learning Technology*, 37(3), 193–210. <https://doi.org/10.1108/IJILT-02-2020-0028>
- [4] Barto, A. G., & Sutton, R. S. (2018). *Reinforcement Learning: An Introduction* (2^a ed.). Massachusetts Institute of Technology Press.

- [5] Bell, S. (2019). Project-based learning for the 21st century: Skills for the future. *The Clearing House*, 92(2), 39–43. <https://doi.org/10.1080/00098655.2019.1586611>
- [6] Biggs, J., & Tang, C. (2011). *Teaching for Quality Learning at University* (4^a ed.). Open University Press. Bower, M. (2019). *Design of Technology-Enhanced Learning: Integrating Research and Practice*. Emerald Publishing.
- [7] Cabero, J., & Llorente, M. C. (2013). La aplicación de las TIC en la educación: Innovación y retos. Editorial Síntesis.
- [8] Deterding, S., Björk, S., Nacke, L., Dixon, D., & Lawley, E. (2019). Designing gamification: Creating gameful and playful experiences. *Proceedings of the Annual Symposium on Computer-Human Interaction in Play*, 10–19. <https://doi.org/10.1145/3311350.3347193>
- [9] Domínguez, A., Saenz-de-Navarrete, J., de-Marcos, L., Fernández-Sanz, L., & Pagés, C. (2023). Gamifying learning experiences: Practical implications and outcomes in higher education. *Computers & Education*, 193, 104649. <https://doi.org/10.1016/j.compedu.2022.104649>
- [10] Géron, A. (2019). *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media. Grover, S., & Pea, R. (2018). Computational thinking: A competency whose time has come. *Computer Science Education*, 28(2), 70–94. <https://doi.org/10.1080/08993408.2018.1459066>
- [11] Hamari, J., Shernoff, D. J., Rowe, E., Coller, B., & Asbell-Clarke, J. (2019). Challenging games help students learn: An empirical study on engagement, flow, and immersion in game-based learning. *Computers in Human Behavior*, 92, 348–356. <https://doi.org/10.1016/j.chb.2018.10.023>
- [12] Laurillard, D. (2012). *Teaching as a Design Science: Building Pedagogical Patterns for Learning and Technology*. Routledge.
- [13] Lee, J., & Martin, L. (2024). Active learning through digital collaboration platforms. *Journal of Interactive Learning Research*, 35(2), 150–168.
- [14] Müller, A. C., & Guido, S. (2020). *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media.
- [15] Nouri, J., Zhang, L., Mannila, L., & Norén, E. (2020). Motivation and learning in digital game-based learning environments: A systematic review. *Educational Technology Research and Development*, 68(4), 2053–2081. <https://doi.org/10.1007/s11423-020-09779-8>
- [16] Pedregosa, F., et al. (2022). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(85), 2825–2830.
- [17] Piaget, J. (1970). *Psychology and Pedagogy*. Viking Press.
- [18] Raschka, S., & Mirjalili, V. (2021). *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-learn, and TensorFlow 2*. Packt Publishing.
- [19] Rashid, T. (2021). *Make Your Own Neural Network*. CreateSpace Independent Publishing.
- [20] Russell, S., & Norvig, P. (2022). *Artificial Intelligence: A Modern Approach* (4^a ed.). Pearson. Schwab, K. (2018). *The Fourth Industrial Revolution*. Currency.
- [21] Su, C. H., & Cheng, C. H. (2022). A mobile gamification learning system for improving learning motivation and achievements. *Journal of Computer Assisted Learning*, 38(2), 312–326. <https://doi.org/10.1111/jcal.12624>
- [22] Susanti, N., et al. (2023). Enhancing student engagement through Mentimeter: A study on interactive learning. *Education and Information Technologies*, 28(5), 5345–5362.
- [23] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement Learning: An Introduction* (2^a ed.). MIT Press.
- [24] Thomas, J. W. (2020). *A Review of Research on Project-Based Learning*. The Autodesk Foundation.
- [25] Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2023). Computational thinking in compulsory education: Toward an agenda for research and practice. *Education and Information Technologies*, 28(1), 101–124. <https://doi.org/10.1007/s10639-022-11189-8>
- [26] Vygotsky, L. S. (1978). *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press.
- [27] Xu, M., David, J. M., & Kim, S. H. (2021). The Fourth Industrial Revolution: Opportunities and challenges. *International Journal of Financial Research*, 12(3), 76–88. <https://doi.org/10.5430/ijfr.v12n3p76>