

# Asistente Virtual Inteligente para estudiantes basado en planificación personalizada y evaluación diagnóstica

## Intelligent Virtual Assistant for students based on personalized scheduling and diagnostic assessment

Mateo Arauz<sup>1</sup>, Carlos Robles<sup>2</sup>, Cristhian Wu<sup>3</sup>, Daniel Troetsch<sup>4</sup>, Daniel Vega<sup>5</sup>, Vladimir Villarreal<sup>6</sup>, Miguel Chavarría<sup>7</sup>

Arauz, M; Robles, C; Wu, C; Troetsch, D; Vega, D; Villarreal, V; Chavarría, M. Asistente virtual inteligente para estudiantes basado en planificación personalizada y evaluación diagnóstica. *Tecnología en Marcha*. Vol. 39 N° especial. Junio, 2026. VIII Congreso Internacional en Inteligencia Ambiental, Ingeniería de Software, Salud Electrónica y Móvil (AmITIC). Pág. 114-121.

 <https://doi.org/10.18845/tm.v39i7.8749>

- 1 Universidad Tecnológica de Panamá. Panamá.  
 [mateo.arauz@utp.ac.pa](mailto:mateo.arauz@utp.ac.pa)  
 <https://orcid.org/0009-0009-2041-7333>
- 2 Universidad Tecnológica de Panamá. Panamá.  
 [carlos.robles@utp.ac.pa](mailto:carlos.robles@utp.ac.pa)  
 <https://orcid.org/0009-0001-9837-737X>
- 3 Universidad Tecnológica de Panamá. Panamá.  
 [cristhian.wu@utp.ac.pa](mailto:cristhian.wu@utp.ac.pa)  
 <https://orcid.org/0009-0009-7065-2024>
- 4 Universidad Tecnológica de Panamá. Panamá.  
 [daniel.troetsch@utp.ac.pa](mailto:daniel.troetsch@utp.ac.pa)  
 <https://orcid.org/0009-0004-7629-8023>
- 5 Universidad Tecnológica de Panamá. Panamá.  
 [daniel.vega4@utp.ac.pa](mailto:daniel.vega4@utp.ac.pa)  
 <http://orcid.org/0009-0008-4346-2420>
- 6 Grupo de Investigación en Tecnologías Computacionales Emergentes. Panamá.  
 [vladimir.villarreal@utp.ac.pa](mailto:vladimir.villarreal@utp.ac.pa)  
 <https://orcid.org/0000-0003-4678-5977>
- 7 Grupo de Investigación en Tecnologías Computacionales Emergentes. Panamá  
 [miguel.chavarria@utp.ac.pa](mailto:miguel.chavarria@utp.ac.pa)  
 <https://orcid.org/0000-0002-4489-235X>



## Palabras clave

Aprendizaje por refuerzo; asistente virtual; evaluación diagnóstica; inteligencia artificial; planificación personalizada.

## Resumen

Se desarrolla TutorIA, un asistente virtual inteligente cuyo objetivo es mejorar el rendimiento académico de los estudiantes universitarios mediante recomendaciones personalizadas de estudio. Esta iniciativa surge ante las dificultades que enfrentan muchos estudiantes para organizar su tiempo, seleccionar materiales adecuados y adaptarse a las crecientes exigencias académicas. El sistema se implementa bajo una arquitectura de microservicios, donde se diseñó un servicio backend encargado de gestionar los datos del usuario y del agente inteligente, mientras que estos servicios son consumidos por una aplicación móvil desarrollada en Flutter. El agente integra un modelo de aprendizaje por refuerzo, utilizando el algoritmo Deep Q-Learning, el cual adapta progresivamente las guías de estudio en función del desempeño del estudiante. Dicho desempeño es evaluado mediante pruebas propuestas por el propio agente, lo que permite identificar áreas de dificultad y ajustar las recomendaciones futuras de forma personalizada. El resultado de este proyecto ha cumplido en gran medida con los objetivos iniciales, entre ellos: la generación de una prueba de evaluación diagnóstica, el desarrollo de una máquina capaz de aprender de los errores del estudiante, la implementación de mecanismos para explicar contenidos específicos de cada asignatura, y la creación de un horario de estudio personalizado adaptado a la disponibilidad y necesidades del usuario.

## Keywords

Artificial intelligence; diagnostic assessment; personalized planning; reinforcement learning; virtual assistant

## Abstract

TutorIA, an intelligent virtual assistant, is being developed to improve the academic performance of university students through personalized study recommendations. This initiative arises from the difficulties many students face in organizing their time, selecting appropriate materials, and adapting to increasing academic demands. The system is implemented using a microservices architecture, where a backend service is designed to manage user and intelligent agent data, while these services are consumed by a mobile application developed in Flutter. The agent integrates a reinforcement learning model using the Deep Q-Learning algorithm, which progressively adapts study guides based on student performance. This performance is evaluated using tests proposed by the agent itself, allowing for the identification of areas of difficulty and the adjustment of future recommendations in a personalized manner. The result of this project has largely met its initial objectives, including: the generation of a diagnostic assessment test, the development of a machine capable of learning from student errors, the implementation of mechanisms to explain subject-specific content, and the creation of a personalized study schedule tailored to the user's availability and needs.

## Introducción

El rendimiento académico de los estudiantes es un factor clave para su éxito tanto académico como profesional. En un contexto donde las exigencias educativas crecen constantemente, resulta fundamental implementar estrategias efectivas para gestionar el tiempo de estudio

y seleccionar recursos adecuados que promuevan resultados óptimos. En este sentido, la inteligencia artificial (IA) ha emergido como una herramienta prometedora para optimizar los procesos de aprendizaje. En el ámbito educativo, la IA permite desarrollar entornos de aprendizaje adaptativos que se ajustan a las necesidades individuales de los estudiantes, maximizando la adquisición de conocimientos [1]. Asimismo [2] destaca que la IA ofrece amplias oportunidades en el entorno universitario, facilitando avances en diversas áreas del conocimiento y apoyando a los estudiantes en la construcción de teorías y aplicaciones científicas, especialmente en un contexto de cambios acelerados en la sociedad del conocimiento. Con esto en mente, desarrollamos TutorIA, un asistente virtual inteligente diseñado para mejorar el rendimiento académico mediante recomendaciones personalizadas de horarios de estudio, sugerencias de materiales relevantes por módulo, identificación de debilidades en el aprendizaje, generación de evaluaciones adaptativas y seguimiento continuo del progreso académico.

## Planteamiento del problema

En el entorno universitario, las crecientes exigencias académicas y la reducción del contacto directo entre docentes y estudiantes pueden impactar negativamente la motivación de los alumnos. Además, muchos estudiantes enfrentan dificultades para organizar adecuadamente su tiempo de estudio, seleccionar materiales relevantes y comprender a fondo los contenidos [3]. Con el desarrollo de herramientas tecnológicas, se abre un abanico de posibilidades para personalizar el aprendizaje, adaptando los contenidos a las necesidades individuales de cada estudiante [4]. En consecuencia, surge la necesidad de un sistema que no solo proporcione recomendaciones de estudio personalizadas, sino que también genere evaluaciones adaptativas basadas en el rendimiento y el material específico de cada estudiante. Por ello, se desarrolló TutorIA, un agente de inteligencia artificial basado en técnicas de aprendizaje por refuerzo, que brinde un soporte preciso, flexible y adaptado a las demandas del entorno académico actual.

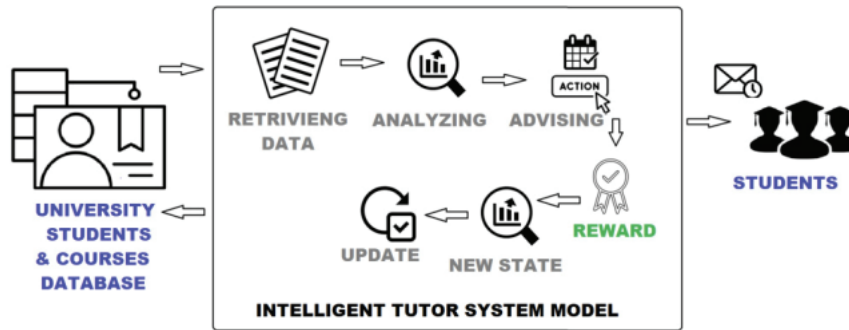
## Marco teórico

A continuación, se presentan trabajos previos que han servido de base para el desarrollo de TutorIA, destacando sus hallazgos y enfoques similares:

### Sistema inteligente de Programación de exámenes personalizada mediante aprendizaje por refuerzo

El trabajo titulado Reinforcement Learning based Intelligent System for Personalized Exam Schedule propone un sistema que utiliza aprendizaje por refuerzo (RL) para generar calendarios de exámenes personalizados. Este enfoque considera que cada estudiante tiene características únicas, como edad, nivel de disciplina, rendimiento académico y número de asignaturas, que influyen en su desempeño. El sistema entrena un agente de RL para optimizar los horarios según patrones de éxito del estudiante, con el objetivo de reducir el estrés académico y aumentar la probabilidad de aprobar los exámenes.

El modelo fue entrenado con datos simulados que representan diversos perfiles estudiantiles y su comportamiento frente a diferentes calendarios. Durante el entrenamiento, el agente identificó combinaciones óptimas de tiempo y carga académica para maximizar el rendimiento. Aunque los resultados son preliminares, se reportó una mejora en la organización del tiempo y una reducción en las tasas de fracaso académico. Este sistema muestra potencial para su implementación en entornos educativos reales, previa validación en pruebas de campo.



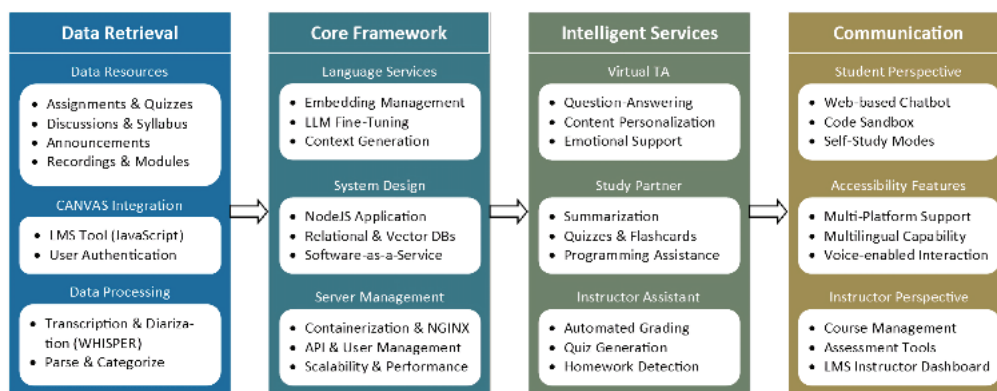
**Figura 1.** Ciclo dinámico basado en datos.

La figura 1 representa un ciclo dinámico basado en datos, donde el agente de RL genera horarios personalizados integrando información de los estudiantes, interacciones con el entorno y retroalimentación para optimizar la planificación de exámenes [5].

### Asistente Virtual Inteligente para Aprendizaje Personalizado

Este trabajo analiza un asistente virtual inteligente diseñado para ofrecer soporte personalizado en el aprendizaje. Este asistente utiliza inteligencia artificial para interpretar consultas en lenguaje natural, generar cuestionarios, tarjetas de estudio y recomendaciones de contenido adaptadas a las necesidades individuales. Su diseño busca mejorar la motivación y reducir la sobrecarga cognitiva, actuando como un tutor personalizado disponible en todo momento. La arquitectura modular del sistema facilita su integración con plataformas educativas existentes.

El asistente adapta sus respuestas y materiales en función del progreso del usuario y sus áreas de dificultad, utilizando un sistema de seguimiento del rendimiento para actualizar dinámicamente el contenido. Además, incorpora técnicas de procesamiento del lenguaje natural (PLN) para interpretar preguntas abiertas y generar explicaciones contextualizadas. Probado con estudiantes universitarios, el sistema mostró mejoras en la retención de información y la satisfacción del usuario [6].



**Figura 2.** Arquitectura del Sistema de VirtualTA.

La figura 2 describe un sistema educativo inteligente que integra un LMS como Canvas con IA para ofrecer asistentes virtuales (tutor, estudio, instructor), procesar contenido automáticamente (transcripciones, cuestionarios, calificaciones) y brindar soporte multilingüe y accesible en una plataforma escalable basada en NodeJS y bases de datos vectoriales [6].

## **Materiales y métodos (metodología)**

Para el desarrollo de TutorIA, se utilizan herramientas de desarrollo, entornos integrados y tecnologías complementarias que garantizan una integración eficiente entre el frontend, el backend, los algoritmos de inteligencia artificial y la gestión de datos. A continuación, se describen los materiales y métodos seleccionados:

Entornos de desarrollo (IDEs):

- Android Studio: Utilizado para pruebas, emulación y depuración de la aplicación móvil desarrollada con Flutter [7].
- Visual Studio Code: Empleado para el desarrollo de la lógica del proyecto, la creación y gestión de bases de datos, así como la implementación de modelos de aprendizaje automático [8].

Frameworks y herramientas:

- Flutter: Framework para el desarrollo de interfaces gráficas multiplataforma (iOS y Android) [9].
- Django: Framework de alto nivel en Python para la lógica del servidor [10].
- Docker: Utilizado para gestionar servicios principales (backend, base de datos, módulos de IA) en contenedores, asegurando portabilidad y consistencia entre desarrollo, pruebas y producción [11].

Bases de datos:

- MySQL: Base de datos relacional para almacenar información estructurada, como datos de usuarios, horarios, asignaturas y resultados de evaluaciones [12].
- Weaviate: Base de datos vectorial para el almacenamiento y recuperación eficiente de embeddings semánticos, utilizada en consultas inteligentes, búsqueda semántica y recomendación basada en contexto mediante técnicas de procesamiento del lenguaje natural (PLN) [13].

Lenguajes de programación:

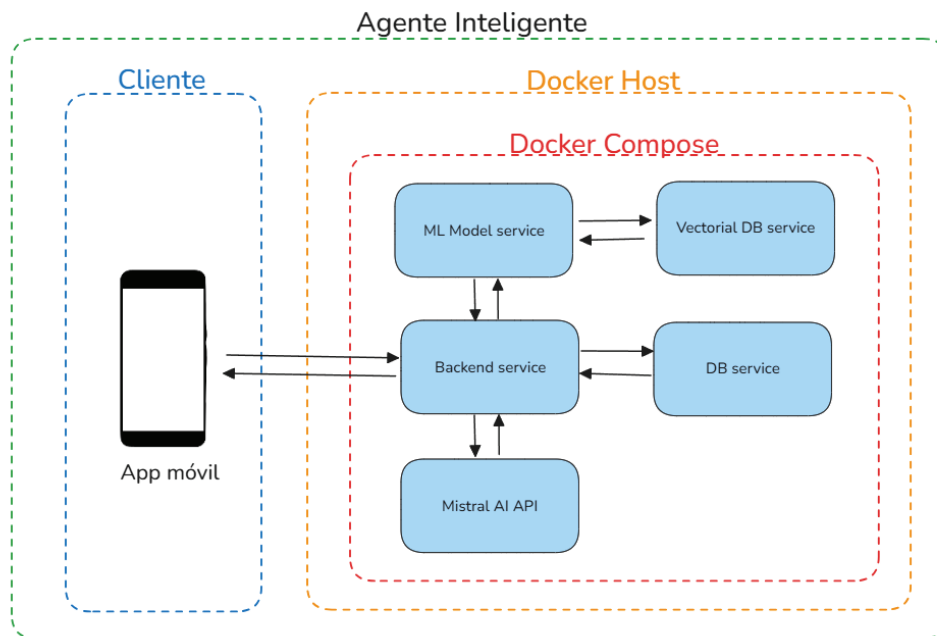
- Python: Lenguaje principal para el backend, procesamiento de datos y desarrollo de modelos de inteligencia artificial, incluyendo el modelo de aprendizaje por refuerzo [14].
- Dart: Lenguaje utilizado con Flutter para el desarrollo de la interfaz móvil, incluyendo vistas, navegación y comunicación con el backend [15].

## **Modelo de Aprendizaje Automático**

Se implementa un modelo de aprendizaje por refuerzo para adaptar los métodos de estudio según los aciertos y errores del estudiante. Se evalúan algoritmos como Deep Q-Learning (DQN) o Proximal Policy Optimization (PPO), que han demostrado eficacia en entornos de toma de decisiones secuenciales y aprendizaje basado en recompensas.

La figura 3 muestra la arquitectura del agente inteligente, que interactúa con un cliente móvil y servicios desplegados mediante Docker Compose en un Docker Host. El flujo es el siguiente:

- La aplicación móvil actúa como la interfaz principal, enviando y recibiendo solicitudes al *backend* mediante peticiones HTTP.
- El *backend* coordina operaciones entre servicios, como la base de datos y el modelo de aprendizaje automático.
- El servicio de aprendizaje automático procesa textos escritos por el estudiante y genera respuestas adaptadas a sus necesidades.
- La base de datos vectorial almacena *embeddings* de textos para consultas inteligentes.
- La base de datos relacional gestiona datos de usuarios y del sistema.
- La API de Mistral AI [16], accedida a través del *backend*, responde a consultas específicas de los estudiantes.



**Figura 3.** Diagrama de arquitectura de TutorIA.

El desarrollo utiliza una arquitectura de microservicios, que divide el sistema en componentes independientes para facilitar la escalabilidad y el mantenimiento. Además, se aplica la metodología ágil Scrum, adaptada a las características del equipo, para trabajar de forma iterativa, colaborativa y con entregas frecuentes, promoviendo la mejora continua.

## Resultados

El desarrollo de TutorIA cumple con los objetivos iniciales del proyecto: implementar una aplicación móvil multiplataforma que permita a los estudiantes evaluar y mejorar sus conocimientos en asignaturas específicas. La integración de aprendizaje por refuerzo y técnicas de inteligencia artificial ha sido fundamental para generar evaluaciones adaptativas que miden el conocimiento previo y el aprendizaje posterior a las sesiones de estudio.

Se espera que TutorIA fomente hábitos de estudio saludables y rutinas eficientes, ayudando a los estudiantes a profundizar en su aprendizaje de manera progresiva mediante guías personalizadas. Durante la fase de desarrollo, se implementan con éxito los siguientes componentes:

- Sistema de autenticación: Basado en correo electrónico y contraseña, permite el registro, inicio de sesión seguro y una experiencia personalizada.
- Base de datos relacional: Organiza y almacena datos de usuarios, cursos, registros de conversaciones y sesiones de estudio, garantizando un acceso eficiente y un mantenimiento adecuado.
- Sistema de interacción con el asistente virtual: Gestiona y almacena conversaciones entre el usuario y el asistente, utilizando la API de Mistral AI para generar respuestas contextualizadas a las consultas de los estudiantes. Debido a limitaciones de hardware, se optó por una API externa, lo que reduce la personalización de las respuestas, pero asegura un rendimiento estable durante la fase de desarrollo.

En cuanto al desarrollo del modelo de aprendizaje por refuerzo dentro de TutorIA, se desarrolla el funcionamiento de adaptación que ajusta dinámicamente las guías de estudio según el desempeño del estudiante. El proceso comienza una vez el estudiante introduce su plan de estudio y define la duración de cada módulo. Con esta información, el sistema organiza un horario personalizado que distribuye los temas del módulo en sesiones de estudio, ofreciendo materiales adecuados para cada uno.

Al finalizar el periodo establecido, se genera un quiz con el objetivo de medir el nivel de comprensión alcanzado por el estudiante. Los resultados de esta evaluación son analizados por el modelo para identificar que temas presentan mayor dificultad y a partir de dichos datos, se reorganizan las futuras recomendaciones de estudio. Cuando el estudiante muestra un bajo desempeño, se refuerzan dichos contenidos.

Este ciclo de planificación, evaluación y adaptación se repite de manera continua para lograr una experiencia de aprendizaje más personalizada y efectiva. De este modo, TutorIA no solo busca evaluar el conocimiento del estudiante, sino que lo guía en base a su progreso real, facilitando una mejora constante y se centra en sus necesidades específicas.

## Conclusiones y/o recomendaciones (discusión)

Después de los resultados obtenidos se concluye que la utilización de la inteligencia artificial y los sistemas que aprenden, actualmente influyen en el aprendizaje, la investigación, la obtención de datos y la creación y generación de multimedia ha generado un gran impacto en la vida de los estudiantes tanto para los profesionales en el área de computación.

Al terminar de desarrollar este proyecto, se afirma que dicho impacto se puede redireccionar específicamente hacia el área de aprendizaje, con el fin de apoyar de manera activa la adquisición de conocimientos, la organización del estudio y la personalización del aprendizaje. El uso de modelos inteligentes que analizan el progreso del estudiante posibilita el poder ofrecer recomendaciones, ejercicios y recursos de acuerdo con su nivel y ritmo, lo cual fomenta la adopción de mejores rutinas. De este modo, la inteligencia artificial no solo actúa como una herramienta para responder consultas, sino como un asistente que guía y refuerza el proceso de aprendizaje de forma significativa.

## Agradecimientos

V. Villarreal es miembro del Sistema Nacional de Investigación de la SENACYT en Panamá, agradecemos su ayuda y corrección de errores, además de las adecuaciones y comentarios para mejoras actuales tanto futuras.

## Referencias

- [1] R. D. M. Padilla, «La llegada de la inteligencia artificial a la educación,» *RITI*, vol. 7, n° 14, pp. 260-270, 2019.
- [2] K. S. Montes, «Uso de la inteligencia artificial en la educación superior entre el 2018 y el 2023. Una revisión sistemática.,» 2023.
- [3] A. Ezcurra, «Diagnóstico preliminar de las dificultades de los alumnos de primer ingreso a la educación superior,» *Perfiles educativos*, vol. 27, n° 107, pp. 118-133, 2005.
- [4] E. U. B. Pozo, «Innovación Pedagógica en la Educación Superior: Retos y Oportunidades en la Era Digital,» *Prosperus*, vol. 2, n° 2, pp. 84-97, 2025.
- [5] M. C. M. U. Z. & P. A. Barone, «Reinforcement Learning based Intelligent System for Personalized Exam Schedule,» *2024 19th Conference on Computer Science and Intelligence Systems (FedCSIS)*, pp. 549-553, 2024.
- [6] R. S. Y. C. M. C. D. & D. I. Sajja, «Artificial intelligence-enabled intelligent assistant for personalized and adaptive learning in higher education.,» *Information*, vol. 15, n° 10, p. 596, 2024.
- [7] Google, «Andoid Studio,» 2025. [En línea]. Available: [https://developer.android.com/studio?gad\\_campaignid=21831783795&hl=es-419](https://developer.android.com/studio?gad_campaignid=21831783795&hl=es-419).
- [8] Microsoft Corporation, «Visual Studio Code,» [En línea]. Available: <https://code.visualstudio.com/>.
- [9] Google, «Flutter,» 2025. [En línea]. Available: <https://flutter.dev/>.
- [10] Django Software Foundation, «Django,» 2025. [En línea]. Available: <https://www.djangoproject.com/>.
- [11] Docker Inc., «Docker,» 2025. [En línea]. Available: <https://www.docker.com/>.
- [12] Oracle Corporation, «MySQL,» 2025. [En línea]. Available: <https://www.mysql.com/>.
- [13] Semi Technologies, «Weavite,» 2025. [En línea]. Available: <https://weaviate.io/>.
- [14] Python Software Foundation, «Python,» 2025. [En línea]. Available: <https://www.python.org/>.
- [15] Google, «Dart,» 2025. [En línea]. Available: <https://dart.dev/>.
- [16] Mistral AI, «MistralAI,» 2025. [En línea]. Available: <https://mistral.ai/>.

## Declaración sobre uso de Inteligencia Artificial (IA)

Para la revisión gramatical y ortográfica de este artículo, así como para la traducción de textos empleamos la herramienta de IA *ChatGPT*. Esta nos permitió identificar errores y mejorar la fluidez del texto. No obstante, realizamos una revisión final para garantizar que el artículo cumpliera con los estándares de calidad de la revista.