
This is the **published version** of the bachelor thesis:

Corral Ball, Sebastián; Valle López, Juan Carlos , dir. EduGPT. Plan de Desarrollo SaaS para Automatización Académica con IA. Integración en Plataforma Web Multiservicio. 2025. (Grau en Gestió de Ciutats Intel·ligents i Sostenibles)

This version is available at <https://ddd.uab.cat/record/315769>

under the terms of the  license

EduGPT

PLAN DE DESARROLLO

SaaS para Automatización Académica con IA: Integración en Plataforma Web Multiservicio

Sebastián Corral Ball

Resumen: EduGPT es una solución basada en inteligencia artificial que responde a la creciente necesidad de automatizar tareas repetitivas en la gestión académica, como la redacción, el análisis o la validación de documentos. Está orientada a estudiantes, docentes y profesionales de sectores con alta carga documental. Su diseño modular permite integrar funciones como resúmenes, validación, detección de IA y generación de tesis en una única plataforma. A diferencia de herramientas genéricas, ofrece resultados estructurados con lógica académica. EduGPT se plantea como un SaaS escalable, con suscripciones según el tipo de usuario. Actualmente se encuentra en fase de investigación, explorando cómo estructurar el conocimiento académico mediante agentes inteligentes y NLP. El objetivo a largo plazo es convertirse en referencia en automatización educativa con IA, abarcando desde la generación de trabajos hasta la corrección y trámites administrativos automatizados.

El trabajo sigue la siguiente estructura:

Introducción – Contexto

1. Objetivos

- 1.1. Herramienta y colaboración
- 1.2. Innovación
- 1.3. Profesores
- 1.4. Alumnos

2. Funcionalidades

3. Tecnologías

- 3.1. FrontEnd
- 3.2. BackEnd
- 3.3. Bases de Datos
- 3.4. IA - Corrección con GPT

4. Método de evaluación

5. Despliegue y presentación

6. Teoría

- 6.1. Revisión bibliográfica (estado actual de la IA para esto)
- 6.2. Desarrollo de la herramienta (construcción de un MVP “sacar fotos de facturaGPT”)
- 6.3. Fases de prueba (ejemplos, estadísticas, rendimiento)

7. Conclusiones

- 7.1. Análisis de ventajas y desventajas
- 7.2. Posibles mejoras en un futuro
- 7.3. Riesgos éticos y limitaciones

Agradecimientos

Referencias

INTRODUCCIÓN - CONTEXTO

LA manera en que interpretamos la realidad no es objetiva, sino una construcción basada en el mapa mental que cada persona ha desarrollado a partir de sus experiencias, creencias y conocimientos. Este mapa, sin embargo, no es el territorio: no representa la realidad en sí, sino una interpretación subjetiva. Comprender esto es el primer paso hacia un cambio de paradigma, una transformación profunda en la forma de pensar, actuar y crear. [4]

En este marco, la dialéctica hegeliana se convierte en una herramienta clave. Según Hegel, el avance de la realidad y del pensamiento, es decir, el cambio de paradigma, ocurre mediante una secuencia de tesis, antítesis y síntesis. La síntesis no es una simple combinación de ideas previas, sino una superación que permite integrar contradicciones aparentes de dos mapas en una visión más amplia[10]. En este proyecto, podemos afirmar que la educación tradicional centrada en la producción manual como desarrollo capacitativo del ser humano (tesis) y el uso de la IA para maximizar la velocidad y la cantidad de producción (antítesis) encuentran en EduGPT una síntesis posible: una tecnología que potencia lo humano y optimiza y regula el uso de la IA.

Con esta perspectiva, entendemos el Valor real (V) como un binomio inseparable: la Producción (P), entendida como resultados tangibles, y la Capacidad de Producción (CP), aquello que permite sostener esos resultados de forma continua[4]. Desde esa base, se aprende a distinguir entre soluciones miopes [8]—que sólo centran la atención en la P y generan dependencia— y soluciones profundas —que fortalecen la CP y crean estructuras de P sostenibles—. La CP es el bienestar y propósito internos provocados por la sensación de que uno se mueve hacia donde quiere ir. Es decir: **CP = HI (Human Intelligence)**. La producción, siempre consecuencia directa de la HI, puede ser potenciada mediante herramientas externas como la inteligencia artificial, lo que nos lleva a decir que según el HI y su habilidad para dominar la IA: $V = HI + IA$. o $V = HI * IA$. o

$$V = HI^{IA}$$

. [9]

EduGPT está pensado desde la importancia de la administración del tiempo, con el “Cuadrante II”: actividades no urgentes e importantes, como un sistema que libera a la persona de tareas repetitivas y superficiales para que pueda invertir su tiempo en lo verdaderamente valioso (CP)[4].

EduGPT es una autopista de doble carril hacia el HI:

1. **Reduce el tiempo dedicado a tareas repetitivas**, lo que permite invertir ese tiempo en relaciones humanas, deporte, descanso y desarrollo personal.
2. **Permite que el tiempo de trabajo se centre en *deep work*** —tareas cognitivamente exigentes que entrenan la atención, fortalecen la presencia mental y generan estados de flujo—. En ese sentido, EduGPT no solo

mejora lo que hacemos, sino cómo y para qué lo hacemos. Su objetivo es sacar el máximo partido de la frase del psicólogo Robin Sharma: *“la mente es un magnífico criado, pero un amo terrible”*[18].

En una era donde la IA se presenta como solución a todo, es esencial volver a situar el centro en el ser humano. La inteligencia artificial puede aumentar la producción, pero solo cuando está al servicio de una capacidad humana bien cultivada para así conseguir el máximo beneficio de Valor real (V).

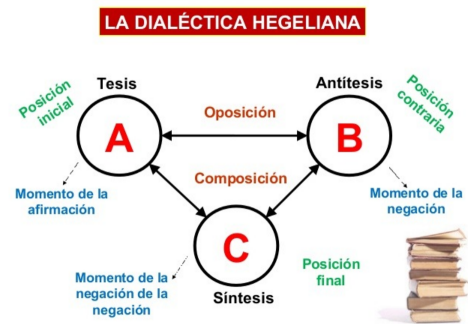


Fig. 1: Dialéctica Hegeliana

La segunda aportación significativa que EduGPT reporta directamente en el valor de nuestra ecuación, además de potenciar la base (HI/CP), como ha sido anteriormente justificado, es en la habilidad de dominar la IA (+/*). Ha sido mediante la integración a través de un conocimiento profundo de principios como la ley de Pareto (automatizar y dar el resultado del 20% de tareas que generan el 80% del valor) y la dialéctica hegeliana (síntesis creativa de extremos)[1], que han sido interiorizados desde la HI del propio autor y su dominio y conocimiento en la IA, y se han aplicado en el diseño de EduGPT con el fin de elevar el signo al que se eleva el HI de cada docente y alumno que recurra a su uso.

$$\text{EduGPT} = HI^{IA}$$

$$V1 = HI^{IA}$$

$$[V2 = HI^{\text{EduGPT}}] = [V2 = HI^{HI^{IA}}] \Rightarrow [V2 > V1].$$

1 OBJETIVOS

El proyecto EduGPT nace para responder a la necesidad de una solución educativa basada en inteligencia artificial que sea eficaz, ética y estructurada. Para ello, los objetivos se definen siguiendo el enfoque SMART[12], ampliamente reconocido en gestión de proyectos e investigación aplicada. Según esta metodología, cada objetivo debe ser específico, medible, alcanzable, relevante y limitado en el tiempo.

A partir de este marco, los objetivos se organizan en cuatro ejes de actuación: herramienta y colaboración, innovación, profesores y alumnos. Cada eje combina metas concretas para asegurar la coherencia y la viabilidad del proyecto dentro de los plazos establecidos.

1.1 Herramienta y Colaboración

EduGPT surge como respuesta directa a una necesidad real detectada en una conversación mantenida con un inversor del entorno tecnológico-educativo: ¿cómo reducir el tiempo invertido en resumir información y extraer conclusiones relevantes en el ámbito académico y profesional? A partir de esta inquietud concreta, se formuló la hipótesis de que la inteligencia artificial podía no solo acelerar estos procesos, sino también aportar valor pedagógico si se aplicaba de forma estructurada, transparente y contextualizada. Así nació EduGPT: como una herramienta orientada a optimizar la comprensión y producción académica, no sustituyendo la inteligencia humana, sino potenciándola.

Desde su concepción, el proyecto se articuló bajo una lógica doble: ser una herramienta eficaz y, al mismo tiempo, integrarse en un entorno ya funcional. Esta dualidad se resuelve a través de su implementación dentro de la plataforma FacturaGPT,[7] inicialmente orientada a la gestión documental y automatización de procesos administrativos (nóminas, facturas, contratos), proporciona un ecosistema técnico robusto, probado en entornos reales, que permite reducir drásticamente los tiempos de desarrollo, validación e implementación del módulo educativo. Aprovechar esta sinergia representa no solo una optimización de recursos, sino una muestra concreta de innovación basada en colaboración interdisciplinar[6].

El objetivo en este eje no es solo construir una herramienta académica eficiente, sino demostrar que el conocimiento pedagógico puede integrarse de forma coherente con sistemas empresariales mediante un diseño adaptativo. La arquitectura compartida permite reutilizar componentes clave (backend lógico, motor de automatización, gestión de usuarios, control de versiones) y dotar al sistema de una flexibilidad singular: los usuarios pueden acceder al módulo educativo desde la misma plataforma, utilizando interfaces familiares y beneficiándose de mejoras continuas en ambas ramas del sistema.

A nivel metodológico, esta colaboración también ha enriquecido el proceso de validación iterativa. Los flujos de uso reales registrados en FacturaGPT sirvieron como base para diseñar los primeros casos de uso de EduGPT, ajustando la lógica de prompt chaining y los criterios de evaluación. Esta retroalimentación cruzada ha sido crucial para alcanzar un producto mínimo viable funcional desde las primeras etapas, demostrando que el conocimiento generado en un ámbito puede transferirse y amplificarse en otro.

1.2 Innovación

La innovación de EduGPT reside en su capacidad para combinar dos mundos tradicionalmente separados: la automatización técnica avanzada y la pedagogía aplicada. Mientras muchas herramientas basadas en IA se centran en ofrecer funcionalidades genéricas, EduGPT ha sido diseñada desde una lógica educativa orientada a la acción significativa. No se trata simplemente de generar contenido, sino de guiar procesos cognitivos complejos —como sintetizar, argumentar o evaluar— de forma estructurada y comprensible para el usuario.

El núcleo innovador del sistema radica en su arquitectura secuencial tipo “input → insight → statement → thesis”, basada en prompt chaining adaptativo[17]. Esta estructura permite no solo automatizar tareas repetitivas, sino también interpretar la intención del usuario y construir salidas relevantes y formativas. Así, por ejemplo, un simple enunciado puede dar lugar a un documento completo, evaluado según rúbricas académicas, con retroalimentación detallada y con sugerencias de mejora pedagógica.

Además, EduGPT incorpora un enfoque human-in-the-loop [15], en el que el usuario —estudiante o docente— mantiene el control sobre el proceso. Lejos de delegar la totalidad de la producción en la máquina, se propone un modelo de co-producción inteligente, donde la herramienta actúa como catalizador del pensamiento, no como sustituto. Esta aproximación, apoyada en investigaciones recientes sobre el impacto positivo del uso guiado de IA en contextos educativos, refuerza su valor como innovación ética y aplicable.

1.3 Profesores

Uno de los colectivos clave beneficiados por EduGPT es el profesorado. En contextos educativos con alta carga docente, donde la corrección, validación y retroalimentación consumen gran parte del tiempo, EduGPT actúa como un asistente pedagógico que libera recursos sin comprometer la calidad.

Gracias a sus funcionalidades de corrección automatizada, análisis semántico y generación de informes, los docentes pueden delegar tareas repetitivas y enfocarse en la dimensión cualitativa de la enseñanza: acompañar, personalizar, motivar. La herramienta permite evaluar textos según rúbricas, detectar errores estructurales, ofrecer sugerencias de mejora y generar retroalimentación escrita en segundos. Esto no solo ahorra tiempo, sino que estandariza criterios y amplifica la equidad evaluativa.

Además, EduGPT permite al profesorado analizar patrones de aprendizaje y evolución a partir del historial de uso. Este análisis, aplicado de forma ética y orientado al desarrollo del estudiante, proporciona indicadores valiosos para la mejora continua de la práctica docente.

EduGPT no pretende reemplazar al profesor, sino reforzar su capacidad de impacto. En este sentido, se posiciona como una tecnología aliada de la pedagogía, capaz de devolver tiempo, aumentar la precisión y mejorar la experiencia docente en su conjunto.

1.4 Alumnos

En el caso del alumnado, EduGPT actúa como un multiplicador de autonomía, comprensión y rendimiento. Su valor no reside solo en producir contenido académico, sino en cómo estructura el pensamiento y entrena habilidades cognitivas clave.

Al automatizar procesos como el resumen, la organización conceptual o la evaluación formativa, EduGPT permite a los estudiantes centrarse en lo esencial: entender, argumentar, aplicar. El sistema no solo genera respuestas, sino que muestra cómo llegó a ellas, permitiendo al alumno aprender del proceso. Esta lógica transparente convierte cada interacción en una experiencia formativa que refuerza el pensamiento crítico, la atención sostenida y la capacidad de análisis.

Además, al reducir la carga mecánica de tareas como la corrección o el formateo, EduGPT libera tiempo valioso que puede ser invertido en relaciones humanas, deporte, descanso y crecimiento personal. En este sentido, la herramienta también opera como un sistema de gestión del tiempo, alineado con el Cuadrante II de Covey: dedicar energía a lo importante antes que a lo urgente.

Finalmente, EduGPT ofrece un entorno seguro para el ensayo y error. Los estudiantes pueden practicar, corregir y mejorar sin presión, promoviendo una cultura del aprendizaje continuo y del desarrollo personal autónomo.

2 FUNCIONALIDADES

Cada caso de uso responde a un tipo de input diferente y todos comparten una misma arquitectura funcional: input → insight → statements → thesis. Esta estructura permite organizar el razonamiento automatizado en pasos claros y adaptativos.

A continuación se detallan los principales casos de uso implementados, enfatizando el "qué" se automatiza, cómo se resuelve y "por qué" tiene sentido usar IA en cada uno.

2.1 Solutions

Se automatiza la redacción estructurada de un texto académico completo a partir de requisitos formales o plantillas. El sistema analiza los requerimientos a través de detección de intención y extracción de estructura (ej. objetivos, secciones esperadas). Luego, mediante generación condicional (prompt chaining), compone el texto siguiendo esa lógica. Esta utilidad, permite ahorrar tiempo en la fase más mecánica del trabajo (esqueleto base), reservando la intervención humana para afinar ideas y profundidad conceptual.

- **Input:** Enunciados, formularios o plantillas.
- **Insights:** Estructura esperada, tipo de lenguaje requerido, objetivos.
- **Statements:** Grado de cumplimiento (Sí/No/%).
- **Thesis:** Documento automatizado listo para revisión o edición.

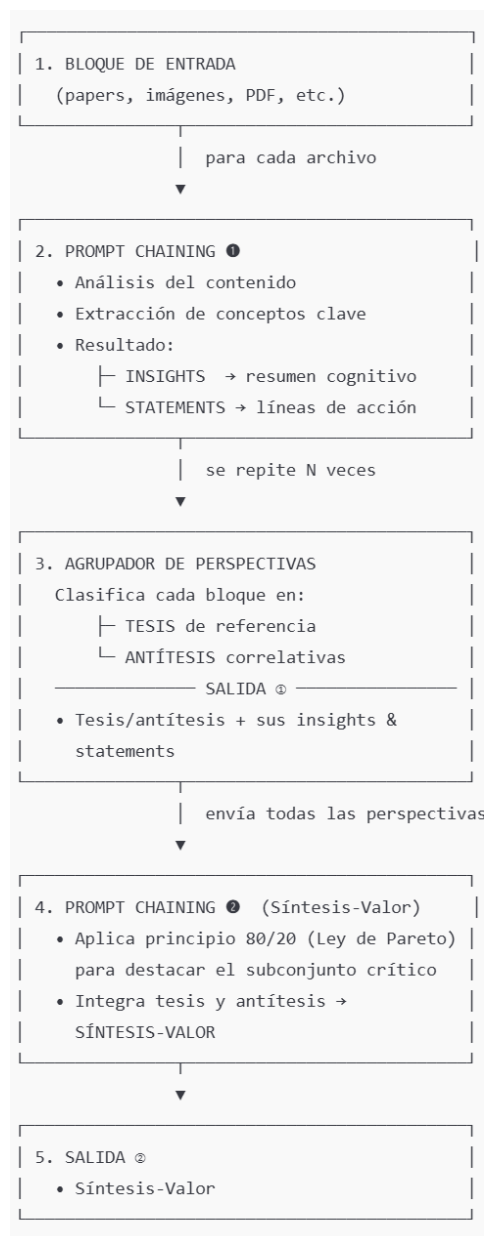


Fig. 2: Funcionamiento Compartido de los distintos servicios

2.2 Creation

Se automatiza el análisis profundo de textos complejos para facilitar el estudio. Extrayendo conceptos clave, se generan resúmenes y se analiza la accesibilidad semántica. NLP pipelines permiten organizar los fragmentos en bloques comprensibles. La finalidad de su uso es ayudar al estudiante a focalizar su atención en lo esencial y ahorrar tiempo en lecturas densas.

- **Input:** Artículos, ensayos, exámenes.
- **Insights:** Ideas clave, hipótesis, conclusiones.
- **Statements:** Claridad y nivel de comprensión (Sí/No/%).
- **Thesis:** Informe de estudio con mapas conceptuales y sugerencias.

2.3 BlackHat

Se automatiza la resolución argumentada de preguntas abiertas o exámenes sin respuestas. Para ello, se analiza el enunciado, genera una respuesta lógica, justificada y relevante, y valida su adecuación con respecto a rúbricas si están presentes. Esta herramienta permite la práctica, referencia o entrenamiento supervisado.

- **Input:** Exámenes, preguntas sueltas.
- **Insights:** Requisitos implícitos, estructura esperada.
- **Statements:** Grado de completitud (Sí/No/%).
- **Thesis:** Documento con respuestas justificadas.

2.4 GreyHat

Se automatiza la revisión técnica, formal y semántica de textos académicos dividiendo la tarea en subprocesos: corrección ortográfica, estilo, adecuación formal y revisión semántica mediante una cadena de prompts secuenciales. Esto lo ofrecemos porque reduce el tiempo de revisión manual y mejora la calidad del feedback recibido.

- **Input:** Documentos de estudiantes.
- **Insights:** Errores, calidad estructural, cohesión.
- **Statements:** Evaluación (Sí/No/%).
- **Thesis:** Informe de retroalimentación con sugerencias específicas.

2.5 WhiteHat

Se automatiza el análisis de autoría y detección de textos generados por IA aplicando modelos de detección de estilo, análisis estadístico y comparación con patrones generados por modelos como GPT. De este modo conseguimos reforzar la ética académica y ofrecer una herramienta objetiva de análisis.

- **Input:** Documentos sospechosos.
- **Insights:** Patrones estilísticos y semánticos.
- **Statements:** Sospecha de uso de IA (Sí/No/%).
- **Thesis:** Informe técnico con justificación.

2.6 DeepDesign

Deep Design es un configurador interactivo de *prompt chains* que permite al docente definir, de forma asistida, la lógica exacta con la que EduGPT generará los tres bloques fundamentales de salida: (i) la cadena única para tesis / antítesis, (ii) la cadena para los statements (líneas de acción) y (iii) la cadena para la síntesis-valor final. Mediante un chatbot contextual, el usuario introduce sus objetivos pedagógicos—extensión, tono, criterios 80/20, enfoque competencial, etc.—y recibe sugerencias automáticas que optimizan cada paso. Una vez validadas, las tres cadenas se exportan como plantilla JSON, se almacenan en la plataforma y pueden ser aplicadas por el backend sobre cualquier conjunto de documentos, garantizando coherencia y reutilización sin necesidad de reconfiguración manual en cada proyecto.

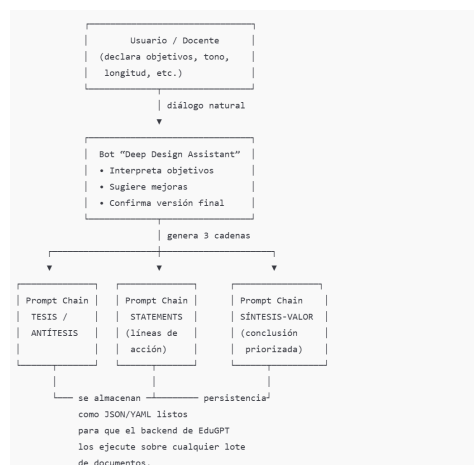


Fig. 3: Funcionamiento de DeepDesign

3 TECNOLOGÍAS

El desarrollo de EduGPT como módulo educativo dentro de la plataforma FacturaGPT se ha diseñado desde una lógica técnica orientada a la escalabilidad, modularidad y especialización por casos de uso. Las decisiones tecnológicas que componen este apartado responden a criterios no solo de viabilidad técnica, sino también de coherencia funcional, rendimiento y mantenibilidad dentro de un entorno SaaS educativo.

Para estructurar correctamente el sistema, se parte de un enfoque lógico unificado[13]: una única lógica de procesamiento que analiza cada input (un texto, enunciado o examen) y, mediante rutas condicionales, ejecuta el proceso correspondiente (corrección, resumen, validación, generación, etc.). Este modelo evita desarrollar múltiples soluciones aisladas y permite tratar todos los casos de uso desde una arquitectura común, más eficiente y fácil de escalar.

Esta lógica se organiza mediante lo que en procesamiento de lenguaje natural se conoce como NLP pipelines[14]: secuencias de pasos estructurados (por ejemplo, identificación del tipo de tarea → análisis semántico → evaluación de cumplimiento → generación o corrección). Estos pasos se implementan con una lógica modular de prompts[17], donde cada módulo tiene una función específica que puede ser afinada, reemplazada o reutilizada sin afectar al resto del sistema.

Para garantizar la independencia entre la lógica y la presentación visual, se adopta el paradigma API-first. Esto implica que todo el sistema está diseñado para que sus funcionalidades principales estén expuestas a través de una interfaz de programación (API), lo cual permite que múltiples interfaces de usuario (web, móvil, chatbot, etc.) puedan consumir los mismos servicios de manera coherente. Gracias a este enfoque, es posible mantener un único backend para todos los casos de uso, permitiendo además futuras integraciones con otros módulos de FacturaGPT.

Esta separación se refuerza con un diseño desacoplado basado en microfrontend architecture, donde cada funcionalidad visible (corregir, generar, evaluar, etc.) puede tener su propia interfaz especializada, sin depender de una estructura monolítica. Esto mejora la experiencia del usuario, facilita el mantenimiento y permite el desarrollo paralelo de nuevos casos de uso.

La arquitectura general de EduGPT se apoya en cuatro pilares tecnológicos:

- El **frontend**, centrado en ofrecer una interfaz clara, accesible y diferenciada para cada tarea académica.
- El **backend**, responsable de ejecutar las lógicas de procesamiento y coordinar los diferentes módulos de IA.
- Las **bases de datos**, encargadas de la gestión de inputs, trazabilidad, historiales y resultados generados.
- La **inteligencia artificial**, que constituye el motor de automatización mediante el uso de modelos avanzados de lenguaje como GPT.

En los siguientes subapartados se detallarán las tecnologías concretas utilizadas en cada uno de estos pilares, así como las razones técnicas y funcionales que justifican su elección.

3.1 FrontEnd

El frontend de EduGPT constituye la capa visual e interactiva de la herramienta, es decir, el entorno con el que el usuario final —ya sea docente, estudiante o evaluador— interactúa para introducir sus datos, seleccionar el tipo de tarea académica a automatizar y recibir un resultado estructurado. En el contexto educativo, esta interfaz no puede ser neutra ni genérica: debe guiar, acompañar y simplificar procesos que, de por sí, ya exigen una carga cognitiva elevada.

Desde una perspectiva técnica, el frontend se ha construido con tecnologías modernas y orientadas a la modularidad, la eficiencia y la escalabilidad. El stack seleccionado combina Next.js, React, Tailwind CSS y Shadcn/UI[3], con el objetivo de ofrecer una experiencia de usuario profesional, rápida y adaptable a múltiples casos de uso.

Next.js, un framework basado en React, permite desarrollar aplicaciones web rápidas y estructuradas, con capacidad para renderizar tanto en el servidor como en el cliente. Esta característica es especialmente útil en EduGPT cuando se gestionan respuestas generadas por inteligencia artificial que pueden requerir un renderizado progresivo o condicional. React, por su parte, estructura la interfaz en componentes independientes y reutilizables, facilitando el mantenimiento, la consistencia visual y la expansión futura del sistema.

El estilo visual se ha trabajado con Tailwind CSS, un sistema de utilidades que permite aplicar estilos de forma ágil y precisa sin necesidad de escribir CSS tradicional. Complementariamente, se ha incorporado Shadcn/UI, una biblioteca de componentes estilizados accesibles y coherentes con los estándares visuales de plataformas SaaS actuales, lo que mejora la estética y la experiencia global del usuario.

La experiencia de usuario en EduGPT se estructura bajo un enfoque task-oriented, lo que significa que se organiza en torno a acciones prácticas y reconocibles por parte del usuario —como “*resumir un artículo*”, “*evaluar una hipótesis*” o “*mejorar redacción académica*”— en lugar de funciones técnicas aisladas. Cada acción disponible en la interfaz activa automáticamente un flujo de procesamiento asociado en el backend, alineado con la arquitectura funcional del sistema (input → insight → statements → thesis).

Este diseño permite que el usuario:

- Elija directamente una tarea desde la interfaz, activando de forma explícita los módulos necesarios.
- O bien, introduzca libremente un input en lenguaje natural, lo que activa el sistema de detección de intención que determina automáticamente la ruta de procesamiento más adecuada.

Esta lógica condicional se implementa fácilmente en React mediante rutas dinámicas y componentes específicos según el tipo de input recibido. Así, cada pantalla se adapta al contexto de uso, presentando solo las opciones relevantes para el flujo de trabajo del usuario, reduciendo la complejidad visual y mejorando la eficacia de cada sesión.

El sistema se beneficia además de un enfoque API-first, en el que el frontend actúa como consumidor de un backend centralizado que alberga la lógica de automatización. Gracias a esta separación clara entre presentación y procesamiento, cada vista puede adaptarse a los diferentes tipos de usuario y tareas sin necesidad de alterar el motor principal del sistema. Esta arquitectura desacoplada es coherente con patrones modernos como la microfrontend architecture, en los que cada sección de la interfaz se comporta como un módulo independiente, capaz de evolucionar y escalar sin generar dependencia entre componentes.

3.2 BackEnd

El backend de EduGPT constituye el núcleo funcional del sistema, el espacio donde se interpreta el input del usuario, se activa la lógica de automatización adecuada y se generan las respuestas estructuradas. Mientras que el frontend actúa como interfaz visual, el backend es el sistema cognitivo que decide qué hacer, cómo y en qué orden, según el tipo de tarea académica recibida.

Concebido bajo una arquitectura modular, el backend se diseña como una estructura de microservicios lógicos. Cada uno de estos servicios resuelve un tipo específico de proceso (resumir, corregir, validar, generar, detectar uso de IA), pero todos comparten una base lógica común. La lógica interna del backend se organiza en torno a una estructura secuencial de cuatro etapas:

1. **Input:** cualquier contenido, pregunta o archivo proporcionado por el usuario.
2. **Insights:** interpretación automática mediante análisis semántico, extracción de conceptos o resúmenes.
3. **Statements:** generación de valoraciones específicas en formato binario (Sí/No), porcentual o categórico.
4. **Thesis:** integración de los análisis anteriores en un producto final estructurado, listo para ser devuelto al usuario.

Esta secuencia permite sistematizar la resolución de tareas académicas complejas bajo una lógica clara, reutilizable y pedagógicamente coherente. Se trata de un flujo adaptativo capaz de ramificarse según el tipo de input o tarea detectada.

El funcionamiento del backend responde a tres pilares esenciales. En primer lugar, se estructura mediante una arquitectura orientada a servicios, en la que cada módulo

actúa como un componente independiente, activado condicionalmente según el tipo de tarea. Esta lógica de enrutado permite que un único motor sea capaz de adaptarse dinámicamente a distintas solicitudes, manteniendo la cohesión operativa y reduciendo la duplicidad de código.

En segundo lugar, el backend se inspira en los principios fundamentales de la Máquina de Turing[5]: una lógica secuencial que resuelve problemas complejos a través de reglas condicionales aplicadas a entradas simbólicas. Esta metáfora computacional permite organizar el procesamiento académico en pasos formales y reutilizables, construyendo un sistema capaz de abordar múltiples casos de uso sin necesidad de redefinir su funcionamiento para cada tarea. En esencia, EduGPT actúa como una arquitectura universal: una misma lógica base que, mediante adaptaciones condicionales, puede resolver desde un resumen hasta una corrección argumentada o una evaluación automatizada.

En tercer lugar, se emplea un sistema de prompt chaining como motor lógico, uno de los conceptos clave que articula esta lógica backend es el prompt chaining, entendido aquí no solo como una técnica para interaccionar con modelos de lenguaje, sino como un motor lógico secuencial. En este enfoque, cada tarea se descompone en subprocesos que se ejecutan en orden según condiciones lógicas predefinidas.

Este encadenamiento de prompts actúa como una máquina de estado adaptativa: dependiendo del input inicial (un texto a corregir, una consigna para generar un artículo, una rúbrica evaluativa...), se activan diferentes ramas del flujo de procesamiento.

Las ramas del flujo de procesamiento son trayectorias condicionales que siguen los datos según la naturaleza del input recibido y el objetivo final del proceso. Estas rutas permiten que una misma arquitectura backend se bifurque dinámicamente, ejecutando diferentes tareas, módulos o pasos según se requiera.

Cómo funcionan dentro del Sistema propuesto:

1. Identificación del tipo de input.
2. Extracción o validación de objetivos. (Decisión sobre que rama del flujo se generará o usará según la conveniencia)
3. Generación o decisión de una rama de flujo de procesamiento para cumplir con los objetivos en el contenido o análisis solicitado.
4. Evaluación del cumplimiento o mejora sugerida.

El primer paso de este flujo lo constituye el módulo de intent detection, que analiza el texto introducido por el usuario para determinar cuál es el objetivo principal: ¿se desea generar un contenido?, ¿evaluarlo?, ¿resumirlo?, ¿corregirlo? automáticamente, eliminando la necesidad de que el usuario seleccione manualmente la tarea. Esto mejora la experiencia de uso, reduce errores y garantiza la coherencia entre el input y el procesamiento posterior. A continuación, se configuran como decisiones lógicas encadenadas. Por ejemplo, si el sistema detecta que el input es un trabajo académico para evaluar, activa una serie de módulos: análisis semántico → evaluación estructural → retroalimentación. En cambio, si el input es una pregunta tipo test, activa otro conjunto: análisis del enunciado → generación de respuesta → validación con fuentes.

Este enfoque, también conocido como task orchestration[1], permite dividir procesos complejos en pasos modulares que pueden adaptarse, sustituirse o afinarse sin afectar al sistema completo. Además, mejora la transparencia, ya que cada paso intermedio puede ser mostrado o auditado.

Además, esta arquitectura desacoplada permite implementar una lógica API-first, donde el backend actúa como proveedor de servicios inteligentes y el frontend como consumidor visual. Esta separación clara garantiza que EduGPT pueda integrarse en otros sistemas sin alterar su núcleo funcional, y que nuevos casos de uso se sumen al motor simplemente como nuevas rutas o combinaciones lógicas.

3.3 Bases de Datos

El diseño de un sistema de automatización académica como EduGPT exige no solo capacidad de procesamiento inteligente, sino también una infraestructura robusta de almacenamiento que garantice trazabilidad, organización y reutilización del conocimiento generado. En este contexto, las bases de datos constituyen el eje estructural que permite registrar y consultar tanto los inputs de los usuarios como los resultados producidos por los modelos de IA, junto con sus respectivas rutas de procesamiento.

En términos generales, una base de datos es un sistema que permite almacenar, estructurar y consultar información de forma sistemática. Su papel en EduGPT es múltiple: facilita la gestión de usuarios y tareas, preserva los documentos procesados, guarda el historial de interacciones y permite generar métricas analíticas sobre el uso de la plataforma. Todo ello es esencial no solo para mantener la integridad funcional del sistema, sino también para garantizar la calidad, la transparencia y la capacidad de mejora continua.

Para cubrir estos requerimientos, el sistema se apoya en un enfoque mixto que combina bases de datos relacionales (SQL) con bases no relacionales (NoSQL). Cada una responde a necesidades distintas y complementarias.

Por un lado, las bases de datos relacionales como PostgreSQL o MySQL permiten gestionar de forma estructurada:

- Las entidades del sistema (usuarios, tareas, roles, permisos).
- La trazabilidad de uso: qué input se recibió, cuándo, por quién y qué tipo de funcionalidad se activó.
- Los resultados asociados a cada sesión o documento, mediante claves relacionales.

Este enfoque ofrece robustez, integridad referencial y facilidad para realizar consultas complejas, lo que lo convierte en una solución adecuada para gestionar el núcleo administrativo y operativo de la plataforma.

Por otro lado, las bases de datos no relacionales como MongoDB o Firebase Firestore son más adecuadas para almacenar:

- Outputs generados por la IA (resúmenes, correcciones, comentarios), que no siguen una estructura fija.

- Rutas dinámicas de procesamiento lógico: por ejemplo, las cadenas de prompts aplicadas a un documento, útiles para auditoría o debugging.
- Historiales ricos y asincrónicos, como anotaciones, versiones o mejoras iterativas.

Este tipo de bases permiten mayor flexibilidad y escalabilidad, especialmente útil cuando se trabaja con información generada automáticamente o con estructuras de datos que varían en cada caso de uso.

El uso conjunto de ambas tecnologías permite al sistema mantener una base estructurada para lo estable y una base flexible para lo dinámico, lo que se traduce en un equilibrio óptimo para un entorno como EduGPT, donde conviven procesos altamente estructurados (registro de usuarios, inputs validados) con flujos dinámicos y adaptativos (resultados generados por IA).

Además, esta estrategia de almacenamiento mixto ofrece beneficios adicionales:

- **Trazabilidad educativa:** se puede reconstruir todo el proceso lógico que llevó de un input a un output, paso a paso.
- **Reutilización de contenidos:** documentos ya procesados pueden ser usados como punto de partida en futuras sesiones o con otros usuarios.
- **Análisis del sistema:** permite obtener métricas del uso por funcionalidad, perfil o nivel de cumplimiento.
- **Entrenamiento futuro:** los datos recogidos podrán alimentar mejoras, como el ajuste fino de prompts o la personalización del sistema.

3.4 Inteligencia Artificial – Corrección con GPT

La implementación de inteligencia artificial en el ámbito académico se consolida como una de las innovaciones más relevantes en la automatización educativa. En EduGPT, el modelo GPT-4 de OpenAI se utiliza como núcleo de procesamiento semántico y corrección académica, permitiendo evaluar, interpretar y mejorar textos generados por estudiantes o docentes a partir de criterios formales y conceptuales.

GPT (Generative Pre-trained Transformer) es un modelo de lenguaje de última generación entrenado sobre grandes corpus de texto, capaz de generar respuestas coherentes, argumentadas y adaptadas al contexto. Su versión más reciente, GPT-4, incorpora capacidades avanzadas de razonamiento, evaluación textual y adaptación a estilos académicos, lo que lo convierte en una herramienta idónea para el entorno educativo.

En EduGPT, GPT-4 no se emplea como un generador genérico, sino como un asistente de corrección especializado. Su función va más allá de la detección gramatical: permite analizar la cohesión de un texto, el grado de cumplimiento de una consigna o rúbrica, la calidad argumentativa, e incluso sugerir mejoras redactadas en lenguaje claro.

En este contexto, el prompt chaining se utiliza como una estrategia de razonamiento pedagógico. Consiste en encastrar instrucciones de manera progresiva para descomponer la evaluación en varias fases comprensibles para el usuario. Por ejemplo:

1. Corrección ortográfica y gramatical.
2. Evaluación de la estructura textual.
3. Análisis del cumplimiento de objetivos académicos.
4. Generación de retroalimentación detallada.

Este proceso permite ofrecer un feedback formativo, donde cada sugerencia no es solo un cambio, sino una explicación que refuerza el aprendizaje del estudiante. Además, mantiene una narrativa coherente a lo largo de la evaluación, aumentando la transparencia y la percepción de equidad.

El modelo de interacción elegido no busca sustituir al juicio humano, sino amplificarlo y enriquecerlo. Mediante un enfoque human-in-the-loop (HITL)[15], el usuario mantiene el control sobre el proceso: puede aceptar o modificar sugerencias, comparar versiones y tomar decisiones informadas a partir del análisis de la IA.

Esta dinámica presenta múltiples beneficios:

- Favorece la autonomía y el pensamiento crítico del estudiante.
- Facilita el trabajo docente sin eliminar su rol evaluador.
- Permite convertir la corrección en una oportunidad formativa, no sólo punitiva.

El valor añadido de este enfoque se respalda en investigaciones recientes que destacan cómo los modelos de lenguaje, usados con moderación e intervención humana, pueden mejorar los resultados educativos sin comprometer la calidad ni la autoría de los trabajos.

4 MÉTODO DE EVALUACIÓN

El método de evaluación tiene como finalidad validar el rendimiento funcional, pedagógico y técnico de EduGPT, asegurando que sus respuestas no solo sean coherentes y útiles, sino también escalables y replicables en distintos contextos educativos reales.

4.1 Tipos de evaluación implementados

Evaluación funcional

Verifica que el sistema responde correctamente ante distintos tipos de input utilizando pruebas predefinidas (unit tests y test cases) que cubren todas las rutas de prompt chaining. Por ejemplo, al introducir un examen de opción múltiple, se activa la ruta de análisis → respuesta → validación. La finalidad reside en asegurar que la lógica modular es sólida y que cada caso de uso está correctamente mapeado.

Evaluación semántica

La semántica es crítica en la automatización del trabajo académico; se evalúan criterios como estilo, argumentación y claridad, por ese motivo evaluamos la calidad del output en términos de lenguaje, coherencia y adecuación al objetivo utilizando rúbricas académicas adaptadas para analizar resultados generados. Por

ejemplo, un ensayo generado automáticamente es evaluado por un docente según criterios de cohesión, argumentación y estilo.

Evaluación pedagógica

Mide el valor educativo del sistema empleando encuestas de satisfacción, tests A/B y análisis de mejora en el rendimiento del usuario tras el uso de EduGPT. Ejemplo: se proporciona a un grupo un texto corregido automáticamente y a otro sin corrección; se comparan los resultados en tareas posteriores. De utiliza porque EduGPT debe ser útil para aprender, no solo para automatizar.

Evaluación de rendimiento técnico

Mide tiempos de respuesta, eficiencia de uso de recursos y robustez utilizando herramientas de monitoreo, logs de servidor, benchmarks y simulaciones de carga. Esto sirve para asegurar una experiencia fluida en producción y optimizar recursos.

4.2 Métricas clave

- Tiempo medio de generación por tarea.
- Porcentaje de cumplimiento de consigna según rúbricas.
- Tasa de aceptación de sugerencias por parte de usuarios.
- Nivel de satisfacción percibido (encuestas).
- Número de modificaciones manuales tras generación automática.

4.3 Casos de uso testados (ejemplos reales)

1. Generación de ensayos a partir de enunciados

- Input: "Redacta un ensayo sobre la ética en la inteligencia artificial."
- Output: Ensayo con estructura de introducción, desarrollo y conclusión.
- Evaluación: Rúbrica docente sobre contenido, claridad, originalidad y forma.

2. Completación automática de exámenes abiertos

- Input: Exámen vacío con enunciados.
- Output: Resultado completo del examen hecho.
- Evaluación: Comparación entre corrección automática y corrección humana.

3. Detección de uso de IA en entregas

- Input: Trabajo sospechoso de haber sido generado por IA.
- Output: Informe con indicadores semánticos y estilo.
- Evaluación: Verificación cruzada con docentes y software especializado.

4.4 Validación iterativa del MVP

La validación del MVP se estructuró en fases sucesivas que permitieron su mejora continua:

• Fase 1: Pruebas internas

- Desarrollo de casos de prueba por el equipo.
- Ajuste de prompts, rutas lógicas y outputs.
- Medición de fallos por tipo de input.

• Fase 2: Beta cerrada con usuarios reales

- Participación de docentes y estudiantes voluntarios.
- Registro de satisfacción, utilidad percibida y errores detectados.

• Fase 3: Integración con FacturaGPT + análisis estadístico

- Logs de uso en producción.
- Evaluación cuantitativa del rendimiento.
- Detección de cuellos de botella y propuestas de mejora.

Este enfoque iterativo garantiza que EduGPT evoluciona con base en datos reales y necesidades concretas, asegurando un ajuste progresivo entre tecnología, pedagogía y usabilidad.

5 DESPLIEGUE I PRESENTACIÓN

El despliegue de EduGPT se ha realizado como un módulo independiente dentro de la plataforma FacturaGPT, lo que ha facilitado su integración en un entorno SaaS ya funcional. Este despliegue se ha organizado en tres fases:

• Fase 1: Desarrollo en entorno local y pruebas internas.

• Fase 2: Beta cerrada con usuarios reales (estudiantes y docentes).

• Fase 3: Integración en entorno de producción con acceso por interfaz web.

La presentación del sistema se ha realizado mediante una sesión demostrativa en la que se mostraron los principales casos de uso, tiempos de respuesta y resultados generados en tiempo real. Esta fase permitió recoger feedback directo para afinar los últimos detalles del MVP.

6 TEORIA

6.1 Revisión bibliográfica: Estado actual de la IA aplicada a la automatización académica

El avance de la inteligencia artificial en el ámbito educativo ha crecido de manera exponencial durante la última década. Modelos como GPT-3 y GPT-4 (OpenAI) han demostrado capacidades sin precedentes para la generación de texto,

análisis semántico y evaluación automatizada. En el contexto de EduGPT, esta evolución permite transformar procesos manuales —como la corrección de exámenes, la redacción de trabajos o el análisis de lecturas académicas— en flujos automatizados, precisos y adaptativos.

Distintas publicaciones recientes respaldan la efectividad de estos modelos como asistentes educativos, especialmente cuando se combinan con la intervención humana (enfoque human-in-the-loop). El reto principal es cómo convertir modelos de lenguaje generalistas en herramientas especializadas, pedagógicas y transparentes. [fig:eum] La Fi-

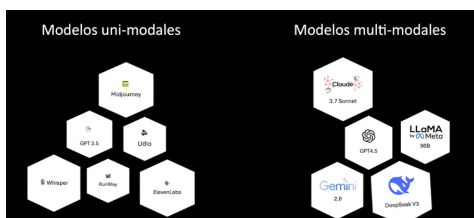


Fig. 4: Uni-Multi

gura 4 ilustra la transición del ecosistema IA desde enfoques de tarea única hacia arquitecturas de propósito general capaces de combinar información heterogénea. Modelos uni-modales ofrecen una precisión sobresaliente en dominios muy concretos, pero requieren orquestarse mediante flujos adicionales cuando la tarea demanda múltiples tipos de entrada. En contraste, los modelos multi-modales [9](Claude 3.7 Sonnet, GPT-4.5, Gemini 2.0, entre otros) integran texto, imagen e incluso audio en un mismo espacio latente, lo que simplifica la creación de *pipelines* educativos híbridos (p. ej., análisis de PDFs con diagramas, extracción de audio-transcripciones y generación de resúmenes). Para EduGPT, esta segunda categoría resulta clave: permite reducir fricción en la captura de datos (un único API) y habilita casos como corrección de exámenes escaneados o explicación de gráficas junto con su texto descriptivo—todo dentro de un flujo coherente y transparente.

6.2 Desarrollo de la herramienta: construcción del MVP

EduGPT fue concebido desde una lógica secuencial tipo Máquina de Turing[5], en la que cada input se transforma mediante reglas condicionales en un output estructurado. El MVP se basa en un sistema de flujos lógicos ramificados (prompt chaining) que activan rutas diferenciadas según el tipo de tarea académica:

Esquema funcional general:

- **Input:** archivo o texto introducido por el usuario.
- **Detección de intención (Intent detection):** identifica el tipo de tarea (corregir, resumir, generar, validar).
- **Ruta lógica adaptativa:** se activa el flujo adecuado de prompts.
- **Output:** producto estructurado con insights, validaciones y retroalimentación.

Tecnologías empleadas:

- **Backend:** arquitectura modular con servicios desacoplados.
- **Frontend:** diseño guiado por función (Task-oriented UX).
- **IA:** GPT-4 con lógica de encadenamiento de prompts.
- **Base de datos:** almacenamiento de inputs/outputs para trazabilidad y mejora continua.

Este MVP ha sido probado inicialmente dentro del entorno FacturaGPT, validando su funcionalidad en tareas de generación y corrección automatizada.

6.3 Fundamentos de la IA aplicada

Redes neuronales y aprendizaje supervisado Los modelos GPT se basan en redes neuronales profundas (deep learning), entrenadas mediante aprendizaje supervisado sobre grandes corpus de texto. Estas redes aprenden representaciones complejas del lenguaje natural y son capaces de realizar tareas como completar, resumir, corregir o clasificar texto.

De perceptrones simples a arquitecturas complejas [19]

El perceptrón, propuesto por Frank Rosenblatt en 1958, es una unidad básica de procesamiento que modela el comportamiento de una neurona artificial. Funciona tomando múltiples entradas, aplicándoles un peso, sumándolas y comparando el resultado con un umbral. Si el valor supera el umbral, se activa una función (por ejemplo, una función de activación no lineal) que determina la salida de la neurona.

Este mecanismo básico se amplía con perceptrones multicapa (MLP), donde varias capas de neuronas interconectadas permiten modelar relaciones no lineales y resolver tareas complejas. Estas redes constituyen la base del deep learning moderno.

Funciones de activación: ReLU, GELU y Sigmoid [21]

Las funciones de activación permiten a las redes neuronales capturar relaciones no lineales. Tres de las más relevantes son:

- **Sigmoid:** transforma cualquier valor en un rango entre 0 y 1. Aunque útil, sufre del problema del "desvanecimiento del gradiente"(vanishing gradient), especialmente en redes profundas, lo que ralentiza el aprendizaje.
- **ReLU (Rectified Linear Unit):** activa solo valores positivos ($f(x) = \max(0, x)$). Mejora el aprendizaje al mantener gradientes constantes para valores positivos, pero puede producir neuronas muertas si muchas salidas se anulan permanentemente.
- **GELU (Gaussian Error Linear Unit):** combina las ventajas de ReLU con una activación más suave y probabilística. Favorece una mejor propagación del gradiente y ha demostrado resultados superiores en modelos como GPT.

La elección de GELU en arquitecturas como GPT se justifica por su capacidad de aprendizaje eficiente en redes profundas, minimizando la pérdida de información y manteniendo activas más neuronas relevantes durante el entrenamiento.

Algoritmos clásicos de ML: un contexto comparativo

Aunque EduGPT se basa en deep learning, es relevante conocer algoritmos clásicos como:

- Regresión lineal/logística
- K-Nearest Neighbors (K-NN)
- Árboles de decisión y Random Forest
- Support Vector Machines (SVM)
- Gradient Boosting Machines (GBM)

Estos algoritmos han sido ampliamente usados en educación para predicción de resultados, análisis de aprendizaje y recomendación de contenidos, aunque no poseen la capacidad generativa ni la comprensión contextual profunda de modelos como GPT.

Redes Convolucionales [20] Las redes convolucionales (CNN) tienen un papel relevante en tareas de visión por computadora. Su inclusión sería pertinente únicamente si se aborda el reconocimiento automatizado de exámenes escaneados o imágenes, en cuyo caso podrían complementar a GPT en la parte de extracción de texto o interpretación visual. Si no se desarrolla este tipo de funcionalidad, se recomienda omitirlas.

CONCLUSIONES

6.4 Análisis de ventajas y desventajas

EduGPT aporta beneficios cuantificables en entornos educativos: acelera la generación y corrección de textos, libera tiempo docente para actividades de alto valor añadido, proporciona retroalimentación explícita alineada con rúbricas y mantiene un registro completo del flujo de procesamiento, lo que facilita la trazabilidad y la investigación pedagógica. Sin embargo, el sistema presenta limitaciones inherentes a toda aplicación basada en modelos de lenguaje de gran escala. Persiste la dependencia crítica de infraestructura computacional y conectividad; los sesgos presentes en los datos de entrenamiento pueden trasladarse a las evaluaciones; la detección de intención es susceptible de error, y la delegación excesiva corre el riesgo de reducir la autonomía y el pensamiento crítico del estudiante. Estas desventajas exigen estrategias de mitigación continuas, desde mecanismos de verificación humana hasta auditorías periódicas del corpus y de los resultados.

6.5 Posibles Mejoras en el futuro

Las líneas de desarrollo prioritarias se centran, en primer lugar, en la personalización semántica del feedback mediante modelos centrados en el perfil cognitivo del usuario, lo que permitiría ajustar la profundidad e intencionalidad de las sugerencias y evitar la uniformidad advertida por Huxley en

Un mundo feliz (1932)[11]. En segundo lugar, la ampliación multimodal, integrando texto, audio e imagen, facilitaría la evaluación de presentaciones y diagramas, mientras que la mejora de los clasificadores de intención mediante *few-shot learning* reduciría las rutas lógicas incorrectas. Finalmente, la publicación de interfaces abiertas y auditables contribuiría a la transparencia y evitaría la concentración del conocimiento en “cámaras selladas” que recuerdan al control informativo descrito por Bradbury en *Fahrenheit 451* (1953)[2].

6.6 Riesgos éticos y limitaciones

Los riesgos principales se agrupan en cuatro dimensiones. La primera es la privacidad: el almacenamiento de históricos detallados exige políticas estrictas de anonimización y consentimiento informado para impedir usos secundarios no deseados. La segunda es la posible homogeneización del discurso, dado que la optimización por similitud puede erosionar la diversidad de perspectivas, reproduciendo el conformismo social señalado por Huxley. La tercera se refiere a la gobernanza: si los criterios de evaluación quedan bajo el control de un número reducido de actores, se corre el riesgo de un desequilibrio de poder análogo al que Orwell satiriza en *Rebelión en la granja* (1945)[16]. La cuarta dimensión es la sostenibilidad: la huella energética de la inferencia masiva debe evaluarse frente al beneficio pedagógico, promoviendo infraestructuras eficientes y compensaciones ambientales. Reconocer y atajar estos riesgos es condición necesaria para que EduGPT consolide su utilidad académica sin comprometer la autonomía intelectual ni los valores éticos fundamentales del proceso educativo.

AGRADECIMIENTOS

Me agradezco profundamente a mí mismo. Por todas esas veces en que me vencí. Por el crecimiento personal, intelectual y humano realizado durante estos tres años de carrera.

A mi padre, por su dedicación incansable a cultivar en mí el hábito de la lectura, el pensamiento reflexivo y la curiosidad. A mi madre, por su amor incondicional y su lucha constante: sin su ejemplo de persistencia y resiliencia este trabajo no existiría. A mis hermanas, que son las personas que más admiro en este planeta por su capacidad de luchar por lo que quieren. A mi tío y padrino David cuya presencia y escucha años atrás fueron vitales para todo este progreso. Y a toda mi espectacular familia por sus inabarcables virtudes que han sido depositadas como semillas en el fértil campo de mi corazón: Carme, Avi Galo, Abuelita Carmen de Tous, Abuelita Carmen de Santcugat, Abuelito Miguel, Tío Edu, Marina, Anna, Jan, Tía Bea, Daniela, Mireia, Clara Elena, Marta, Meri, Albert, Jana, Berta, Tío Jesús y Abuela Lupe.

A mis amigos de la universidad —Alejandro González, Alex Ruiz, Álex Navajas, Joel Canales y Xavi Fernández— por estar, por sostenerme y por regalarme tiempo para crecer más allá del ámbito académico.

A mi tutor de TFG, por su paciencia infinita, su buena disposición constante y por ese “buenos días!incluso cuando las entregas llegaban tarde.

A Núria, mi terapeuta, por escucharme, por acompañarme y ayudarme en muchos momentos. A los amigos,

colegas y conocidos que durante estos años me han regalado alegría, conversación o simplemente momentos valiosos que me ayudaron a llegar hasta aquí.

Mención especial a Daniel Tapia, Joan Manel García, Abril Bertrán, Carlos Gómez y Gemma Riba. Gracias por ser parte del camino.

REFERÈNCIES

- [1] ANDREESSEN HOROWITZ. Emerging architectures for llm applications, 2023.
- [2] BRADBURY, R. *Fahrenheit 451*. Debolsillo, 2021. Traducción al español de Alfredo Crespo.
- [3] CODENOVA, M. Front-end architecture with next.js, react and tailwind css. <https://medium.com/@codenova/system-design-template-front-end-architecture-with-next-js-react-and-tailwind-css-0b364f0b1fe9>, 2024. Accedido el 21 jun 2025.
- [4] COVEY, S. R. *The 7 Habits of Highly Effective People: Powerful Lessons in Personal Change*. Free Press, 1989.
- [5] DE MOL, L. Turing machines. In *The Stanford Encyclopedia of Philosophy*, E. N. Zalta and U. Nodelman, Eds., summer 2025 ed. Metaphysics Research Lab, Stanford University, 2025.
- [6] DRAGOMIRESCU, O.-A., CRĂCIUN, P.-C., AND BOLOGA, A.-R. Enhancing invoice processing automation through the integration of devops methodologies and machine learning. *Systems* 13, 2 (2025), 87.
- [7] FACTURAGPT. Facturagpt: Automatiza la facturación de tus gastos con ia, 2025. Plataforma de facturación automática basada en inteligencia artificial.
- [8] GOLEMAN, D. *Focus: The Hidden Driver of Excellence*. HarperCollins, 2013.
- [9] GROWIT ACADEMY. Curso online: Chatgpt + ia. <https://plataforma.growitschool.com/>, 2024. Accedido el 19 junio 2025.
- [10] HEGEL, G. W. F. *Fenomenología del Espíritu*. Fondo de Cultura Económica, 1807.
- [11] HUXLEY, A. *Un mundo feliz*. Debolsillo, 2017. Traducción al español de Ramón Huerta.
- [12] KOHTAMÄKI, M., RABETINO, R., PARIDA, V., SJÖDIN, D., AND HENNEBERG, S. Managing digital servitization toward smart solutions: Framing the connections between technologies, business models, and ecosystems. *Industrial Marketing Management* 105 (2022), 253–267.
- [13] LAMSAL, R., RODRIGUEZ READ, M., AND KARUNASEKERA, S. Langformers: Unified nlp pipelines for language models. *arXiv preprint* (2025).
- [14] MAGHSOUDI, A., SADA, Y. H., NOWAKOWSKI, S., GUFFEY, D., ZHU, H., YARLAGADDA, S. R., LI, A., AND RAZJOUYAN, J. A multi-institutional natural language processing pipeline to extract performance status from electronic health records. *Cancer Control* 31 (2024), 10732748241279518.
- [15] MEMARIAN, B., AND DOLECK, T. Human-in-the-loop in artificial intelligence in education: A review and entity-relationship analysis. *Computers in Human Behavior: Artificial Humans* 2 (2024), 100053.
- [16] ORWELL, G. *Rebelión en la granja*. Debolsillo, 2015. Traducción al español de Marcial Souto.
- [17] SEO, S., YOO, S., AND JANG, Y. A prompt chaining framework for long-term recall in llm-powered intelligent assistant. In *Proceedings of the 30th International Conference on Intelligent User Interfaces (IUI '25)* (2025), Association for Computing Machinery, pp. 89–105.
- [18] SHARMA, R. S. *El Monje que Vendió su Ferrari*. HarperCollins, 1997.
- [19] TECH, R. Cómo funcionan las redes neuronales – inteligencia artificial [video]. Tech. rep., YouTube, 2020.
- [20] TECH, R. Redes neuronales convolucionales – clasificación avanzada de imágenes con ia/ml (cnn) [video]. Tech. rep., YouTube, 2021.
- [21] TECH, R. Funciones de activación a detalle (redes neuronales) [video]. Tech. rep., YouTube, 2022.