UNIVERSIDAD DE CASTILLA-LA MANCHA

GRADO EN INGENIERÍA INFORMÁTICA

Intensificación cursada: Computación

TRABAJO FIN DE GRADO

Servicio web para la generación automática de contenidos en formato de Lectura Educativa Inclusiva (LEdui)

Ángel Luis González Serrano

Escuela Superior de Informática



Ciudad Real, febrero de 2026



UNIVERSIDAD DE CASTILLA-LA MANCHA
ESCUELA SUPERIOR DE INFORMÁTICA

TRABAJO FIN DE GRADO

Servicio web para la generación automática de contenidos en formato de Lectura Educativa Inclusiva (LEdui)

Autor: Ángel Luis González Serrano

Tutor: Carlos González Morcillo

febrero, 2026

Servicio web para la generación automática de contenidos en formato de Lectura Educativa Inclusiva (LEdui)
© Ángel Luis González Serrano, 2026

Este documento se distribuye con licencia CC BY-NC-SA 4.0. El texto completo de la licencia se puede obtener en https://creativecommons.org/licenses/by-nc-sa/4.0/. La copia y distribución de esta obra está permitida en todo el mundo, sin regalías y por cualquier medio, siempre que esta nota sea preservada. Se concede permiso para copiar y distribuir traducciones de este libro desde el español original a otro idioma, siempre que la traducción sea aprobada por el autor del libro y tanto el aviso de copyright como esta nota de permiso, sean preservados en todas las copias.



Este texto ha sido preparado con la plantilla LaTeX para Trabajo Fin de Estudios en Ingeniería Informática para la UCLM publicada por Jesús Salido en el repositorio público Zenodo, DOI: 10.5281/zenodo.4561708.

A Ali, Hernán y Amelia,
motivación para intentar cada día construir un mundo mejor

TRABAJO FIN DE GRADO - ESCUELA SUPERIOR DE INFORMÁTICA (UCLM)

Servicio web para la generación automática de contenidos en formato de Lectura Educativa Inclusiva (LEdui)

Ángel Luis González Serrano
Ciudad Real, febrero 2026

Resumen

En el ámbito educativo actual, la inclusión del alumnado con necesidades educativas especiales continúa siendo un reto significativo. A pesar de la existencia de numerosos estándares y marcos normativos orientados a la accesibilidad y a la educación inclusiva, su aplicación práctica en la elaboración de tareas y enunciados educativos resulta compleja para gran parte del profesorado. Esta dificultad puede generar barreras cognitivas que impiden al alumnado demostrar adecuadamente los conocimientos adquiridos. El presente Trabajo Fin de Grado aborda esta problemática con el objetivo de facilitar la creación de contenidos educativos más claros, accesibles e inclusivos mediante el uso de la tecnología. Para ello, se ha diseñado el formato **Lectura Educativa Inclusiva (LEdui)**, que sintetiza principios inclusivos procedentes del Diseño Universal para el Aprendizaje (DUA), la Lectura Fácil, estándares WCAG y la norma UNE 71362:2020, entre otros. A partir de este formato, se ha desarrollado un servicio web REST con FastAPI, PostgreSQL y autenticación OAuth 2.0 (Google) que integra modelos de IA generativa (Claude 3.7 Sonnet, Google Gemini) para la transformación automática de enunciados educativos. El sistema integra técnicas de Inteligencia Artificial de forma controlada y determinista, generando siempre una estructura fija que incluye objetivos de aprendizaje, pasos guiados, opciones de entrega multimodales, apoyos visuales, preguntas de reflexión y estimaciones temporales. Además, se incorporan funcionalidades complementarias como la generación de explicaciones pedagógicas para el docente y la creación de imágenes de contexto y pictogramas educativos. El formato y el sistema han sido validados mediante un panel de 5 expertos (3 docentes de Formación Profesional, 1 profesional de editorial educativa especializada en accesibilidad y 1 orientador educativo), quienes han evaluado 15 enunciados representativos de distintas áreas de conocimiento y niveles educativos. Los resultados obtenidos muestran que el sistema desarrollado permite reducir barreras cognitivas y mejorar la comprensión de las tareas educativas, actuando como una herramienta de apoyo al profesorado. LEdui se presenta como un formato de aplicabilidad universal, beneficioso tanto para alumnado con necesidades educativas especiales como para el conjunto del alumnado. El trabajo pone de manifiesto el potencial de los sistemas inteligentes como apoyo a la inclusión educativa, así como la necesidad de su uso responsable y alineado con principios pedagógicos y normativos.

Palabras clave: educación inclusiva, accesibilidad cognitiva, Diseño Universal para el Aprendizaje, inteligencia artificial, tecnología educativa .

BACHELOR DISSERTATION - ESCUELA SUPERIOR DE INFORMÁTICA (UCLM)

Web service for the automatic generation of content in Inclusive Educational Reading (LEdui) format

Ángel Luis González Serrano
Ciudad Real, febrero 2026

Abstract

In the current educational context, the inclusion of students with special educational needs remains a significant challenge. Despite the existence of numerous standards and regulatory frameworks focused on accessibility and inclusive education, their practical application in the design of educational tasks and assignments is often complex for a large proportion of teachers. This difficulty can lead to cognitive barriers that prevent students from adequately demonstrating the knowledge they have acquired. This Bachelor's Thesis addresses this issue with the aim of facilitating the creation of clearer, more accessible and inclusive educational content through the use of technology. To this end, the **Inclusive Educational Reading (LEdui)** format has been designed, synthesising inclusive principles derived from Universal Design for Learning (UDL), Easy-to-Read guidelines, WCAG standards and the UNE 71362:2020 standard, among others. Based on this format, a REST web service has been developed using FastAPI, PostgreSQL and OAuth 2.0 authentication (Google) that integrates generative AI models (Claude 3.7 Sonnet, Google Gemini) for the automatic transformation of educational statements. The system integrates Artificial Intelligence techniques in a controlled and deterministic manner, always generating a fixed structure that includes learning objectives, guided steps, multimodal submission options, visual supports, reflection questions and time estimates. In addition, complementary functionalities are provided, such as the generation of pedagogical explanations for teachers and the creation of contextual images and educational pictograms. The format and system have been validated by a panel of 5 experts (3 Vocational Training teachers, 1 educational publishing professional specialized in accessibility, and 1 educational counselor), who evaluated 15 representative statements from different knowledge areas and educational levels. The results obtained show that the developed system helps to reduce cognitive barriers and improve the understanding of educational tasks, acting as a support tool for teachers. LEdui is presented as a universally applicable format, beneficial both for students with special educational needs and for the student body as a whole. This work highlights the potential of intelligent systems to support educational inclusion, as well as the need for their responsible use aligned with pedagogical and regulatory principles.

Keywords: inclusive education, cognitive accessibility, Universal Design for Learning, artificial intelligence, educational technology .

Agradecimientos

Deseo expresar mi más sincero agradecimiento a mi tutor, **Carlos González Morcillo**. Sin su motivación constante, su ánimo y la cercanía con la que ha acompañado cada etapa del proceso, este Trabajo Fin de Grado difícilmente habría llegado a ver la luz. Su manera de guiar y facilitar cada paso ha sido clave para convertir esta idea en realidad y para cerrar, al fin, una espina que me acompañaba desde hace más de una década.

Mi gratitud se extiende, de manera muy especial, a mi familia, mi refugio permanente: el lugar que me sostiene y alimenta esa dulce locura que consiste en pensar y, mejor aún, en luchar por construir un mundo mejor. Mención especial merece mi mujer, **Alicia**. Estas páginas son también testimonio de cómo los momentos difíciles, tras un largo recorrido, pueden transformarse en aprendizajes valiosos y en una aportación positiva para los demás.

Y, por supuesto, a mis alumnos. A todos, sin excepción, y muy especialmente a esos pequeños superhéroes que cada curso llegan con un informe de necesidades educativas especiales bajo el brazo. Gracias por recordarme cada día el verdadero sentido de haber dedicado mi vida a la docencia: mirar el mundo desde nuevas perspectivas y ofrecer caminos de aprendizaje donde no importen las dificultades, las etiquetas ni las circunstancias.

Ángel Luis González Serrano
Ciudad Real, 2026

Notación y acrónimos

LISTA DE ACRÓNIMOS

API	:	Application Programming Interface
CAST	:	Center for Applied Special Technology
CSS	:	Cascading Style Sheets
DUA	:	Diseño Universal de Aprendizaje
ECTS	:	European Credit Transfer and Accumulation System
HTML	:	HyperText Markup Language
IA	:	Inteligencia Artificial
IDE	:	Integrated Development Environment
IFLA	:	International Federation of Library Associations and Institutions
LEdui	:	Lectura Educativa Inclusiva
MVP	:	Producto Mínimo Viable
OOD	:	Object-Oriented Design
PhD	:	Philosophiae Doctor
RAD	:	Rapid Application Development
SAAC	:	Sistemas Aumentativos y Alternativos de Comunicación
SSADM	:	Structured Systems Analysis & Design Method
TFG	:	Trabajo Fin de Grado
UDL	:	Universal Design for Learning
UNE	:	Asociación Española de Normalización

Índice general

Resumen vii

Abstract ix

Agradecimientos xi

Notación y acrónimos xiii

Índice de figuras xvii

Índice de tablas xix

1 Introducción 1
- 1.1 Motivación . . . 1
- 1.2 Problemática concreta a resolver . . . 2
- 1.3 Contexto disciplinar y tecnológico . . . 2
- 1.4 Contribuciones principales del trabajo . . . 7
- 1.5 Estructura del documento . . . 8

2 Objetivo 9
- 2.1 Objetivo general . . . 9
- 2.2 Objetivos complementarios . . . 9

3 Plan de gestión del trabajo 11
- 3.1 Metodología de trabajo . . . 11
- 3.2 Fases de desarrollo y entregables . . . 12
- 3.3 Cronograma y dedicación temporal . . . 14
- 3.4 Recursos . . . 15
- 3.5 Análisis de riesgos y plan de contingencia . . . 19

4 Desarrollo y resultados 21
- 4.1 Definición y validación del estándar LEdui . . . 21
- 4.2 Definición de requisitos del sistema inteligente. . . . 29
- 4.3 Arquitectura y diseño técnico del sistema . . . 30
- 4.4 Flujo de procesamiento de una transformación completa . . . 38
- 4.5 Implementación del *backend* . . . 42
- 4.6 Implementación del *frontend* . . . 50
- 4.7 Despliegue del sistema . . . 57
- 4.8 Plan de pruebas y validación . . . 64

5 Conclusiones **69**
5.1 Objetivos alcanzados . . . 69
5.2 Justificación de competencias adquiridas . . . 73
5.3 Trabajos derivados y futuros . . . 75
5.4 Valoración personal . . . 76

Bibliografía **77**

A Anexo A. Descripción detallada de las reglas LEdui **83**
A.1 Regla 1. Claridad lingüística . . . 83
A.2 Regla 2. Estructura guiada . . . 84
A.3 Regla 3. Coherencia expositiva . . . 85
A.4 Regla 4. Tono y motivación . . . 86
A.5 Regla 5. Aprendizaje autónomo . . . 87
A.6 Regla 6. Intención pedagógica . . . 88
A.7 Regla 7. Aprendizaje colaborativo . . . 89
A.8 Regla 8. Pensamiento crítico y reflexión . . . 90
A.9 Regla 9. Conexión profesional y contextual . . . 91
A.10 Regla 10. Recursos multimedia y multimodales . . . 92

B Anexo B. Enunciados utilizados en la validación pedagógica **95**
B.1 Educación Secundaria Obligatoria . . . 96
B.2 Formación Profesional . . . 97
B.3 Universidad . . . 101

C Anexo C. Intrumentos de validación pedagógica **105**
C.1 Instrucciones para evaluadores . . . 105
C.2 Rúbrica de evaluación de enunciados transformados . . . 106
C.3 Cuestionario cualitativo . . . 108

D Anexo D. Documentación de la API REST **111**
D.1 Introducción . . . 111
D.2 Autenticación y gestión de usuarios . . . 112
D.3 Transformación de contenidos educativos . . . 114
D.4 Gestión del historial del usuario . . . 119
D.5 Panel de administración . . . 122
D.6 Modelos de datos (Pydantic schemas) . . . 124
D.7 Ejemplos de uso completos . . . 125
D.8 Manejo de errores y códigos de estado . . . 129
D.9 Consideraciones técnicas adicionales . . . 130
D.10 Resumen de endpoints . . . 131

Índice de figuras

4.1 Arquitectura general del sistema LEdui 31
4.2 Estructura de módulos del *backend* 33
4.3 Diagrama entidad-relación de la base de datos 35
4.4 Diagrama de secuencia: Autenticación y transformación de contenido (Etapas 1-3) 38
4.5 Diagrama de secuencia: Generación de recursos visuales y respuesta (Etapas 4-6) 40
4.6 Pantalla de inicio de sesión 50
4.7 Pantalla para realizar transformación 51
4.8 Pantalla mostrando enunciado adaptado 52
4.9 Pantalla mostrando explicación pedagógicas 52
4.10 Pantalla mostrando pictogramas 53
4.11 Pantalla mostrando imágenes de contexto 53
4.12 Pantalla con el historial de un usuario 54
4.13 Pantalla mostrando transformación concreta y botón descargar 55
4.14 Pantalla de administración: usuarios 56
4.15 Pantalla de administración: transformaciones 56

Índice de tablas

3.1 Distribución de horas por fase del proyecto 14
3.2 Cronograma temporal del proyecto (Sep 2025 - Ene 2026) 14
3.3 Coste económico total del proyecto 18

4.1 Conexión de las reglas LEdui con dimensiones operativas 24
4.2 Conexión reglas LEdui y necesidades educativas especiales 27
4.3 Endpoint de adaptación de contenidos a LEdui 42
4.4 Comparativa técnica de modelos de lenguaje evaluados. 44
4.5 Endpoints de generación de imágenes de contexto y pictogramas 46
4.6 Endpoint de autenticación y gestión de usuario 48
4.7 Endpoint de gestión del historial y persistencia 48
4.8 Endpoints de administración 49
4.9 Variables de entorno obligatorias del sistema 58
4.10 Variables de entorno opcionales 58

5.1 Resumen de estimaciones de desarrollo de trabajos futuros 75

D.1 Códigos de estado HTTP utilizados por la API 112
D.2 Campos de respuesta del endpoint de estadísticas 114
D.3 Códigos de estado HTTP y situaciones de error 129
D.4 Endpoints de autenticación y gestión de usuarios 131
D.5 Endpoints de transformación de contenidos 132
D.6 Endpoints de gestión del historial del usuario 132
D.7 Endpoints del panel de administración 132

CAPÍTULO 1
Introducción

El presente Trabajo Fin de Grado aborda la problemática de la comprensión de contenidos educativos desde una perspectiva inclusiva, integrando fundamentos pedagógicos con soluciones tecnológicas.

En este primer capítulo se expone la motivación que ha impulsado su desarrollo y se delimita la problemática específica que se pretende abordar. Asimismo, se presenta el contexto disciplinar y tecnológico necesario para situar el trabajo dentro del marco académico en el que se inscribe.

Como cierre del capítulo, se detallan las principales contribuciones de este trabajo y se ofrece una breve descripción de la estructura del resto de capítulos que conforman el documento.

1.1. MOTIVACIÓN

La inclusión y la accesibilidad educativa representan uno de los principales desafíos de los sistemas educativos actuales, especialmente ante la creciente diversidad del alumnado. Sin enfoques inclusivos, muchos estudiantes no pueden demostrar su verdadero potencial, no por falta de capacidad, sino por las barreras cognitivas, lingüísticas u organizativas presentes en la forma en que se presentan los contenidos.

Esta realidad se manifiesta de manera habitual en la práctica docente. Como profesor de Formación Profesional en la familia de Informática y Comunicaciones, he observado cómo numerosos alumnos se bloquean ante enunciados poco claros, excesivamente abstractos o mal estructurados. En muchos casos, la dificultad no reside en el contenido en sí, sino en la ausencia de una guía inicial que les permita comprender qué se espera de ellos y cómo abordar la tarea.

A esta experiencia profesional se suma una vivencia personal especialmente significativa. Como padre de un niño con necesidades educativas especiales, he podido comprobar que muchas de las dificultades no están en la capacidad para realizar una actividad, sino en la comprensión de sus instrucciones. Cuando estas se reformulan de manera más clara, estructurada y guiada, el rendimiento mejora de forma notable. Esta experiencia refuerza una convicción: sin un enfoque inclusivo, muchos alumnos no llegan a mostrar realmente de lo que son capaces.

Desde esta doble perspectiva, docente y personal, surge una pregunta recurrente: ¿cómo trasladar de forma efectiva los principios de inclusión al diseño real de materiales educativos? Aunque existen estándares y marcos orientados a la inclusión, en la práctica diaria suelen percibirse como orientaciones generales difíciles de aplicar de manera sistemática. La falta

de herramientas que los conviertan en acciones concretas hace que, con frecuencia, su uso quede relegado a la intención más que a la práctica.

Este Trabajo Fin de Grado nace, por tanto, de **la motivación de hacer operativos dichos estándares**, explorando cómo la computación puede ayudar a transformar principios inclusivos en apoyos reales para el profesorado. El objetivo es facilitar la creación de contenidos más claros, estructurados y accesibles, de forma que el alumnado pueda centrarse en aprender y demostrar sus capacidades, sin que ello suponga un aumento significativo de la carga de trabajo docente.

1.2. PROBLEMÁTICA CONCRETA A RESOLVER

En la práctica educativa diaria, una de las principales barreras para el alumnado aparece en el momento inicial de una tarea o actividad propuesta por el docente. Con frecuencia, el problema no reside en la dificultad de la actividad, sino en la comprensión del enunciado que define qué se debe hacer y cómo abordarlo.

Enunciados poco claros, excesivamente abstractos o carentes de una estructura que guíe los pasos a seguir generan inseguridad y bloqueos en las fases iniciales de la tarea. Esta falta de comprensión impide que muchos estudiantes lleguen siquiera a intentarla, lo que limita su capacidad para demostrar los conocimientos y competencias que realmente poseen. En todos estos casos, el enunciado actúa como una barrera cognitiva que interfiere entre el conocimiento del alumno y su capacidad para expresarlo.

Este Trabajo Fin de Grado aborda directamente esta problemática, centrando su atención en la mejora de la comprensión de los enunciados educativos. Para ello, se propone una solución computacional orientada a transformar los enunciados en instrucciones más claras, estructuradas y accesibles, con el objetivo de reducir la inseguridad inicial, evitar bloqueos y permitir que el alumnado pueda demostrar de manera efectiva lo que sabe y es capaz de hacer.

1.3. CONTEXTO DISCIPLINAR Y TECNOLÓGICO

El análisis del contexto disciplinar y tecnológico en el que se enmarca este Trabajo Fin de Grado resulta fundamental para comprender la naturaleza del problema abordado, así como la solución propuesta planteada.

Esta sección aborda de manera integrada los principales fundamentos disciplinares y tecnológicos que sustentan este trabajo. En primer lugar, se revisan los marcos normativos de inclusión educativa que establecen los criterios teóricos y pedagógicos de referencia. A continuación, se analiza el papel de los apoyos visuales y los recursos multimedia como estrategias consolidadas para facilitar la comprensión y la accesibilidad cognitiva. Finalmente, se examina el potencial de los sistemas inteligentes aplicados a la adaptación de contenidos educativos como vía para superar las limitaciones de los enfoques manuales y avanzar hacia soluciones tecnológicas alineadas con los principios de la educación inclusiva.

1.3.1. Marcos normativos de inclusión educativa

El desarrollo de soluciones tecnológicas orientadas a facilitar prácticas educativas inclusivas requiere, como punto de partida, un conocimiento riguroso de las normativas, estándares y marcos de referencia que actualmente se aplican, de forma principalmente manual, en los centros educativos. Estos marcos constituyen el resultado de décadas de investigación peda-

gógica y normativa, y proporcionan criterios consolidados para la adaptación de materiales y actividades educativas a contextos de diversidad.

Uno de los enfoques más relevantes en este ámbito es el **Diseño Universal para el Aprendizaje (DUA)**, desarrollado por el Center for Applied Special Technology (CAST) [5]. El DUA propone diseñar los procesos de enseñanza desde su origen para atender a la diversidad del alumnado, evitando la necesidad de adaptaciones posteriores. Este marco se articula en torno a tres principios fundamentales: la provisión de múltiples formas de representación de la información, múltiples formas de acción y expresión, y múltiples formas de implicación. Numerosos estudios destacan que la aplicación de estos principios favorece la participación activa del alumnado y reduce barreras en el acceso al aprendizaje, especialmente en contextos inclusivos.

Otro marco de referencia ampliamente utilizado es el de la **Lectura Fácil**. La Norma UNE 153101:2018 EX [1] proporciona pautas y recomendaciones para la adaptación y creación de documentos en Lectura Fácil, estableciendo criterios de redacción, estructura y diseño que facilitan la comprensión de la información escrita para personas con dificultades de lectura. Esta norma, articulada operativamente por organizaciones como Plena Inclusión [21], centra su aplicación en la adaptación de textos mediante el uso de un lenguaje claro, frases cortas, estructuras sintácticas sencillas y una organización visual coherente. Aunque la Lectura Fácil ha sido aplicada tradicionalmente a textos informativos, administrativos y legales, organizaciones especializadas como Plena Inclusión [21] destacan su utilidad en el ámbito educativo, donde la comprensión de enunciados e instrucciones condiciona de forma directa la participación del alumnado en las actividades de aprendizaje, aspecto clave de este trabajo.

Desde un enfoque de **calidad de los materiales educativos digitales**, la norma UNE 71362:2020 [2] presenta como enfoque novedoso la accesibilidad como indicador de calidad. Esta norma establece un conjunto de criterios que abarcan aspectos técnicos, pedagógicos y de accesibilidad, prestando especial atención a la inclusión y a la adaptabilidad de los recursos digitales. 2 de sus 15 criterios son específicamente relativos a la accesibilidad. Los mismos, entre sus recomendaciones se encuentran la claridad en la presentación de la información, la adecuación del lenguaje al público destinatario, la estructuración coherente de los contenidos y la consideración de diferentes necesidades del alumnado, lo que la convierte en una referencia clave para el diseño de materiales educativos digitales inclusivos.

A nivel internacional, la accesibilidad digital se encuentra regulada por las **Pautas de Accesibilidad para el Contenido Web (WCAG)** [33], desarrolladas por el World Wide Web Consortium (W3C). Aunque estas pautas están orientadas principalmente al diseño de contenidos web accesibles, sus principios fundamentales —perceptible, operable, comprensible y robusto— son plenamente aplicables al ámbito educativo digital. En particular, el principio de comprensibilidad enfatiza la necesidad de ofrecer instrucciones claras, coherentes y predecibles, reduciendo la carga cognitiva asociada a la interacción con los contenidos [32].

Complementario al estudio de estos estándares de referencia en accesibilidad, resulta especialmente relevante para este trabajo abordar este ámbito **desde un enfoque específicamente cognitivo** más propio del término inclusión que accesibilidad. Mientras que la accesibilidad general se centra principalmente en eliminar barreras físicas, sensoriales o técnicas (como la navegación por teclado, el contraste visual o los textos alternativos), la accesibilidad cognitiva pone el foco en **cómo las personas procesan, comprenden y retienen la información**. Es decir, no se trata únicamente de que el contenido sea técnicamente

accesible desde un punto de vista operacional, sino de que sea realmente comprensible, y por extensión útil, para personas con diferentes capacidades cognitivas.

En este ámbito, diversas instituciones y organizaciones han desarrollado guías centradas en la reducción de la sobrecarga cognitiva, la simplificación del lenguaje y la estructuración clara de la información. Destacan los trabajos del **OACEX (Observatorio de Accesibilidad Cognitiva)** [22], vinculado a Plena Inclusión, que enfatizan la necesidad de adaptar los contenidos para facilitar una comprensión efectiva, más allá del cumplimiento formal de criterios técnicos. Aspectos como el uso de frases cortas, vocabulario concreto, estructuras predecibles o apoyos visuales coherentes son elementos clave de este enfoque, especialmente relevantes en el diseño de materiales educativos para alumnado con necesidades específicas de apoyo cognitivo. El análisis conjunto de estos marcos normativos y de referencia pone de manifiesto una **convergencia significativa** entre ellos, especialmente en lo relativo a los aspectos cognitivos de la adaptación de materiales educativos. Elementos como la claridad lingüística, la estructuración de la información, la previsibilidad de las tareas y el uso de apoyos visuales aparecen de forma recurrente en todos estos enfoques, destacando especialmente los informes de UNESCO en esta línea [31]. Esta coincidencia sugiere la existencia de un núcleo común de principios que, aunque formulados desde perspectivas diferentes, persiguen un mismo objetivo: facilitar la comprensión y el acceso efectivo al aprendizaje.

Por este motivo, todos los estándares y normativas mencionados han sido considerados como base para la definición de la propuesta presentada en este Trabajo Fin de Grado. El objetivo no es sustituir ni redefinir dichos marcos, sino analizarlos de forma conjunta y extraer de ellos sus principios fundamentales, con el fin de permitir su aplicación sistemática mediante una solución computacional orientada a la inclusión educativa.

1.3.2. Uso de apoyos visuales y recursos multimedia en entornos educativos inclusivos

Los marcos normativos expuestos en la subsección anterior coinciden en señalar el uso de apoyos visuales y recursos multimedia como una estrategia fundamental en el ámbito de la educación inclusiva. Especialmente CAST (2018) en las directrices DUA [5] expone que la incorporación de elementos visuales, especialmente pictogramas e imágenes de contexto, permite **crear múltiples vías de acceso a la información**: mientras que algunos alumnos procesan mejor el lenguaje escrito, otros comprenden más eficazmente a través de representaciones visuales. Reduciendo la carga cognitiva asociada al procesamiento de textos complejos y permitiendo que el alumnado centre sus recursos mentales en la comprensión del contenido, no en el desciframiento de las instrucciones.

Desde un enfoque académico, Paivio (1986) ya demostró que la combinación de información verbal y visual favorece el aprendizaje al activar múltiples canales de procesamiento mediante su teoría de la codificación dual [20], posteriormente consolidada por los principios del aprendizaje multimedia de Mayer (2009) ([16]. Estos enfoques teóricos sostienen que el alumnado comprende y retiene mejor la información cuando esta se presenta mediante diferentes formatos complementarios, siempre que exista una coherencia semántica entre los elementos visuales y el contenido textual.

En el ámbito de la educación inclusiva, los apoyos visuales resultan especialmente relevantes para alumnado con trastornos del espectro autista (TEA), dificultades específicas del lenguaje, discapacidad intelectual, problemas de atención o necesidades educativas especiales de carácter cognitivo. Sin duda, perfiles más que habituales entre el alumnado

con necesidades educativas especiales. No obstante, las directrices DUA desarrolladas por CAST [5] confirman que su utilidad no se limita a estos perfiles, sino que beneficia de forma generalizada a estudiantes con bajo nivel de competencia lectora, alumnado de incorporación tardía al sistema educativo y, en general, a cualquier persona que se encuentre en situación de vulnerabilidad comunicativa. De hecho, Rose y Meyer (2002) ya establecieron en el marco del DUA que la provisión de múltiples formas de representación constituye un principio fundamental para atender a la diversidad del alumnado [27]. Desde esta perspectiva, los apoyos visuales y multimedia no deben concebirse como adaptaciones excepcionales dirigidas únicamente a determinados estudiantes, sino como elementos de diseño pedagógico aplicables al conjunto del alumnado.

Dentro de los apoyos visuales, cabe mencionar especialmente los pictogramas. ARASAAC [6] constituye el referente internacional en recursos pictográficos de libre acceso para sistemas aumentativos y alternativos de comunicación (SAAC), habiendo sido ampliamente adoptado en entornos educativos, sanitarios y sociales por su rigor pedagógico y comunicativo alineado con las necesidades de accesibilidad cognitiva. Sin embargo, la eficacia de los apoyos visuales no se limita al uso de pictogramas: imágenes de contexto, esquemas visuales o representaciones gráficas complementarias desempeñan un papel fundamental en la construcción de significado, especialmente cuando los contenidos implican procesos, secuencias de acciones o conceptos abstractos.

No obstante, Mayer (2009) advierte que una selección inadecuada de imágenes o una falta de correspondencia semántica entre texto y elementos visuales puede generar confusión o aumentar la carga cognitiva [16], subrayando la necesidad de que los apoyos visuales estén cuidadosamente contextualizados y alineados con la intención pedagógica del contenido.

En síntesis, la combinación de pictogramas, imágenes de contexto y otros recursos visuales, respaldada por marcos teóricos consolidados como DUA e iniciativas multimedia como ARASAAC, proporciona una base sólida para el desarrollo de soluciones educativas digitales orientadas a mejorar la comprensión y favorecer el acceso equitativo al aprendizaje.

1.3.3. Sistemas inteligentes aplicados a la adaptación de contenidos educativos

En los últimos años, los sistemas inteligentes basados en inteligencia artificial han adquirido una relevancia creciente en el ámbito educativo, especialmente en relación con la personalización y adaptación del aprendizaje. Investigaciones como las de Luckin et al. (2016) [15] y organismos internacionales como la OCDE (2021) [19] evidencian que la capacidad de ajustar la presentación, el nivel de complejidad y la estructura de la información a las características del alumnado constituye un factor clave para **mejorar la comprensión, la motivación y el rendimiento académico**. Particularmente Holmes (2019) destaca su utilidad en contextos educativos diversos y heterogéneos [14], una situación que más actualemnte UNESCO (2023) ha refrendado [30].

Uno de los ámbitos en los que la inteligencia artificial ha demostrado un mayor potencial es la adaptación lingüística de los contenidos educativos. Estudios como los de Snow y Uccelli (2009) [28] han evidenciado que una parte significativa de las dificultades del alumnado no reside en la falta de capacidades cognitivas, sino en la complejidad del lenguaje utilizado en los enunciados y materiales didácticos. Un sistema inteligente basado en inteligencia artificial podría analizar la estructura sintáctica y semántica de los textos, detectar elementos potencialmente problemáticos y reformularlos mediante un lenguaje más claro, frases más breves y estructuras más explícitas. Este tipo de intervención resulta especialmente relevante

en contextos de educación inclusiva, donde la claridad del lenguaje condiciona el acceso efectivo al aprendizaje.

En este ámbito, herramientas como **Rewordify.com** han demostrado utilidad para la simplificación léxica automática de textos mediante la sustitución de palabras complejas por sinónimos más sencillos [26], siendo ampliamente utilizada en contextos educativos para mejorar la fluidez lectora. No obstante, estas soluciones se centran principalmente en la simplificación vocabular sin abordar la estructura sintáctica, la organización pedagógica o la incorporación de apoyos multimodales contextualizados.

En el contexto de los sistemas aumentativos y alternativos de comunicación (SAAC), el Portal Aragonés de la Comunicación Aumentativa y Alternativa (ARASAAC) ha desarrollado **ARAWrite**, una herramienta web que emplea procesamiento del lenguaje natural para convertir textos en frases acompañadas de pictogramas de forma automática [23]. Si bien ARAWrite resulta útil para la creación rápida de materiales con apoyos visuales, su funcionalidad se limita a frases breves y oraciones simples, sin contemplar la adaptación pedagógica estructurada de enunciados educativos complejos que requiere la educación formal [3].

Desde otra perspectiva, la iniciativa **Learn Your Way** de Google Labs representa un enfoque orientado a la personalización multimodal de contenidos educativos mediante modelos de lenguaje avanzados (LearnLM integrado en Gemini) [13]. El sistema genera múltiples representaciones del contenido (mapas mentales, cuestionarios adaptativos, lecciones narradas) y adapta el nivel de complejidad según el grado académico e intereses del estudiante, mostrando mejoras significativas en la retención de conocimientos [12]. Sin embargo, su enfoque se centra en la personalización por nivel e intereses personales, no en la aplicación sistemática de estándares de accesibilidad cognitiva como la norma UNE 71362:2020 o los principios del Diseño Universal para el Aprendizaje orientados específicamente a la reducción de barreras para alumnado con necesidades educativas especiales.

En el ámbito de la investigación académica, trabajos recientes como el sistema ALGA-Ed proponen marcos integrados que combinan perfiles de usuario, modelos generativos y retroalimentación adaptativa para estudiantes con TDAH, dislexia y discapacidades sensoriales [10]. En esa misma línea, diversos estudios analizan el impacto de los Sistemas Tutores Inteligentes que utilizan procesamiento del lenguaje natural para adaptar el ritmo de aprendizaje a necesidades individuales [9] [17]. Estas iniciativas demuestran el creciente interés en aplicar técnicas de IA a la educación inclusiva, demostrando por lo general positivos resultados, aunque en muchos casos su enfoque se centra en aspectos eminentemente técnicos de accesibilidad sin abordar de manera integral la dimensión pedagógica y estructural de los contenidos educativos [11]. A su vez, estos estudios no contemplan el uso de los mismos por parte del profesorado. Un hecho que organismos como la OCDE (2021) enfatizan en importancia, destacando que estos sistemas sean transparentes y comprensibles para el profesorado, de modo que puedan ser percibidos como herramientas de apoyo y no como mecanismos opacos que sustituyen el criterio docente [19].

En síntesis, los trabajos revisados evidencian que los sistemas inteligentes basados en tecnologías de inteligencia artificial constituyen en concepto un recurso con un elevado potencial para abordar algunas de las principales problemáticas detectadas en la educación inclusiva actual. Sin embargo, y bajando a niveles de detalle operativos, cabe comentar que las soluciones actuales presentan limitaciones relevantes que en sí mismas justifican el desarrollo de este trabajo:

1. La mayoría se centra en la simplificación léxica sin abordar la estructura pedagógica completa del contenido.
2. No integran de forma sistemática estándares de accesibilidad cognitiva como la norma UNE 71362:2020 o los principios DUA.
3. Carecen de mecanismos de transparencia que permitan al docente comprender y validar las transformaciones realizadas.

Al permitir la adaptación automática, coherente y escalable de los contenidos educativos mediante un formato estructurado y predecible, el presente trabajo pretende contribuir a reducir estas barreras, facilitando el acceso equitativo al aprendizaje y avanzando hacia una aplicación más efectiva de los marcos de inclusión educativa en contextos reales de enseñanza-aprendizaje.

1.4. CONTRIBUCIONES PRINCIPALES DEL TRABAJO

El presente Trabajo Fin de Grado realiza tres aportaciones principales que pretenden contribuir a potenciar la inclusión educativa mediante el uso de tecnologías de inteligencia artificial:

1. Definición del formato Lectura Educativa Inclusiva (LEdui). Se ha diseñado y formalizado un formato estructurado de adaptación de contenidos educativos que sintetiza de manera operativa principios procedentes del Diseño Universal para el Aprendizaje (DUA), la Lectura Fácil, las directrices WCAG, la norma UNE 71362:2020 y principios de accesibilidad cognitiva. A diferencia de los sistemas de IA generativa tradicionales, que producen salidas variables e impredecibles, LEdui define una estructura fija y consistente que incluye siempre objetivos de aprendizaje explícitos, pasos guiados secuenciales, opciones de entrega multimodales, apoyos visuales contextualizados (pictogramas e imágenes), preguntas de reflexión metacognitiva y estimaciones temporales. Esta estructura predecible facilita tanto la autorregulación del alumnado como la anticipación del resultado por parte del profesorado, aspectos críticos para la adopción de tecnología en contextos educativos reales.

2. Desarrollo de un servicio web basado en IA controlada y determinista. Se ha implementado un sistema informático completo que materializa el formato LEdui mediante una arquitectura cliente–servidor construida con FastAPI, PostgreSQL y autenticación OAuth 2.0 (Google). El sistema integra modelos de inteligencia artificial generativa (Claude 3.7 Sonnet y Google Gemini) de forma controlada, garantizando que las transformaciones realizadas respeten siempre la estructura definida por LEdui y se alineen con los estándares de accesibilidad cognitiva establecidos. Además, el sistema incorpora la generación automática de pictogramas educativos procedentes del repositorio ARASAAC y la creación de imágenes de contexto mediante IA, asegurando su alineación semántica con el contenido específico de cada enunciado. Complementariamente, el sistema genera explicaciones pedagógicas dirigidas al profesorado que detallan las transformaciones aplicadas y las razones pedagógicas que las sustentan, promoviendo un uso reflexivo y fundamentado de la tecnología.

3. Enfoque centrado en el docente como agente de inclusión. A diferencia de otras herramientas de IA aplicadas a la educación, donde el profesorado es considerado un usuario más, LEdui se ha concebido desde el principio con el docente como protagonista y agente activo del proceso de inclusión. El sistema no pretende sustituir el criterio pedagógico del profesorado, sino actuar como herramienta de apoyo que le permita materializar de manera eficiente principios inclusivos que, de otro modo, requerirían un tiempo y una especialización técnica difícilmente asumibles en la práctica educativa cotidiana. Esta filosofía se

materializa en funcionalidades específicas como la generación de explicaciones pedagógicas que detallan las transformaciones realizadas y su fundamentación teórica, permitiendo al docente comprender, valorar y, si lo considera oportuno, modificar las adaptaciones propuestas. El sistema ha sido validado mediante pruebas técnicas y evaluación cualitativa por parte de docentes, orientadores educativos y profesionales de editorial, confirmando su utilidad práctica como herramienta de apoyo al profesorado en contextos de educación inclusiva.

En conjunto, estas contribuciones ponen de manifiesto que es posible combinar el potencial de los modelos de lenguaje de gran tamaño con el rigor de los estándares pedagógicos y de accesibilidad, ofreciendo al profesorado herramientas prácticas que faciliten la creación de contenidos educativos más claros, estructurados e inclusivos.

1.5. ESTRUCTURA DEL DOCUMENTO

El presente documento se estructura en una serie de capítulos que permiten presentar de forma progresiva el contexto, los objetivos, el desarrollo y los resultados del Trabajo Fin de Grado, así como las conclusiones derivadas del mismo.

1. **El Capítulo 1, Introducción,** en el que está esta sección, expone la motivación del trabajo, la problemática concreta que se pretende abordar y el contexto disciplinar y tecnológico en el que se enmarca la propuesta desarrollada.
2. **En el Capítulo 2, Objetivo,** se define el objetivo general del proyecto y se detallan los objetivos complementarios que guían el desarrollo del trabajo.
3. **El Capítulo 3, Plan de gestión del trabajo,** describe la metodología empleada, las fases en las que se ha estructurado el desarrollo del proyecto y los recursos técnicos utilizados para su realización.
4. **El Capítulo 4, Desarrollo y resultados,** presenta la definición y validación del formato de Lectura Educativa Inclusiva (LEdui), el diseño e implementación del sistema inteligente desarrollado, la arquitectura del servicio web y la interfaz de usuario, así como los resultados obtenidos tras su despliegue y validación.
5. **Por último, el Capítulo 5,** Conclusiones, recoge el análisis del grado de cumplimiento de los objetivos planteados y de los requisitos definidos, la justificación de las competencias adquiridas durante los estudios de grado, una reflexión sobre los trabajos derivados y futuros, y una valoración personal del trabajo realizado.

El documento se completa con la Bibliografía, que incluye las referencias citadas a lo largo de la memoria, y con los Anexos, donde se recogen contenidos complementarios que amplían la información presentada en los capítulos principales.

CAPÍTULO 2
Objetivo

En este capítulo se presentan los objetivos del presente Trabajo Fin de Grado, definidos a partir de la problemática expuesta en el capítulo anterior. Estos objetivos delimitan el alcance del proyecto y orientan el diseño y desarrollo de la solución propuesta.

2.1. OBJETIVO GENERAL

El objetivo general se ha definido con el propósito de abordar de forma integrada la adaptación inclusiva de contenidos educativos y su materialización en una solución software funcional, combinando criterios pedagógicos y tecnológicos. Siendo el mismo:

OG. *El objetivo general de este Trabajo Fin de Grado es el desarrollo de un servicio web orientado a la transformación automática de contenidos educativos textuales en formato de Lectura Educativa Inclusiva (LEdui), mediante el uso de inteligencia artificial.*

2.2. OBJETIVOS COMPLEMENTARIOS

Para alcanzar el objetivo general definido en la sección justo anterior, se establecen los siguientes objetivos complementarios:

OC1. Identificar y analizar los principales marcos normativos y estándares relacionados con la educación inclusiva y la accesibilidad cognitiva, con el fin de extraer criterios aplicables al diseño de materiales educativos digitales.

OC2. Definir formalmente el formato de Lectura Educativa Inclusiva (LEdui), especificando un conjunto estructurado de reglas que puedan ser aplicadas de manera sistemática a contenidos educativos textuales.

OC3. Diseñar un modelo de transformación automática de contenidos educativos basado en dichas reglas, que permita convertir textos originales en contenidos adaptados al formato LEdui.

OC4. Implementar un servicio web que materialice el modelo de transformación definido, exponiendo una API capaz de procesar y adaptar contenidos educativos de forma automática.

OC5. Integrar al servicio web las funcionalidades necesarias para la generación de apoyos visuales mediante la incorporación de pictogramas y recursos multimedia asociados al contenido textual adaptado, garantizando su coherencia semántica.

OC6. Desarrollar una interfaz web que permita al usuario interactuar con el sistema y visualizar los contenidos adaptados de manera clara y estructurada.

OC7. Evaluar el funcionamiento del sistema mediante casos de uso representativos, aplicándolo al menos a 15 enunciados educativos de distintas áreas y niveles educativos, y verificando el cumplimiento de las reglas LEdui definidas en cada transformación.

OC8. Analizar el rendimiento y la escalabilidad del sistema desarrollado, evaluando su idoneidad para un uso real en escenarios de concurrencia moderada y crecimiento progresivo del número de usuarios.

En conjunto, estos objetivos concretan el alcance del proyecto y establecen la base para el desarrollo, la evaluación e incluso la futura expansión del sistema propuesto.

CAPÍTULO 3
Plan de gestión del trabajo

Este capítulo describe la planificación y metodología seguida en el desarrollo del presente Trabajo Fin de Grado, detallando las fases en las que se ha estructurado y los recursos técnicos empleados.

El objetivo es ofrecer una visión organizada del proceso, proporcionando el marco necesario para comprender cómo se han abordado las distintas etapas y cómo se han articulado los elementos metodológicos, temporales y técnicos que han dado lugar a la solución final.

3.1. METODOLOGÍA DE TRABAJO

La metodología de trabajo adoptada en este Trabajo Fin de Grado se encuadra dentro de un enfoque de desarrollo iterativo e incremental, próximo a la metodología de **Rapid Application Development (RAD)**, y alejado deliberadamente de modelos estrictamente secuenciales como SSADM. Esta elección responde a la naturaleza aplicada del proyecto y a la necesidad de validar de forma progresiva tanto las decisiones técnicas como las funcionales [24] [29].

En este contexto, el desarrollo del sistema se ha planteado siguiendo los principios de RAD, priorizando la construcción temprana de prototipos funcionales y la validación iterativa de las decisiones adoptadas. Este enfoque ha permitido materializar rápidamente las ideas conceptuales en componentes ejecutables, evaluar su viabilidad técnica y ajustar el diseño del sistema conforme se identificaban nuevas necesidades o limitaciones. La iteración sobre prototipos ha resultado especialmente relevante en la definición del modelo de transformación de contenidos y en la integración de sistemas inteligentes mediante APIs externas.

Desde el punto de vista del diseño del software, el trabajo se apoya en principios propios del **Object-Oriented Design (OOD)** [4], especialmente en lo relativo a la modularidad, la separación de responsabilidades y la reutilización de componentes. Aunque no se ha seguido de forma estricta un proceso formal de modelado UML completo, el diseño del sistema se ha estructurado en torno a entidades claramente definidas, servicios desacoplados y componentes con responsabilidades bien delimitadas, lo que ha facilitado la mantenibilidad y extensibilidad del sistema desarrollado.

Desde una perspectiva conceptual más amplia, este proyecto podría enmarcarse también dentro de una **metodología de Aprendizaje-Servicio (ApS)**. El Aprendizaje-Servicio es un enfoque pedagógico que integra el aprendizaje académico con el servicio a la comunidad, permitiendo que el estudiante aplique conocimientos técnicos mientras genera un impacto social positivo [25]. En este caso, el desarrollo del sistema no responde únicamente a un ejercicio académico de ingeniería del software, sino que tiene una vocación explícita de

utilidad real: mejorar la accesibilidad educativa y facilitar la inclusión de alumnado con necesidades educativas especiales. Esta dimensión de servicio ha condicionado las decisiones de diseño, orientando el proyecto hacia una solución práctica, escalable y alineada con las necesidades reales del profesorado y de todo el alumnado.

Asimismo, el proceso de desarrollo ha incorporado de forma indirecta elementos característicos de las **metodologías ágiles**, como el énfasis en la obtención temprana de software funcional, la priorización de funcionalidades clave y la evaluación continua del resultado frente a los objetivos del trabajo. No obstante, dado el carácter individual del proyecto y el contexto académico en el que se desarrolla, no se ha aplicado de manera formal un marco concreto como Scrum o Kanban, sino que se han adoptado sus principios generales de forma pragmática y adaptada a la realidad del TFG.

En conjunto, la metodología adoptada combina un desarrollo iterativo basado en prototipos (RAD), principios de diseño orientado a objetos (OOD), un enfoque de Aprendizaje-Servicio y elementos de metodologías ágiles, permitiendo abordar el trabajo de manera flexible, estructurada y coherente con su doble dimensión técnica y social. Este planteamiento metodológico ha resultado adecuado para garantizar la alineación entre los objetivos definidos, el proceso de desarrollo seguido y la solución software finalmente obtenida.

3.2. FASES DE DESARROLLO Y ENTREGABLES

El desarrollo del proyecto se ha organizado en fases consecutivas, siguiendo un enfoque progresivo e incremental. Cada fase ha producido entregables concretos que han servido como base para la siguiente, asegurando la trazabilidad entre objetivos, desarrollo y resultados.

3.2.1. Fase 1. Definición del estándar LEdui

Esta fase analiza los principales marcos normativos y pedagógicos relacionados con la educación inclusiva y la accesibilidad cognitiva, para definir un estándar de Lectura Educativa Inclusiva (LEdui) formulado de manera analítica y orientada a su aplicación computacional.

Entregables:

1. **Documento de análisis de marcos normativos y estándares de inclusión educativa.**
2. **Definición formal de las 10 reglas del estándar LEdui.**

El primero está resumido en el capítulo 1 (sección de Marcos normativos de inclusión educativa) y de forma más aterrizada en el capítulo 4. La definición formal de las reglas LEdui se incluye también se incluye en el capítulo 4, así como en el anexo A.

3.2.2. Fase 2. Definición de requisitos del sistema

Esta fase traduce el estándar LEdui en un conjunto de requisitos funcionales que permitan su implementación en un sistema software de adaptación automática de contenidos.

Entregables:

1. **Especificación de requisitos del MVP** (Minimum Viable Product).
2. **Identificación de requisitos deseables** para iteraciones posteriores.

Ambos entregables se resumen en el capítulo 4.

3.2.3. Fase 3. Desarrollo del producto mínimo viable (MVP)

Esta fase implementa una primera versión funcional del sistema que permite validar la viabilidad técnica del estándar LEdui y del modelo de transformación automática de contenidos.

Entregables:

1. **MVP del servicio web** de adaptación de contenidos educativos.
2. **API funcional** para la adaptación automática de textos educativos.

Desarrollado en el capítulo 4.

3.2.4. Fase 4. Desarrollo iterativo y ampliación de funcionalidades

Esta fase evoluciona el MVP mediante iteraciones sucesivas, incorporando nuevas funcionalidades y mejoras a partir de los requisitos identificados durante el uso y evaluación del sistema.

Entregables:

1. **Integración de apoyos visuales y recursos multimedia** en la API.
2. **Desarrollo de la interfaz web** para el usuario final.
3. **Gestión de usuarios y persistencia de datos.**
4. **Mejoras iterativas** en la arquitectura y en el modelo de transformación.

Desarrollado en el capítulo 4.

3.2.5. Fase 5. Validación y refinamiento final

Esta fase evalúa el funcionamiento global del sistema y la adecuación de las adaptaciones generadas al estándar LEdui, introduciendo los ajustes necesarios antes de la finalización del trabajo.

Entregables:

1. **Resultados de pruebas** funcionales y de integración.
2. **Evaluación cualitativa** de los contenidos adaptados.
3. **Ajustes finales** del estándar LEdui y del sistema software.
4. **Versión final** del sistema y de la memoria del TFG.

Desarrollado en el capítulo 5.

3.3. CRONOGRAMA Y DEDICACIÓN TEMPORAL

El desarrollo del proyecto se ha extendido a lo largo de **5 meses** (1 de septiembre de 2025 a 31 de enero de 2026), con una dedicación total estimada de **310 horas**. Esta cifra refleja la complejidad inherente al proyecto: el diseño de un estándar educativo formal (LEdui), el desarrollo de un sistema software completo con arquitectura cliente-servidor, la integración de múltiples servicios de inteligencia artificial, y la validación con profesionales del ámbito educativo. La distribución temporal del trabajo se ha ajustado en función de la disponibilidad horaria y de la propia naturaleza de cada fase.

3.3.1. Distribución de horas por fase

La tabla 3.1 muestra la distribución de horas dedicadas a cada fase del proyecto, incluyendo tanto el tiempo de desarrollo técnico como el de documentación y validación.

Tabla 3.1: Distribución de horas por fase del proyecto

Fase	Horas	% del total	Periodo
Fase 1. Definición del estándar LEdui	45h	14.5 %	Sep 2025
Fase 2. Definición de requisitos	35h	11.3 %	Sep-Oct 2025
Fase 3. Desarrollo MVP	65h	21.0 %	Oct-Nov 2025
Fase 4. Desarrollo iterativo	105h	33.9 %	Nov 2025-Ene 2026
Fase 5. Validación y refinamiento	40h	12.9 %	Ene 2026
Redacción de memoria	20h	6.5 %	Dic 2025-Ene 2026
TOTAL	**310h**	**100 %**	**Sep 2025-Ene 2026**

Como puede observarse, la mayor parte del esfuerzo (33.9 %) se ha concentrado en la Fase 4 de desarrollo iterativo. Le sigue en importancia la Fase 3 de desarrollo del MVP (21.0 %), que ha supuesto la materialización inicial del sistema.

3.3.2. Temporalidad del proyecto

La tabla 3.2 presenta el cronograma temporal del proyecto, mostrando la duración de cada fase y su distribución a lo largo del periodo de desarrollo. Las fases se han solapado parcialmente, reflejando la naturaleza iterativa del desarrollo, en la que la validación de una fase podía coincidir con el inicio de la siguiente.

Tabla 3.2: Cronograma temporal del proyecto (Sep 2025 - Ene 2026)

Fase / Mes	Sep	Oct	Nov	Dic	Ene
Fase 1. Definición LEdui	X				
Fase 2. Requisitos	X	X			
Fase 3. MVP		X	X		
Fase 4. Desarrollo iterativo			X	X	X
Fase 5. Validación					X
Redacción memoria				X	X

Este enfoque ha permitido avanzar de forma progresiva, validando decisiones técnicas y funcionales antes de incorporar nuevas funcionalidades, y garantizando que el sistema final cumpla con los objetivos definidos en el capítulo 2 del presente trabajo.

3.4. RECURSOS

Esta sección describe de forma concreta los principales recursos técnicos empleados, incluyendo las versiones específicas utilizadas, así como la función que desempeña cada uno de ellos dentro del sistema desarrollado.

La selección de estos recursos se ha realizado atendiendo a criterios de adecuación técnica, disponibilidad, compatibilidad con arquitecturas basadas en servicios y alineación con buenas prácticas de ingeniería del software. El objetivo ha sido disponer de las tecnologías estrictamente necesarias para implementar de forma eficaz y coherente la solución planteada, evitando complejidad innecesaria.

3.4.1. Entorno de desarrollo

El desarrollo del sistema se ha realizado en un ordenador portátil **Lenovo IdeaPad Slim 3 15IRH8**. El mismo está equipado con:

- Procesador Intel Core i7-13620H de 13ª generación
- 16 GB de memoria RAM
- Una unidad de almacenamiento SSD NVMe de 1 TB, con el sistema operativo Ubuntu 24.04 LTS.

Estas características han permitido ejecutar de forma simultánea el entorno de desarrollo, los servicios backend, la base de datos y las pruebas locales del sistema sin limitaciones significativas de rendimiento.

Como entorno de desarrollo integrado (IDE) se ha utilizado **Visual Studio Code**, seleccionado por su soporte multiplataforma, su compatibilidad excelente con el ecosistema Linux, su amplio conjunto de extensiones para implementar aplicaciones web y su capacidad para facilitar la edición, depuración y organización del código a lo largo de todo el proceso de desarrollo.

3.4.2. *Backend* y arquitectura del servicio web

El *backend* del sistema se ha implementado utilizando el lenguaje de programación **Python (versión 3.11)**, elegido por su amplia adopción en el ámbito del desarrollo de servicios web y de los sistemas inteligentes, así como por la disponibilidad de bibliotecas especializadas para la integración con APIs externas.

Para la implementación del servicio web se ha empleado el framework **FastAPI (versión 0.115.5)**, que permite el desarrollo de APIs REST de alto rendimiento. FastAPI ha sido seleccionado por las siguientes razones:

- Alto rendimiento y baja latencia.
- Validación automática de datos mediante esquemas Pydantic (versión 2.12.5).
- Facilidad para la integración de servicios externos mediante APIs.
- Arquitectura clara y mantenible.

El servidor de aplicaciones utilizado es **Uvicorn (versión 0.24.0)**, un servidor ASGI de alto rendimiento especialmente optimizado para aplicaciones asíncronas en Python.

El sistema sigue una arquitectura cliente–servidor, en la que el *backend* expone distintos endpoints destinados a la adaptación de contenidos educativos, la gestión de usuarios y el acceso a los resultados generados.

3.4.3. Sistemas inteligentes e Inteligencia Artificial utilizada

La adaptación automática de los contenidos educativos se ha basado en el uso de servicios de Inteligencia Artificial consumidos mediante APIs, integrados directamente en el *backend* del sistema. Este enfoque ha permitido incorporar capacidades avanzadas de análisis y generación de contenido sin acoplar la lógica del sistema a un proveedor concreto, favoreciendo así una arquitectura modular y extensible.

Para el procesamiento y transformación de los contenidos textuales se ha utilizado el modelo **Claude 3.7 Sonnet de Anthropic**, accedido mediante la biblioteca `anthropic` (versión 0.75.0). Este modelo se ha empleado para el análisis semántico y sintáctico de los textos educativos, la reformulación de contenidos conforme a criterios de claridad y accesibilidad, y la generación de explicaciones complementarias alineadas con el formato LEdui.

Para la generación de recursos visuales se ha integrado el servicio **Google Gemini** (accedido mediante la biblioteca `google-genai` versión 1.52.0), empleando específicamente el modelo Gemini 2.5 Flash para la generación de imágenes educativas de contexto. Este modelo ha demostrado resultados satisfactorios en términos de calidad visual, coherencia con el contenido educativo y tiempo de respuesta.

Adicionalmente, el sistema integra el repositorio de pictogramas **ARASAAC** mediante peticiones HTTP a su API pública. Estos pictogramas se emplean como apoyo visual complementario al contenido textual adaptado, contribuyendo a reforzar la comprensión del significado de las tareas propuestas.

La integración de todos estos sistemas inteligentes mediante llamadas a APIs HTTP ha permitido desacoplar el núcleo funcional del sistema de los proveedores externos de Inteligencia Artificial, facilitando la sustitución o ampliación futura de los servicios utilizados y manteniendo la coherencia con las buenas prácticas de ingeniería del software.

3.4.4. Base de datos y persistencia

Para la persistencia de la información generada por el sistema se ha utilizado **PostgreSQL (versión 15)** como sistema gestor de bases de datos relacional. La integración con el *backend* se realiza mediante **SQLAlchemy** (versión 2.0.36), un **ORM** (Object-Relational Mapping) que facilita la gestión de modelos de datos y consultas de forma orientada a objetos.

La base de datos se emplea para almacenar información relacionada con los usuarios, el historial de adaptaciones realizadas y los metadatos asociados a los contenidos adaptados. PostgreSQL ha sido seleccionado por su fiabilidad, robustez y capacidad para gestionar datos estructurados de forma eficiente en aplicaciones web de carácter transaccional. El acceso a la base de datos desde Python se realiza mediante el driver `psycopg2-binary` (versión 2.9.9).

3.4.5. Autenticación

La autenticación de usuarios del sistema se ha implementado mediante la integración con los servicios de Google, utilizando el protocolo **OAuth 2.0.** La implementación se ha realizado mediante la biblioteca Authlib (versión 1.3.0), que proporciona soporte completo para flujos de autenticación OAuth. Esta solución permite delegar el proceso de identificación en un proveedor ampliamente adoptado, reduciendo la gestión directa de credenciales y reforzando la seguridad del acceso.

3.4.6. Interfaz de usuario y tecnologías web

La interfaz de usuario se ha desarrollado utilizando tecnologías web estándar: **HTML5, CSS3 y JavaScript (estándar ES6+)**, garantizando la compatibilidad con distintos navegadores y dispositivos. El diseño de la interfaz se ha orientado a la claridad, la simplicidad y la reducción de la carga cognitiva, en coherencia con los principios del formato de Lectura Educativa Inclusiva (LEdui).

Para la comunicación entre el *frontend* y el *backend* se utiliza la biblioteca `httpx` (versión 0.28.1) en el lado del servidor, que facilita las peticiones HTTP asíncronas de forma eficiente.

3.4.7. Despliegue y entorno de pruebas

Para la validación del sistema en un entorno cercano a producción se ha utilizado la plataforma **Railway**, que permite el despliegue ágil de aplicaciones web y servicios *backend*. El uso de este entorno ha resultado especialmente útil para realizar pruebas compartidas entre las distintas personas que han colaborado en la validación, ya que facilita el acceso remoto al sistema sin necesidad de configuraciones locales adicionales.

Este enfoque ha permitido comprobar de forma conjunta la estabilidad del servicio, la correcta integración entre *backend*, base de datos y *frontend*, y el comportamiento del sistema en condiciones de uso real, favoreciendo una evaluación más eficiente y coordinada del funcionamiento global del sistema.

3.4.8. Estimación de costes del proyecto

Además de los aspectos técnicos, resulta relevante estimar el coste económico total del proyecto con el fin de evaluar su viabilidad en un contexto de desarrollo profesional o de una posible implementación real en entornos educativos. Este análisis permite contextualizar el esfuerzo realizado y valorar la sostenibilidad del sistema más allá del ámbito académico. Las cifras presentadas a continuación corresponden a precios de referencia de las tecnologías utilizadas a fecha de realización del proyecto, concretamente durante el último trimestre de 2025 y principios del año 2026, y deben entenderse como una estimación orientativa sujeta a variaciones del mercado.

El desarrollo del sistema ha requerido una dedicación aproximada de **310 horas**, distribuidas a lo largo de cinco meses, abarcando tareas de análisis, diseño, implementación, pruebas y despliegue. Para su valoración económica se ha considerado una tarifa conservadora de **25 €/hora**, habitual para perfiles de desarrollo junior o de nivel medio en el mercado laboral español, lo que supone un coste estimado total de **7.750 €.** En el contexto de un Trabajo Fin de Grado, este esfuerzo no conlleva un coste económico real, pero permite dimensionar de forma realista el valor del desarrollo y facilita la comparación con proyectos de características similares si se abordara como un encargo profesional o una iniciativa institucional.

Durante el desarrollo se han utilizado servicios externos de inteligencia artificial, principalmente para procesamiento de texto y generación de imágenes. Los servicios de inteligencia artificial utilizados durante el desarrollo del proyecto presentan un coste variable en función del volumen de uso. El uso aproximado ha sido el siguiente:

- **Claude 3.7 Sonnet (Anthropic):** con un coste aproximado de 2,8 € por millón de tokens de entrada y 14 € por millón de tokens de salida, el uso estimado durante las fases de desarrollo y pruebas (en torno a 5 millones de tokens totales) ha supuesto un coste aproximado de entre **40 y 60 €**.

- **Google Gemini (generación de imágenes):** con un coste aproximado de 0,002 € por imagen generada, y un uso estimado de 500 imágenes durante las pruebas y validaciones, el coste total ha sido de aproximadamente **1 €**.
- **ARASAAC (pictogramas):** este servicio público es de acceso gratuito, por lo que no ha supuesto ningún coste económico.

El coste total de los servicios de inteligencia artificial utilizados durante el desarrollo se estima, por tanto, en torno a 50–60 €. Estos costes dependen directamente del volumen de uso del sistema y podrían incrementarse en un escenario de producción con utilización intensiva, lo que haría necesario aplicar estrategias de control como sistemas de caché, limitación de uso por usuario o modelos de acceso escalonados, tal y como se describe en el análisis de riesgos incluido en este mismo capítulo.

La infraestructura de despliegue y pruebas se ha basado en servicios de alojamiento en la nube, concretamente en Railway, una plataforma de hosting que ofrece planes desde 5 €/mes (plan Hobby) hasta 20 €/mes (plan Developer). Considerando un periodo de desarrollo de 5 meses, el coste estimado de infraestructura se sitúa entre 25 y 100 €, según el plan utilizado. No se ha requerido la contratación de dominio propio ni servicios adicionales de bases de datos, al estar estos incluidos en la plataforma utilizada.

Las herramientas de desarrollo empleadas son, en su mayoría, gratuitas o de código abierto, por lo que no han supuesto ningún coste económico adicional.

En síntesis, la tabla 3.3 resume el coste total estimado del proyecto:

Tabla 3.3: Coste económico total del proyecto

Concepto	**Coste estimado (€)**
Horas de trabajo (310h × 25€/h)	7.750 €
Servicios de IA (Claude + Gemini)	50-60 €
Infraestructura (Railway, 5 meses)	25-100 €
Herramientas de desarrollo	0 €
TOTAL ESTIMADO	**7.825 - 7.910 €**

En conjunto, el coste económico total estimado del proyecto se sitúa entre 7.825 y 7.910 €, siendo el trabajo de desarrollo el principal componente del coste. Los servicios externos y la infraestructura representan una fracción reducida del total, lo que pone de manifiesto la viabilidad económica del sistema y su potencial aplicación en contextos educativos reales.

3.5. ANÁLISIS DE RIESGOS Y PLAN DE CONTINGENCIA

El desarrollo de un sistema que integra servicios externos mediante APIs conlleva una serie de riesgos técnicos que deben ser identificados y gestionados adecuadamente. A continuación se enumeran los principales riesgos técnicos identificados durante el desarrollo del proyecto, junto con las estrategias de mitigación y contingencia previstas:

- **Riesgo 1. Dependencia de servicios externos de IA** Cambios en las APIs, modelos de autenticación o interrupciones del servicio pueden afectar a la funcionalidad principal del sistema. *Mitigación:* arquitectura modular y desacoplada, uso de capas de abstracción para las integraciones externas y configuración mediante variables de entorno, lo que permite actualizar o sustituir proveedores en apenas minutos, con un impacto mínimo.
- **Riesgo 2. Disponibilidad del servicio ARASAAC** Posibles interrupciones en el acceso a pictogramas utilizados como apoyo visual. *Mitigación:* gestión de errores que permite continuar el funcionamiento sin pictogramas. Como mejora futura, se contempla la generación de imágenes mediante IA o la integración con repositorios alternativos de pictogramas.
- **Riesgo 3. Coste de los servicios de IA** Un aumento del volumen de uso o cambios en las tarifas de los proveedores podrían incrementar los costes operativos. *Mitigación:* almacenamiento de transformaciones para evitar peticiones redundantes, monitorización de uso y, como mejora futura, la posibilidad de implementar límites según perfil de usuario.
- **Riesgo 4. Escalabilidad del sistema** Un incremento significativo de usuarios simultáneos podría afectar al rendimiento y a los tiempos de respuesta. *Mitigación:* despliegue en infraestructura escalable cloud en infrastructuras de primera nivel que permitan escalado horizontal en función de la demanda (como AWS o Google Cloud).

El análisis realizado indica que los principales riesgos del proyecto están relacionados con la dependencia de servicios externos y el uso de tecnologías de inteligencia artificial. No obstante, las decisiones arquitectónicas adoptadas (como modularidad, desacoplamiento de integraciones y configuración flexible) permiten mitigar estos riesgos de forma eficaz. En conjunto, el sistema presenta un diseño resiliente y adaptable, adecuado para su evolución y posible despliegue en contextos educativos reales.

CAPÍTULO 4
Desarrollo y resultados

Este capítulo documenta el desarrollo completo del sistema propuesto. El contenido se organiza en torno a tres ejes principales: la formalización del formato LEdui, la implementación del sistema web que automatiza su aplicación y el despliegue de su plan de pruebas.

La primera parte del capítulo establece los fundamentos conceptuales del trabajo: la definición formal de las 10 reglas LEdui derivadas del análisis de marcos normativos, su validación pedagógica y la especificación de requisitos del sistema a desarrollar. La segunda parte describe el proceso de ingeniería de software seguido: arquitectura del sistema, flujo de acciones de funcionamiento, implementación del *backend* y *frontend*, y despliegue en producción. Finalmente, se incluye una sección donde se presenta el plan de pruebas ejecutado y los resultados de la validación con un panel de expertos en educación inclusiva.

4.1. DEFINICIÓN Y VALIDACIÓN DEL ESTÁNDAR LEDUI

En esta sección se desarrolla la **Fase 1 del plan de trabajo**, centrada en la definición conceptual del formato de Lectura Educativa Inclusiva (LEdui), siguiendo tres etapas: análisis de marcos normativos, definición de 10 reglas operativas y validación del formato.

4.1.1. Análisis de coincidencias de marcos normativos de inclusión

El análisis comparativo de los principales marcos normativos y de referencia revisados en el marco conceptual de este trabajo, concretamente el Diseño Universal para el Aprendizaje (DUA), la Lectura Fácil, la norma de calidad de los materiales educativos digitales, las Pautas de Accesibilidad para el Contenido Web (WCAG) y las guías específicas de accesibilidad cognitiva, pone de manifiesto la existencia de una convergencia significativa en relación con los principios que orientan la adaptación inclusiva de los contenidos educativos.

Con el objetivo de sintetizar dichas coincidencias de forma sistemática, se han **identificado una serie de principios comunes** con presencia explícita o implícita en los distintos marcos analizados:

1. Claridad lingüística y sintáctica.
2. Estructuración guiada de la información.
3. Reducción de la carga cognitiva.
4. Explicitación de objetivos y expectativas.
5. Uso de apoyos visuales y multimodales.
6. Enfoque inclusivo y de diseño universal.

El primero de los principios coincidentes es la **claridad lingüística**, entendida como el uso de un lenguaje sencillo, directo y libre de ambigüedades. Todos los marcos analizados coinciden en la necesidad de emplear frases cortas, vocabulario común y construcciones sintácticas simples, especialmente relevantes en contextos educativos con alumnado que presenta dificultades de comprensión lectora o procesamiento cognitivo.

Un segundo principio compartido es la **estructuración guiada de la información**. Los estándares revisados insisten en la importancia de organizar los contenidos de manera predecible y ordenada, facilitando la identificación de objetivos, instrucciones, pasos intermedios y resultados esperados. Esta estructuración resulta clave para reducir la desorientación y favorecer la autonomía del alumnado.

La **reducción de la carga cognitiva** aparece igualmente como un criterio transversal. Los distintos marcos recomiendan priorizar la información esencial, evitar la sobrecarga informativa y eliminar elementos superfluos que puedan interferir en la comprensión de la tarea, permitiendo así un procesamiento más eficiente de la información.

Otro punto de convergencia relevante es la **explicitación de expectativas**, que se traduce en indicar de forma clara qué debe hacer el alumnado, cómo debe hacerlo y bajo qué criterios se evaluará su trabajo. Este principio reduce la incertidumbre, mejora la autorregulación y favorece una participación más activa en el proceso de aprendizaje.

Asimismo, los estándares coinciden en la conveniencia de incorporar **apoyos visuales y multimodales** como complemento al texto escrito, tal y como se ha expuesto en el capítulo 1. Aunque con distinto grado de formalización, todos reconocen el valor de imágenes, pictogramas, esquemas u otros recursos visuales para reforzar el significado del contenido y facilitar su comprensión.

Finalmente, todos los marcos analizados comparten un **enfoque inclusivo basado en el diseño universal**, entendiendo la accesibilidad no como una adaptación puntual o excepcional, sino como un criterio que debe integrarse desde el origen en el diseño de los materiales educativos, con el objetivo de atender a la diversidad del alumnado de forma sistemática.

En conjunto, este análisis pone de manifiesto la existencia de una base conceptual común sólida entre los distintos estándares de accesibilidad e inclusión. Estas coincidencias han servido como punto de partida para la definición del formato de Lectura Educativa Inclusiva (LEdui), permitiendo abstraer los principios compartidos y formalizarlos en un conjunto de reglas operativas orientadas a su aplicación automática mediante sistemas inteligentes.

4.1.2. Definición de las 10 reglas LEdui

Los principios comunes identificados en la subsección anterior constituyen una base conceptual sólida, pero resultan insuficientes para su aplicación directa y, sobre todo, sistemática en el diseño de contenidos educativos, especialmente cuando se pretende automatizar su aplicación mediante sistemas inteligentes.

El formato de Lectura Educativa Inclusiva (LEdui) traduce estos principios en **10 reglas operativas** que pueden aplicarse tanto de forma manual por el profesorado como de manera automática por un sistema computacional. Estas reglas no pretenden sustituir ni redefinir los marcos normativos existentes, sino operacionalizar sus principios compartidos en un formato más concreto, accionable y verificable.

Las reglas pueden aplicarse de forma independiente, pero su efectividad se ve reforzada

cuando se combinan entre sí, ofreciendo un resultado mayor que la estricta suma de los resultados individuales. A continuación se presentan las 10 reglas que conforman el formato LEdui:

1. **Claridad lingüística.** Los textos deben redactarse utilizando frases cortas, voz activa y vocabulario cotidiano. Los tecnicismos, siempre que sean necesarios, deben explicarse mediante ejemplos cercanos y comprensibles. El objetivo es facilitar un acceso directo al significado del texto, evitando barreras derivadas de un lenguaje excesivamente complejo o técnico.
2. **Estructura guiada.** Los contenidos deben presentarse de forma ordenada y secuencial, preferentemente mediante pasos claros o listas numeradas. Esta estructura guiada permite que cada acción quede claramente identificada, reduciendo la sobrecarga cognitiva y facilitando el seguimiento de las instrucciones por parte del alumnado.
3. **Coherencia expositiva.** El texto debe organizarse siguiendo una estructura lógica que incluya introducción, desarrollo y cierre. Asimismo, se recomienda resumir ideas complejas y utilizar conectores textuales que ayuden a mantener la coherencia y la continuidad del discurso, facilitando la retención y comprensión de la información.
4. **Tono y motivación.** LEdui promueve el uso de un tono positivo, empático y cercano en la redacción de los contenidos educativos. Esta regla busca reforzar la confianza del alumnado y fomentar su implicación en la tarea, reduciendo el miedo al error y la sensación de frustración ante actividades percibidas como complejas.
5. **Aprendizaje autónomo.** Los textos deben redactarse como si el alumnado aprendiera de forma autónoma, incorporando aclaraciones, ejemplos o metáforas sencillas que puedan sustituir la figura del tutor en el momento inicial de la comprensión. Esta regla pretende favorecer la autorregulación y el aprendizaje a ritmo propio.
6. **Intención pedagógica.** Todo contenido adaptado a LEdui debe indicar de forma explícita los objetivos de la actividad, los criterios de entrega y, cuando sea posible, el tiempo estimado de trabajo. Esta explicitación permite al alumnado comprender qué se espera de él y planificar adecuadamente la realización de la tarea.
7. **Aprendizaje colaborativo.** LEdui fomenta la inclusión de propuestas que favorezcan la cooperación, la ayuda mutua y la evaluación entre iguales. Esta regla busca potenciar las habilidades sociales y comunicativas del alumnado, así como promover dinámicas de aprendizaje compartido.
8. **Pensamiento crítico y reflexión.** El formato incluye la incorporación de preguntas abiertas o dilemas breves que inviten al análisis personal y a la reflexión, sin imponer una respuesta cerrada. Esta regla favorece la profundización en los contenidos y estimula el pensamiento crítico del alumnado.
9. **Conexión profesional y contextual.** Los contenidos deben relacionarse con situaciones reales del entorno cotidiano o profesional del alumnado. Esta conexión contextual facilita la transferencia del aprendizaje y ayuda a comprender la utilidad práctica de los conocimientos trabajados.
10. **Recursos multimedia y multimodales.** LEdui contempla la posibilidad de trabajar y entregar contenidos mediante distintos formatos, como vídeo, audio, infografías o podcasts, adaptando el canal de comunicación a los diferentes estilos de aprendizaje y necesidades del alumnado.

Las diez reglas presentadas constituyen un marco operativo completo para la adaptación de contenidos educativos con criterios de accesibilidad cognitiva e inclusión. Su diseño responde a la necesidad de traducir principios pedagógicos abstractos en especificaciones concretas, aplicables tanto de forma manual por el profesorado como automáticamente mediante sistemas computacionales. La complementariedad de las reglas permite abordar las distintas dimensiones del aprendizaje descritas, así como los pricipios comunes en los marcos analizados. Esta diversidad de enfoques garantiza que el formato LEdui no se limite a una simplificación superficial del lenguaje, sino que proporcione una transformación integral orientada a reducir barreras cognitivas y favorecer el acceso equitativo al aprendizaje. En el Anexo A se incluye una explicación más detallada de cada regla, incluyendo fundamentación pedagógica y ejemplos de aplicación.

4.1.2.1. Dimensiones operativas de las reglas LEdui

Las 10 reglas LEdui pueden analizarse desde tres dimensiones complementarias que permiten comprender su operacionalización y su conexión con los principios teóricos identificados:

- **Dimensión lingüística:** aborda el nivel léxico-sintáctico, la organización secuencial y la coherencia global del discurso.
- **Dimensión pedagógica:** traduce el enfoque inclusivo en estrategias didácticas concretas aplicables a diferentes modalidades de trabajo.
- **Dimensión multimodal y contextual:** vincula los apoyos no verbales con la contextualización real del aprendizaje.

La Tabla 4.1 muestra la conexión de cada regla LEdui con estas dimensiones operativas.

Tabla 4.1: Conexión de las reglas LEdui con dimensiones operativas

Regla	Nombre de la regla	Dimensión principal
1	Claridad lingüística	Lingüística
2	Estructura guiada	Lingüística
3	Coherencia expositiva	Lingüística
4	Tono y motivación	Pedagógica
5	Aprendizaje autónomo	Pedagógica
6	Intención pedagógica	Pedagógica
7	Aprendizaje colaborativo	Pedagógica
8	Pensamiento crítico y reflexión	Pedagógica
9	Conexión profesional y contextual	Multimodal y contextual
10	Recursos multimedia y multimodales	Multimodal y contextual

Esta organización por dimensiones no implica que las reglas deban aplicarse de forma aislada. Como se ha comentado, las reglas LEdui están diseñadas para aplicarse de forma **sinérgica**, de modo que la aplicación de una regla refuerza la efectividad de las demás.

4.1.2.2. Jerarquía de prioridades

Los marcos normativos analizados (DUA, WCAG, Lectura Fácil, UNE 71362:2020 y guías de accesibilidad cognitiva) no establecen jerarquías férreas entre sus pautas, ya que todas ellas son perfectamente complementarias y compatibles entre sí. De igual modo, las 10 reglas LEdui están diseñadas para aplicarse de forma conjunta y sinérgica, reforzándose mutuamente sin generar conflictos.

No obstante, es más fácil teorizar que implementar, es por ello que en la práctica computacional resulta útil establecer una jerarquía de prioridades que oriente la aplicación de las reglas. Esta jerarquización no implica que unas reglas puedan ignorarse en favor de otras, sino que establece un orden de aplicación que garantiza que los aspectos más críticos se aborden en primer lugar.

Es por ello que las reglas LEdui se organizan en tres niveles de prioridad:

1. **Nivel 1 – Reglas nucleares (prioridad crítica):** Reglas 1, 2 y 6. Constituyen el núcleo mínimo del formato LEdui. Su cumplimiento garantiza que el alumnado pueda comprender qué debe hacer, cómo debe hacerlo y en qué orden, reduciendo las barreras básicas de acceso al contenido.
2. **Nivel 2 – Reglas estructurales (prioridad alta):** Reglas 3, 5 y 10. Complementan el núcleo básico incorporando elementos que refuerzan la comprensión global, la autonomía del alumnado y la diversificación de formatos. Su aplicación mejora significativamente la calidad de la adaptación sin modificar el objetivo pedagógico esencial.
3. **Nivel 3 – Reglas de enriquecimiento pedagógico (prioridad media):** Reglas 4, 7, 8 y 9. Amplían el alcance pedagógico del contenido, fomentando aspectos motivacionales, sociales, reflexivos y contextuales. Su aplicación enriquece notablemente la experiencia de aprendizaje, aunque su ausencia temporal no compromete la accesibilidad básica del material.

Un ejemplo ilustrativo de la complementariedad de las reglas puede observarse en la adaptación de contenidos de Formación Profesional que requieren terminología técnica especializada. La Regla 9 (Conexión profesional y contextual) demanda el uso de términos propios del ámbito profesional, mientras que la Regla 1 (Claridad lingüística) exige evitar vocabulario complejo sin explicación. Lejos de entrar en conflicto, ambas reglas se aplican conjuntamente mediante la estrategia de *tecnicismo explicado*: se mantiene el término técnico necesario para la conexión con el contexto real, pero se acompaña de una explicación sencilla que garantiza su comprensión. Por ejemplo:

> *"Realiza el troubleshooting (resolución de problemas paso a paso) de la red siguiendo el modelo OSI."*

De este modo, se preserva la precisión terminológica requerida en el ámbito profesional sin sacrificar la accesibilidad cognitiva del enunciado, cumpliendo simultáneamente ambas reglas.

4.1.3. Validación del formato LEdui

Una vez definidas las 10 reglas que conforman el formato de Lectura Educativa Inclusiva (LEdui), resulta necesario validar su coherencia, completitud y utilidad práctica. Esta validación se aborda desde dos perspectivas complementarias: por un lado, se analiza la capacidad de las reglas para responder a las necesidades educativas especiales más habituales; por otro, se verifica su alineación con los marcos normativos de referencia que han sustentado el diseño conceptual del formato, constituyendo este último un proceso de validación inversa.

4.1.3.1. Conexión con necesidades educativas especiales

La utilidad real de las reglas LEdui depende especialmente de su capacidad para responder de manera efectiva a las necesidades educativas especiales presentes en el aula. Las reglas

solo pueden considerarse adecuadas si contribuyen a reducir barreras reales en el acceso al aprendizaje.

Dado que las reglas LEdui han sido formuladas como principios operativos independientes, su aplicación no requiere necesariamente la activación simultánea de todas ellas. El diseño del formato permite priorizar subconjuntos de reglas en función del tipo de necesidad educativa que se pretenda atender, lo que refuerza su flexibilidad pedagógica y su aplicabilidad práctica.

Con el objetivo de validar la completitud y coherencia pedagógica del conjunto de reglas propuesto, se ha llevado a cabo un análisis de correspondencia entre las reglas LEdui y los tipos de necesidades educativas especiales más habituales en las aulas. Este análisis no pretende establecer relaciones excluyentes ni definitivas, sino identificar qué reglas resultan especialmente relevantes para cada perfil. Estos perfiles han sido identificados de estudios impulsados por organizaciones de referencia como Plena Inclusión [7] y legislaciones de relevancia como las que actualmente están en funcionamiento en las aulas de Castilla-La Mancha [8] que incluyen clasificaciones pedagógicas de alumnado con necesidades educativas especiales.

La Tabla 4.2 presenta una síntesis de esta correspondencia, relacionando cada tipo de necesidad educativa con las reglas LEdui que conviene priorizar y con una breve justificación pedagógica de dicha selección.

Este análisis pone de manifiesto que el conjunto de reglas LEdui cubre de forma transversal un amplio espectro de necesidades educativas especiales, sin depender de adaptaciones específicas para cada perfil. Esta flexibilidad refuerza el carácter inclusivo del formato y valida su idoneidad como base para la adaptación sistemática de contenidos educativos.

4.1.3.2. Validación inversa con marcos normativos

Además de validar la utilidad práctica de las reglas LEdui frente a necesidades educativas especiales, resulta necesario verificar que el nivel de concreción alcanzado en las reglas no introduzca contradicciones con los marcos de referencia originales. LEdui no se concibe como un nuevo estándar normativo independiente, sino como una síntesis operativa de principios ya consolidados. Su validez, por tanto, depende de que las reglas propuestas mantengan una alineación clara con los objetivos, enfoques y criterios fundamentales de los marcos de inclusión educativa de los que ha partido su definición. A continuación se resume la validación desarrollada:

1. **Alineación con el Diseño Universal para el Aprendizaje (DUA).** Las reglas LEdui presentan una correspondencia directa con los tres principios fundamentales del DUA:
    - **Principio de implicación:** Reglas 4 (tono y motivación), 6 (intención pedagógica), 7 (aprendizaje colaborativo) y 8 (pensamiento crítico) favorecen la autorregulación, la persistencia y la motivación del alumnado.
    - **Principio de representación:** Reglas 1 (claridad lingüística), 2 (estructura guiada), 3 (coherencia expositiva), 5 (aprendizaje autónomo) y 10 (recursos multimedia) facilitan el acceso y la comprensión de la información.
    - **Principio de acción y expresión:** Reglas 2 (secuenciación de tareas), 5 (aprendizaje autónomo), 6 (explicitación de objetivos) y 10 (formatos multimodales) apoyan la planificación, la ejecución y la expresión del aprendizaje.

Tabla 4.2: Conexión reglas LEdui y necesidades educativas especiales

Tipo de necesidad educativa	**Reglas LEdui a priorizar**	**Justificación pedagógica**
Necesidades cognitivas y de aprendizaje	Regla 1 Regla 2 Regla 4	La claridad del lenguaje y la estructura secuencial de los contenidos facilitan la comprensión y la organización del aprendizaje. El tono motivador ayuda a mantener la atención y a mejorar la asimilación de la información.
Necesidades de representación y comunicación	Regla 1 Regla 3 Regla 10	La precisión del lenguaje y el uso de recursos visuales y auditivos facilitan la representación de la información y la comunicación de ideas. La variedad de formatos ayuda a los alumnos a expresarse según sus capacidades.
Necesidades instrumentales	Regla 2 Regla 5 Regla 6	La organización clara, las instrucciones autónomas y la explicitación de objetivos permiten a los alumnos acceder a la información y trabajar con mayor independencia, compensando posibles dificultades instrumentales.
Necesidades asociadas a trastornos del desarrollo y la conducta	Regla 1 Regla 4 Regla 6	La claridad lingüística y la explicitación de expectativas reducen la incertidumbre. El tono empático favorece la autorregulación y la participación del alumnado con trastornos del desarrollo y de la conducta.
Necesidades vinculadas a situaciones socioculturales desfavorecidas	Regla 1 Regla 2 Regla 9	Un lenguaje claro, una estructura ordenada y la conexión con contextos reconocibles contribuyen a compensar posibles carencias de estimulación previa y a facilitar la comprensión del contexto escolar.
Altas capacidades	Regla 7 Regla 8 Regla 9	El fomento del trabajo colaborativo, las preguntas de reflexión crítica y la conexión con contextos profesionales permiten ampliar y profundizar en los contenidos, adaptando el nivel de exigencia a las capacidades del alumnado.

De este modo, LEdui traduce los principios del DUA a reglas concretas sin alterar su enfoque global.

2. **Alineación con las Pautas de Accesibilidad para el Contenido Web (WCAG).** La alineación del formato LEdui se produce especialmente con el principio de comprensibilidad de WCAG. Las reglas 1 (lenguaje claro), 2 y 3 (organización coherente), 5 (explicaciones integradas) y 6 (explicitación de expectativas) refuerzan directamente este principio. La regla 10 (recursos multimedia) se alinea con los requisitos de alternativas para contenido multimedia. Respecto al resto de principios WCAG, aunque LEdui no aborda exhaustivamente los aspectos técnicos de accesibilidad web, su especificación no entra en conflicto con ellos.
3. **Alineación con la Lectura Fácil (UNE 153101:2018 EX).** El formato LEdui se alinea con los principios esenciales de la Lectura Fácil: claridad lingüística, simplicidad sintáctica y organización predecible. Las reglas 1 (frases cortas, vocabulario común), 2 (estructuras claras) y 6 (explicitación de expectativas) responden directamente a las recomendaciones de la norma UNE 153101:2018 EX. LEdui traslada estos principios al ámbito educativo, formulándolos como reglas operativas aplicables a enunciados y actividades de aprendizaje.
4. **Alineación con la norma UNE 71362:2020 de calidad de materiales educativos.** Las reglas LEdui se ajustan a los criterios de accesibilidad del contenido textual y audiovisual establecidos en la norma UNE 71362:2020. Las reglas 2 y 3 (estructuración clara), regla 9 (conexión contextual) y regla 10 (apoyos visuales con explicaciones textuales) se corresponden con los criterios de accesibilidad recogidos en dicha norma. El diseño del formato evita la transmisión de información exclusivamente mediante características sensoriales y promueve una presentación comprensible compatible con tecnologías de apoyo.
5. **Alineación con guías de accesibilidad cognitiva** Las reglas LEdui muestran una alineación directa con las guías especializadas de accesibilidad cognitiva, que priorizan la reducción de la carga cognitiva, la previsibilidad de la información y la facilitación de la comprensión real del contenido. Las reglas 2 (estructura guiada), 3 (coherencia expositiva), 4 (tono cercano), 5 (aprendizaje autónomo) y 10 (recursos multimedia) responden explícitamente a las recomendaciones de estas guías, reforzando el carácter cognitivo del enfoque adoptado.

En síntesis, esta validación permite afirmar que las diez reglas que conforman el formato LEdui no contradicen ni fragmentan los marcos normativos de referencia, sino que mantienen una coherencia global con sus principios fundamentales e, incluso, de alguna manera también los unifican. De este modo, LEdui se consolida como una propuesta coherente, operativa y normativamente alineada para la adaptación sistemática de contenidos educativos.

4.2. DEFINICIÓN DE REQUISITOS DEL SISTEMA INTELIGENTE.

La definición de los requisitos del sistema corresponde a la **Fase 2 del plan de trabajo**. Los mismos se han organizado en dos categorías: requisitos necesarios (MVP) y requisitos deseables, concebidas como extensiones iterativas del MVP.

4.2.1. Requisitos del producto mínimo viable (MVP)

Concretamente los siguientes:

RMVP1. El sistema debe permitir la introducción de enunciados educativos en formato textual para su posterior procesamiento.

RMVP2. El sistema debe transformar automáticamente los contenidos introducidos aplicando de forma armonizada y consistente las diez reglas definidas en el estándar LEdui.

RMVP3. El sistema debe generar una versión adaptada del contenido cuya salida sea determinista, garantizando que un mismo enunciado de entrada produzca siempre resultados de calidad similar.

RMVP4. El sistema debe ofrecer el servicio de transformación mediante un servicio web accesible a través de una API REST.

RMVP5. El sistema debe integrar apoyos visuales en forma de pictogramas o imágenes de contexto asociados de manera coherente al contenido textual adaptado.

RMVP6. El sistema debe proporcionar al usuario indicaciones claras sobre qué reglas del estándar LEdui han sido más relevantes en el proceso de transformación, con el objetivo de incentivar la integración progresiva de las reglas en la elaboración de futuros enunciados educativos.

4.2.2. Requisitos deseables

Concretamente los siguientes:

RD1. El sistema debe ofrecer una interfaz web orientada al profesorado que facilite la interacción con el servicio y la visualización de los contenidos adaptados.

RD2. El sistema debe ofrecer una interfaz que cumplas los criterios básicos de accesibilidad, prestando especial atención a la claridad en la presentación de la información.

RD3. El sistema debe permitir la gestión de usuarios y roles para controlar el acceso a las funcionalidades disponibles.

RD4. El sistema debe almacenar el historial de transformaciones realizadas para su consulta y posible reutlización futura.

RD5. El sistema debe permitir descargar el resultado de lsa trasnformaciones en un formato ofimático educativa.

RD6. El sistema debe estar diseñado de forma modular y extensible, permitiendo la incorporación futura de nuevas reglas LEdui o servicios externos.

RD7. El sistema debe incluir una sección de administración que permita consultar el conjunto de transformaciones realizadas y gestionar los usuarios y sus permisos de acceso.

RD8. El sistema debe permitir la personalización del nivel educativo del enunciado de entrada, con el fin de generar adaptaciones más ajustadas al perfil del alumnado destinatario.

RD9. El sistema debe ser persistente, apoyándose en una base de datos para almacenar la información necesaria para su correcto funcionamiento.

4.3. ARQUITECTURA Y DISEÑO TÉCNICO DEL SISTEMA

Esta sección presenta la arquitectura del sistema mediante diagramas que ilustran su estructura, modelo de datos, flujos de interacción y las decisiones técnicas fundamentales tomadas durante el diseño. El objetivo es ofrecer una perspectiva general del sistema que contextualizará los detalles de implementación en secciones posteriores.

El sistema se basa en un modelo cliente–servidor, en el que un servicio web actúa como núcleo del procesamiento y expone una serie de endpoints accesibles mediante una API REST. Esta estructura técnica es el resultado del desarrollo inicial llevado a cabo durante la **Fase 3. Desarrollo del producto mínimo viable (MVP)**, donde se implementaron las funcionalidades esenciales del servicio de adaptación, y de su evolución posterior en la **Fase 4. Desarrollo iterativo y ampliación de funcionalidades**, donde se incorporaron y refinaron nuevos componentes del sistema.

El sistema desarrollado sigue una arquitectura en capas que separa claramente las responsabilidades de presentación, lógica de negocio y persistencia de datos, facilitando su mantenimiento, escalabilidad y evolución futura. Esta separación arquitectónica constituye uno de los pilares fundamentales del diseño, permitiendo modificar o reemplazar componentes individuales sin afectar al resto del sistema.

La Figura 4.1 ilustra la arquitectura en tres capas del sistema LEdui. En la **capa de presentación**, implementada mediante una interfaz web desarrollada con HTML, CSS y JavaScript, el usuario interactúa con el sistema a través de formularios de entrada de texto, visualización de resultados y gestión de historial. Esta capa se comunica exclusivamente con la capa intermedia mediante peticiones HTTP estandarizadas.

La **capa de aplicación**, constituida por una API REST desarrollada con FastAPI, actúa como núcleo del sistema y se encarga de orquestar todas las operaciones. Esta capa gestiona la autenticación mediante Google OAuth 2.0 y tokens JWT, valida las peticiones entrantes, coordina las llamadas a los servicios externos de inteligencia artificial (Claude API para transformación de texto y Gemini API para generación de imágenes), consulta el catálogo de pictogramas de ARASAAC, y persiste toda la información en la base de datos. La separación de esta capa intermedia permite que la lógica de adaptación pedagógica esté completamente desacoplada de la presentación, facilitando su reutilización en futuros clientes (aplicaciones móviles, integraciones con LMS, etc.).

Finalmente, la **capa de persistencia**, implementada mediante PostgreSQL, almacena de forma estructurada la información de usuarios, transformaciones realizadas (incluyendo enunciados originales, adaptados y explicaciones pedagógicas), pictogramas asociados e imágenes de contexto generadas. Esta arquitectura de tres capas garantiza la separación de responsabilidades, mejora la mantenibilidad del código y facilita la escalabilidad horizontal del sistema mediante la replicación de instancias de la capa de lógica de negocio.

En las subsecciones a continuación se detallan cada una de estas capas.

4.3.1. Capa de presentación (*frontend*)

Implementada mediante tecnologías web estándar (HTML5, CSS3 y JavaScript), esta capa proporciona la interfaz de usuario a través de la cual los docentes interactúan con el sistema. Los componentes principales incluyen:

- **Formulario de entrada:** Permite al usuario introducir el enunciado original, seleccionar el nivel educativo y especificar preferencias de transformación.

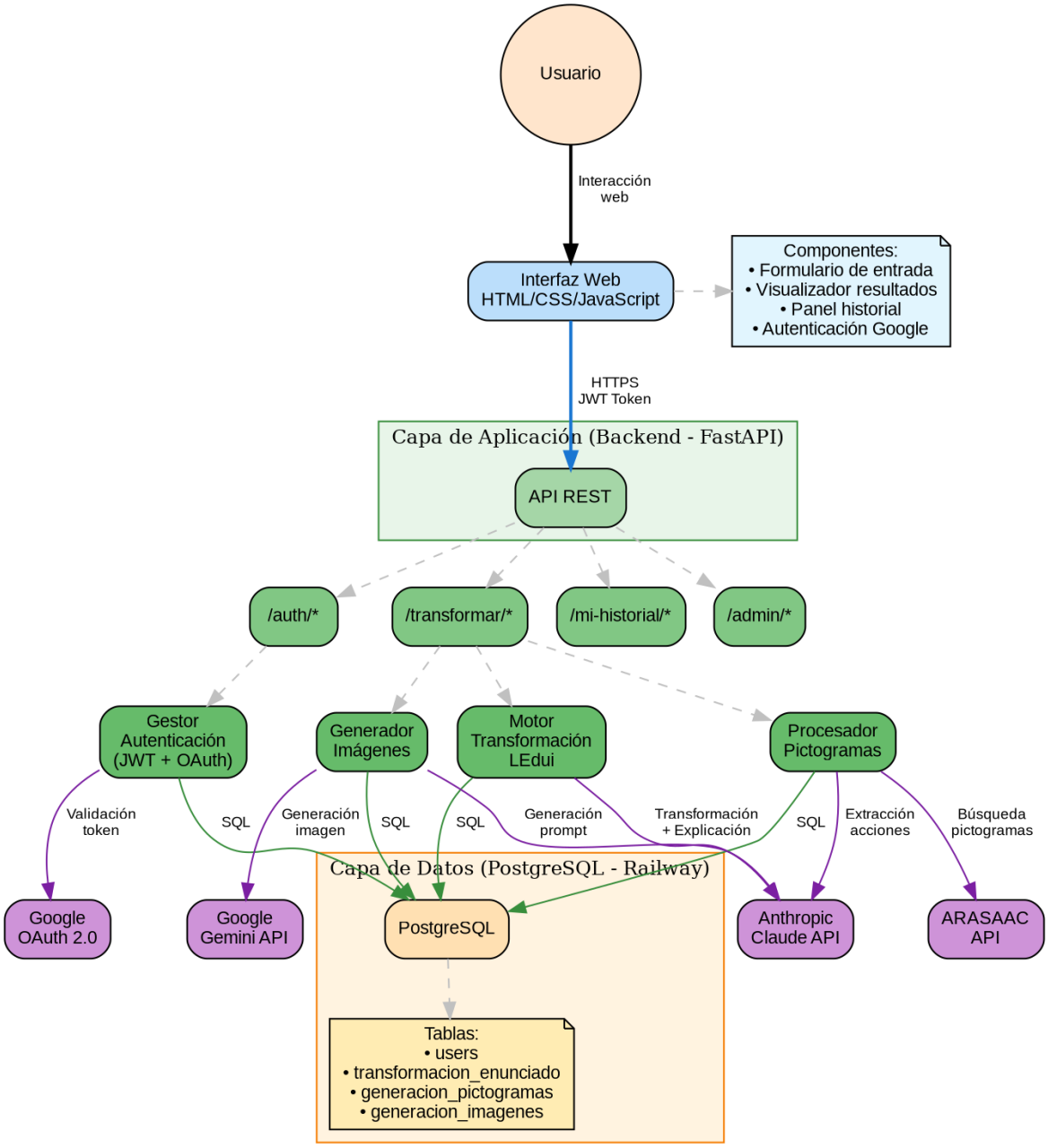


Figura 4.1: Arquitectura general del sistema LEdui

- **Visualizador de resultados:** Presenta de forma clara y estructurada el enunciado adaptado, los pictogramas generados, las imágenes de contexto y las explicaciones pedagógicas.
- **Panel de historial:** Proporciona acceso al registro completo de transformaciones previas del usuario, permitiendo consultar, descargar y reutilizar contenidos generados anteriormente.
- **Sistema de autenticación:** Integra el flujo de Google OAuth 2.0, gestionando el proceso de login y la persistencia del token JWT en el navegador.

Esta capa se comunica con el *backend* exclusivamente mediante peticiones HTTP/HTTPS a la API REST, sin mantener estado ni lógica de negocio, actuando como cliente ligero del sistema.

4.3.2. Capa de aplicación (*backend*)

Constituye el núcleo del sistema, implementando toda la lógica de negocio y coordinando las interacciones entre componentes. Desarrollada con FastAPI, expone un conjunto de endpoints REST organizados por funcionalidad:

- `/auth/*`: Gestión de autenticación y autorización de usuarios.
- `/transformar/*`: Endpoints de transformación de contenidos educativos.
- `/mi-historial/*`: Consulta y gestión del historial personal del usuario.
- `/admin/*`: Funcionalidades avanzadas reservadas a usuarios administradores.

Internamente, el *backend* se estructura en módulos especializados:

- **Gestor de autenticación:** Implementa el flujo OAuth 2.0 con Google y la generación/validación de tokens JWT para autenticación de sesiones.
- **Motor de transformación LEdui:** Orquesta el proceso completo de transformación de enunciados, gestionando las llamadas a la API de Claude con los prompts especializados y procesando las respuestas.
- **Procesador de pictogramas:** Analiza los enunciados adaptados para identificar acciones clave, consulta la API de ARASAAC, descarga las imágenes y las compone en un formato unificado.
- **Generador de imágenes:** Coordina la generación de imágenes educativas mediante Google Gemini, desde el análisis del contenido hasta el almacenamiento del resultado.

4.3.2.1. Servicios externos del *backend*

A su vez, los mismos se apoyan en cuatro servicios externos fundamentales:

- **Anthropic Claude API:** motor de inteligencia artificial responsable de la transformación de enunciados, generación de explicaciones pedagógicas y análisis de contenidos. Modelo utilizado: Claude 3.7 Sonnet.
- **Google Gemini API:** servicio de generación de imágenes mediante IA, utilizado para crear visualizaciones educativas contextualizadas. Modelo utilizado: Gemini 2.5 Flash (Nano Banana).
- **ARASAAC API:** repositorio público de pictogramas accesibles, proporcionando más de 10.000 pictogramas en español para representar acciones y conceptos.

- **Google OAuth 2.0:** sistema de autenticación delegada que permite a los usuarios identificarse mediante sus cuentas de Google sin necesidad de gestionar credenciales adicionales.

Todas las comunicaciones entre capas y con servicios externos se realizan mediante protocolos seguros (HTTPS), garantizando la confidencialidad e integridad de los datos transmitidos. La arquitectura implementa adicionalmente mecanismos de manejo de errores y reintentos para garantizar la robustez del sistema ante fallos temporales de servicios externos.

4.3.2.2. Estructura de módulos del *backend*

El *backend* del sistema LEdui sigue una arquitectura modular que separa claramente las responsabilidades funcionales en distintos módulos Python. La Figura 4.2 ilustra la organización de estos módulos y sus relaciones de dependencia.

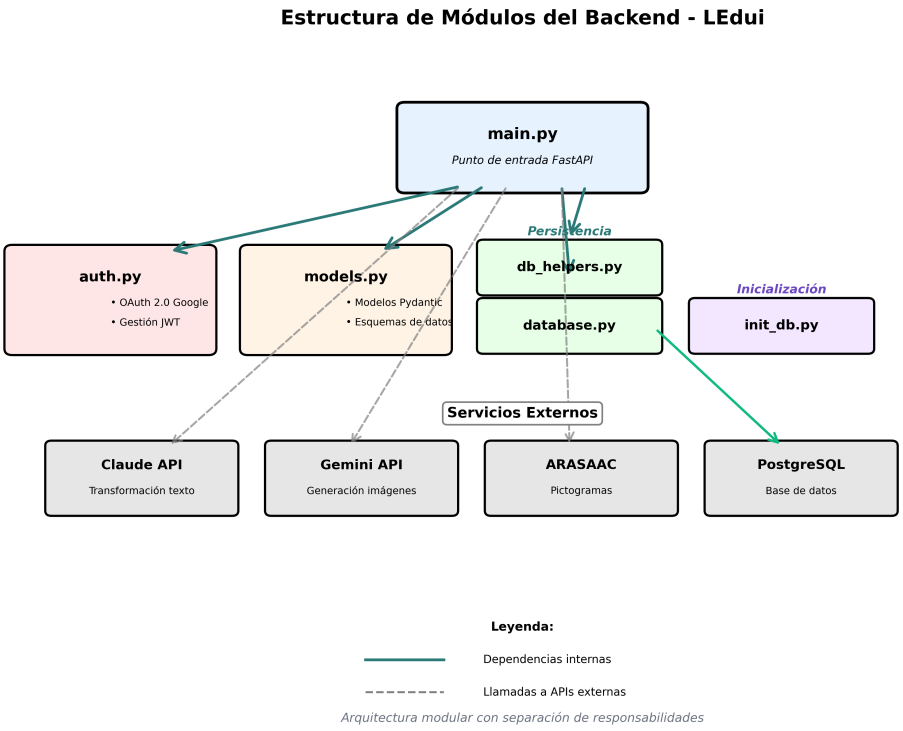


Figura 4.2: Estructura de módulos del *backend*

El módulo `main.py` actúa como punto de entrada de la aplicación FastAPI, orquestando las peticiones HTTP entrantes y coordinando las llamadas a los demás módulos del sistema. Este módulo define los *endpoints* de la API REST y gestiona el ciclo de vida de las peticiones.

La **parte de autenticación** se implementa en el módulo `auth.py`, que encapsula toda la lógica relacionada con la autenticación mediante OAuth 2.0 de Google y la gestión de tokens JWT. Este módulo proporciona funciones para validar credenciales, generar y verificar tokens, y obtener información de usuario autenticado.

El módulo `models.py` contiene los **modelos de datos** expuestos en la capa de persistencia implementados mediante Pydantic, que definen los esquemas de validación para las peticiones y respuestas de la API. Estos modelos garantizan la integridad de los datos

intercambiados entre el *frontend* y el *backend*, realizando validaciones automáticas de tipos y formatos.

Para apoyar las operaciones de la **capa de persistencia** se dispone dos módulos complementarios: `database.py`, que configura la conexión con PostgreSQL mediante SQLAlchemy y define los modelos ORM de las tablas de la base de datos; y `db_helpers.py`, que implementa funciones auxiliares para operaciones CRUD (crear, leer, actualizar, eliminar) sobre las entidades del sistema. Esta separación facilita la reutilización de operaciones comunes y simplifica el mantenimiento del código de acceso a datos.

Además, el módulo `init_db.py` es una utilidad de inicialización que permite crear las tablas de la base de datos y poblarlas con datos iniciales si es necesario. Este módulo se ejecuta típicamente durante el despliegue inicial del sistema o en entornos de desarrollo.

Desde el punto de vista de las dependencias externas, el módulo `main.py` realiza llamadas directas a las APIs de Claude (para transformación de texto), Gemini (para generación de imágenes de contexto) y ARASAAC (para obtención de pictogramas), mientras que la capa de persistencia interactúa con PostgreSQL a través de la configuración establecida en `database.py`.

Esta estructura modular facilita el mantenimiento evolutivo del sistema, permite la realización de pruebas unitarias sobre módulos aislados y favorece la escalabilidad al separar claramente las responsabilidades de cada componente.

4.3.3. Capa de persistencia

Implementa la persistencia de toda la información del sistema mediante una base de datos relacional PostgreSQL. Esta capa garantiza:

- Almacenamiento persistente de usuarios, transformaciones y recursos multimedia.
- Integridad referencial mediante claves foráneas y restricciones.
- Trazabilidad completa mediante timestamps de creación y modificación.
- Recuperación eficiente del historial de transformaciones mediante índices optimizados.

La base de datos PostgreSQL implementa un esquema relacional normalizado siguiendo las formas normales 3FN (Tercera Forma Normal), garantizando la integridad referencial, minimizando redundancias y optimizando las operaciones de consulta y actualización.

El **modelo de datos**, representado en la Figura 4.3, se organiza en torno a cuatro tablas principales relacionadas mediante claves externas que garantizan la consistencia del sistema. A continuación se detallan cada una de estas tablas.

4.3.3.1. Tabla users

Almacena la información de los usuarios registrados en el sistema. Cada usuario se identifica de forma única mediante dos claves:

- `id`: Clave primaria autogenerada, utilizada internamente para establecer relaciones con otras tablas.
- `google_id`: Identificador único proporcionado por Google OAuth, garantizando que no se creen usuarios duplicados.

Los campos `name` y `picture` son opcionales (nullable), permitiendo casos en los que Google no proporcione esta información o el usuario no autorice su compartición. El campo

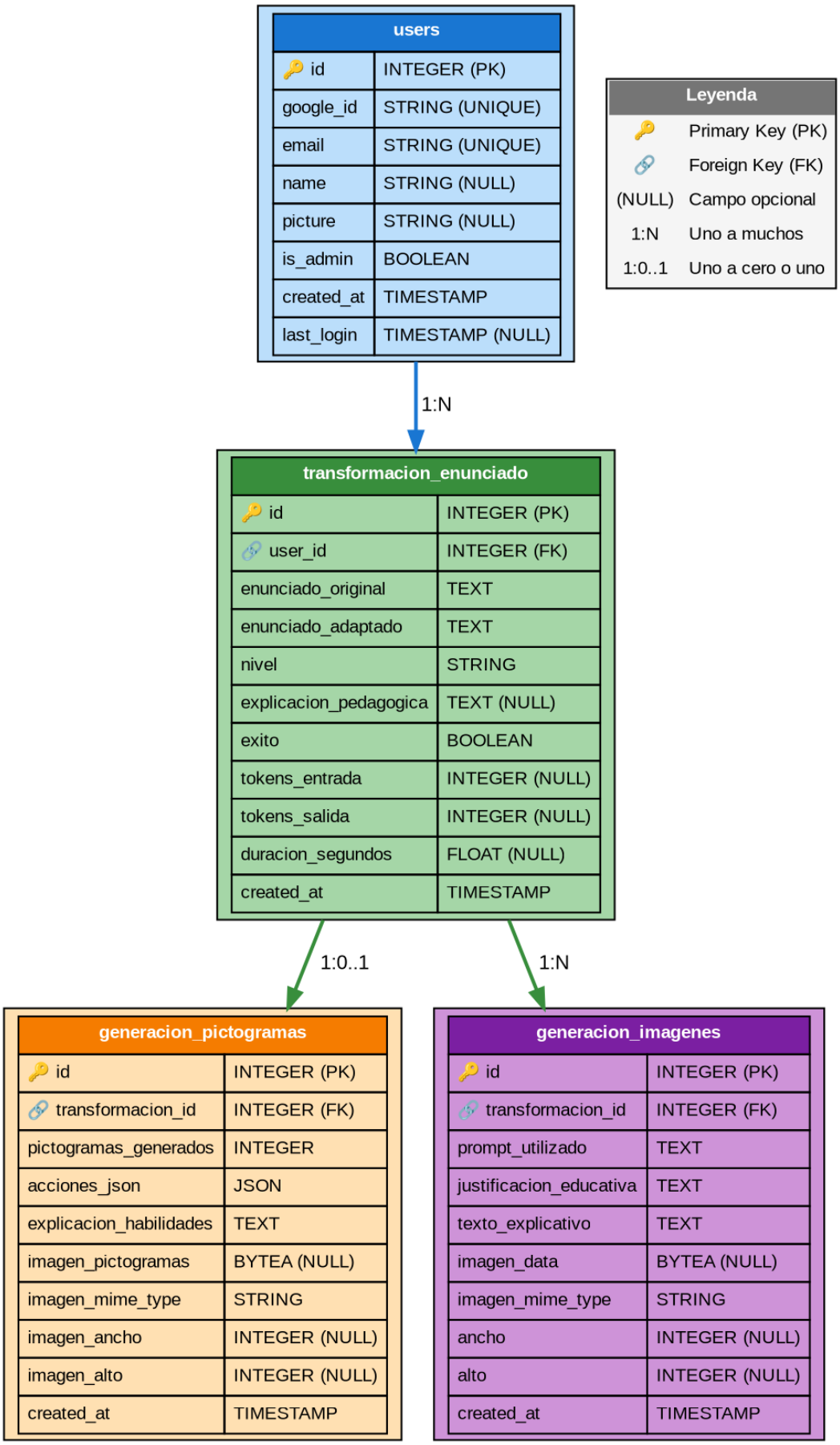


Figura 4.3: Diagrama entidad-relación de la base de datos

booleano `is_admin` permite distinguir entre usuarios estándar y administradores, habilitando el control de acceso granular a funcionalidades privilegiadas del sistema.

El campo `last_login` registra la fecha y hora del último acceso del usuario, información útil para análisis de uso y detección de cuentas inactivas.

4.3.3.2. Tabla transformacion_enunciado

Constituye la tabla central del modelo de datos, almacenando cada transformación de enunciado realizada en el sistema. Sus campos principales incluyen:

- `enunciado_original` y `enunciado_adaptado`: almacenan los textos completos, permitiendo comparaciones y análisis posteriores.
- `nivel`: categoriza la transformación según el nivel educativo (secundaria, formación_profesional, universidad), facilitando consultas filtradas.
- `explicacion_pedagogica`: la explicación de las reglas LEdui aplicadas, dirigida al profesorado.
- `tokens_entrada`, `tokens_salida` y `duracion_segundos`: metadatos de uso que permiten analizar el consumo de recursos y optimizar el rendimiento del sistema.
- `exito`: campo booleano que indica si la transformación se completó correctamente, útil para estadísticas y detección de errores.

La relación con `users` (clave foránea `user_id`) asocia cada transformación a su autor, garantizando el aislamiento de datos entre usuarios y permitiendo implementar sistemas de cuotas y control de uso personalizado.

En este punto, cabe comentar decisión de diseño destacable en esta tabla es el almacenamiento de las explicaciones pedagógicas dentro de la tabla `transformacion_enunciado` en lugar de en una tabla independiente. Esta decisión de diseño se ha fundamentado en los siguientes criterios:

1. **Dependencia existencial:** las explicaciones pedagógicas no tienen sentido de forma independiente, sino que constituyen metadatos descriptivos del proceso de transformación del enunciado educativo. Además, cada transformación genera obligatoriamente una única explicación, estableciendo una relación 1:1 que no justifica la creación de una entidad separada.
2. **Integridad semántica:** mantener enunciado original, enunciado adaptado y explicación pedagógica en un único registro refuerza la cohesión conceptual de la transformación como unidad funcional indivisible, eliminando la posibilidad de inconsistencias (transformaciones sin explicación o explicaciones huérfanas) y potenciando el enfoque pedagógico del sistema. Un enfoque que busca no sólo ofrecer un servicio al usuario sino explicarle en detalle como se ha realizado para que lo interiorice en su propio proceso creativo.
3. **Eficiencia de consultas:** evitar operaciones de *JOIN* sistemáticamente necesarias si las explicaciones residieran en una tabla separada, dado que el 100 % de las consultas sobre transformaciones han de requirir acceso a las explicaciones asociadas al considerarse una información clave a transmitir al usuario.

Esta decisión prioriza la cohesión conceptual y la eficiencia operativa sobre una normalización que, en este contexto específico, no aportaría beneficios tangibles.

4.3.3.3. Tabla generacion_pictogramas

Gestiona los conjuntos de pictogramas asociados a cada transformación. La relación con `transformacion_enunciado` es de tipo 1:0..1, ya que una transformación puede existir con o sin pictogramas asociados.

Campos destacables:

- `acciones_json`: Almacena en formato JSON la estructura completa de las acciones identificadas, incluyendo la acción educativa original, la acción simplificada utilizada para la búsqueda y la explicación pedagógica asociada. Este diseño permite flexibilidad en la estructura de datos sin modificar el esquema de la base de datos.
- `imagen_pictogramas`: Campo de tipo BYTEA (binary data) que almacena la imagen compuesta final en formato PNG. El almacenamiento de imágenes directamente en la base de datos simplifica la gestión y garantiza la atomicidad de las transacciones.
- `imagen_ancho` e `imagen_alto`: Metadatos que permiten conocer las dimensiones de la imagen sin necesidad de decodificarla, útil para optimizar la renderización en el *frontend*.

4.3.3.4. Tabla generacion_imagenes

Almacena las imágenes de contexto educativas generadas mediante Google Gemini. A diferencia de los pictogramas, una transformación puede asociarse con múltiples imágenes, por lo que la relación con `transformacion_enunciado` es de tipo 1:N.

Campos relevantes:

- `prompt_utilizado`: preserva el prompt exacto enviado a Gemini, permitiendo auditoría, análisis de calidad y refinamiento iterativo de los prompts.
- `justificacion_educativa`: texto generado por Claude que explica por qué esta imagen concreta es pedagógicamente valiosa para el enunciado, dirigido al profesorado.
- `texto_explicativo`: versión simplificada de la justificación, redactada en lenguaje accesible para el alumnado, explicando cómo la imagen les ayudará a comprender la actividad.
- `imagen_data`: almacena la imagen en formato binario, análogamente a los pictogramas.

4.4. FLUJO DE PROCESAMIENTO DE UNA TRANSFORMACIÓN COMPLETA

El proceso de transformación de un enunciado educativo al formato LEdui implica múltiples interacciones secuenciales y paralelas entre componentes del sistema y servicios externos. Esta subsección analiza en detalle el flujo completo de operaciones, desde la autenticación del usuario hasta la presentación de los resultados finales.

Las Figuras 4.4 y 4.5 ilustran mediante un diagrama de secuencia UML el flujo completo de interacciones que se producen cuando un usuario accede al sistema y solicita la transformación completa de un enunciado educativo. El proceso se estructura en **seis etapas** claramente diferenciadas, ejecutadas de forma secuencial para garantizar la consistencia de los datos y la trazabilidad del proceso.

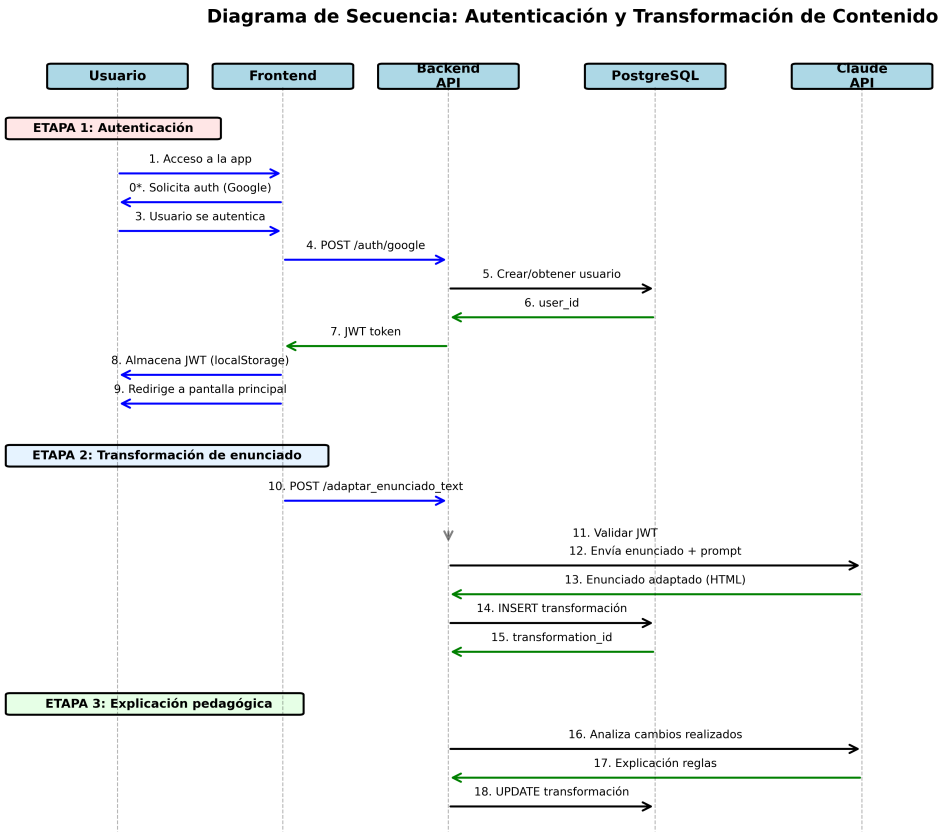


Figura 4.4: Diagrama de secuencia: Autenticación y transformación de contenido (Etapas 1-3)

4.4.1. Etapa 1. Autenticación del usuario

El acceso al sistema requiere autenticación previa mediante Google OAuth 2.0. Este es el primer paso obligatorio antes que el usuario pueda utilizar cualquier funcionalidad y se detalla visualmente en la Figura 4.4:

1. El usuario accede a la aplicación web.
2. El *frontend* detecta que no existe un token JWT válido almacenado en el navegador y redirige automáticamente al usuario al flujo de autorización de Google.
3. El usuario introduce sus credenciales de Google y autoriza a la aplicación a acceder a su información básica de perfil (email, nombre, foto).

4. Google valida las credenciales y devuelve un token de acceso temporal al *frontend*.
5. El *frontend* envía este token al *backend* mediante el endpoint `POST /auth/google`.
6. El *backend* valida el token con los servidores de Google, verifica su autenticidad y extrae la información del usuario.
7. El *backend* consulta la base de datos para determinar si el usuario ya existe (`google_id` único). Si no existe, lo crea; si existe, actualiza el campo `last_login`.
8. El *backend* genera un token JWT firmado que incluye el `user_id` y lo devuelve al *frontend* junto con los datos del usuario.
9. El *frontend* almacena el token JWT en `localStorage` y redirige al usuario a la pantalla principal de transformaciones.

Este diseño garantiza que el *backend* nunca gestiona credenciales de usuarios (contraseñas), delegando esta responsabilidad en Google y reduciendo significativamente la superficie de ataque del sistema. El token JWT tiene una validez de 24 horas, tras las cuales el usuario deberá autenticarse nuevamente.

4.4.2. Etapa 2. Transformación del enunciado al formato LEdui

Una vez autenticado, el usuario puede acceder a la funcionalidad de transformación. El proceso de adaptación del enunciado se desarrolla de la siguiente manera:

10. El usuario introduce un enunciado original, selecciona el nivel educativo y solicita su transformación.
11. El *backend* valida el token JWT, extrae el `user_id` y verifica que no ha expirado.
12. El *frontend* envía el enunciado original, el nivel educativo seleccionado y el límite de palabras al endpoint `POST /adaptar_enunciado_ledu`, incluyendo el token JWT en el header de autenticación. El *backend* construye una petición a la API de Anthropic Claude, incluyendo:
    - Un system prompt detallado que especifica las 10 reglas del formato LEdui, la estructura HTML esperada y el tono pedagógico deseado.
    - El enunciado original proporcionado por el usuario.
    - Parámetros de generación (temperatura, max_tokens, etc.).
13. Claude procesa la petición, aplicando las reglas especificadas, y devuelve el enunciado adaptado en formato HTML junto con metadatos sobre el procesamiento (tokens de entrada consumidos, tokens de salida generados).
14. El *backend* extrae el contenido HTML generado, verifica su estructura y lo almacena en la base de datos creando un nuevo registro en la tabla `transformacion_enunciado`.
15. PostgreSQL devuelve el `transformacion_id` autogenerado, que servirá como referencia para todas las operaciones posteriores de esta transformación.

El tiempo medio de esta etapa oscila entre 3 y 5 segundos, dependiendo de la longitud del enunciado original y de la carga de los servidores de Anthropic.

4.4.3. Etapa 3. Generación de la explicación pedagógica

Paralelamente (o inmediatamente después de la transformación), el sistema genera una explicación dirigida al profesorado:

16. El *backend* solicita a Claude que analice el enunciado adaptado e identifique las reglas LEdui más relevantes aplicadas.
17. Claude genera un texto estructurado que señala las carencias del enunciado original, identifica las reglas aplicadas con ejemplos concretos y ofrece orientaciones para la mejora de futuros contenidos.
18. El *backend* almacena esta explicación como actualización del registro de transformación existente (`UPDATE` en campo `explicacion_pedagogica`).

Esta explicación tiene una doble función pedagógica: informar al profesorado sobre el trabajo realizado por la IA y, más importante aún, educar al docente para que pueda crear contenidos LEdui de forma autónoma en el futuro.

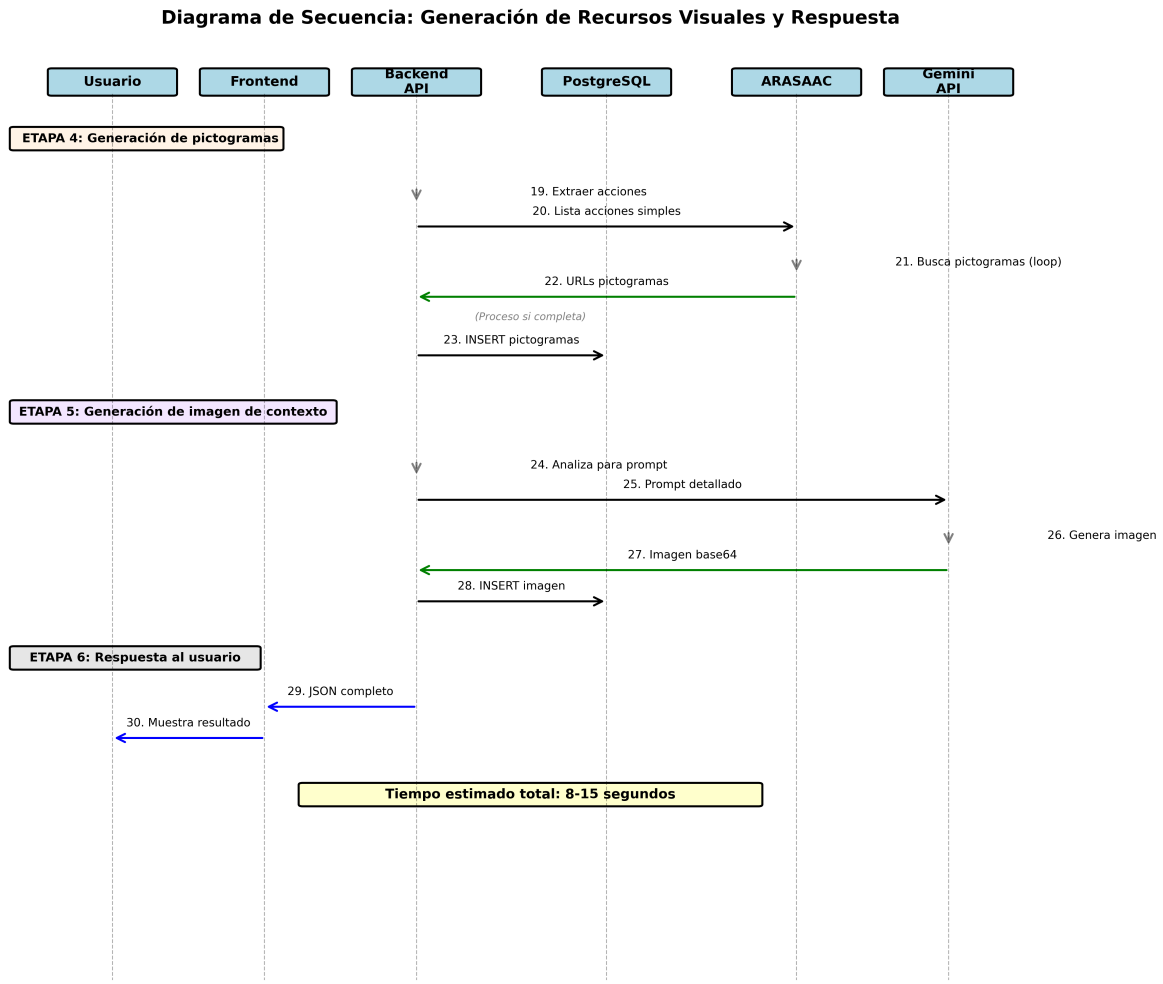


Figura 4.5: Diagrama de secuencia: Generación de recursos visuales y respuesta (Etapas 4-6)

4.4.4. Etapa 4. Generación y composición de pictogramas

El sistema procede a identificar las competencias clave del enunciado y representarlas visualmente:

19. El *backend* solicita a Claude que extraiga las acciones y competencias principales del enunciado adaptado.

20. Claude analiza el contenido, identifica las acciones educativas relevantes y las simplifica a verbos básicos comprensibles por ARASAAC (por ejemplo, "Investigar y seleccionar componentes" se simplifica a "buscar"). El *backend* recibe la lista de acciones simplificadas y, para cada una, consulta la API de ARASAAC mediante peticiones HTTP.
21. ARASAAC devuelve las URLs de los pictogramas más relevantes para cada término de búsqueda.
22. El *backend* descarga todas las imágenes de pictogramas, las redimensiona a un tamaño uniforme (300x300 píxeles) y las compone horizontalmente en una imagen única, añadiendo separadores visuales entre ellas.
23. La imagen compuesta, junto con las explicaciones textuales de cada habilidad, se almacena en la tabla `generacion_pictogramas`.

4.4.5. Etapa 5. Generación de imagen de contexto educativa

Finalmente, el sistema crea una imagen educativa que complementa el enunciado adaptado:

24. El *backend* solicita a Claude que analice el enunciado y genere un prompt detallado para la creación de una imagen educativa.
25. Claude genera un prompt en inglés que describe la imagen deseada, especificando estilo educativo, elementos a incluir, perspectiva y contexto pedagógico. Adicionalmente, proporciona una justificación educativa de por qué esta imagen concreta es valiosa.
26. El *backend* envía el prompt a la API de Google Gemini, especificando formato de salida (16:9, alta resolución).
27. Gemini procesa el prompt y devuelve la imagen generada en formato base64.
28. El *backend* decodifica la imagen, la almacena junto con el prompt, la justificación educativa y el texto explicativo en la tabla `generacion_imagenes`.

4.4.6. Etapa 6. Respuesta integral al usuario

Una vez completadas todas las etapas de generación, el *backend* consolida la información:

29. El *backend* construye un objeto JSON que incluye el `transformacion_id`, el enunciado adaptado, la explicación pedagógica, información sobre los pictogramas generados (número, lista de acciones, disponibilidad de imagen) e información sobre las imágenes de contexto (prompts, justificaciones, disponibilidad de datos). Este objeto JSON se devuelve al *frontend* con código HTTP 200.
30. El *frontend* procesa el JSON, actualiza la interfaz de usuario y presenta los resultados de forma estructurada: enunciado adaptado en la parte principal, pictogramas en un panel lateral, imágenes de contexto debajo del enunciado y explicación pedagógica en una sección colapsable.

El tiempo total de procesamiento de una transformación completa es inferior a 30 segundos, dependiendo en gran medida de la longitud del enunciado original y el número de pictogramas a generar y los tiempos de respuesta de las APIs externas. Este tiempo se considera aceptable para un contexto educativo no crítico, donde la calidad del resultado prima sobre la inmediatez de la respuesta.

4.5. IMPLEMENTACIÓN DEL *BACKEND*

El *backend* del sistema ha sido implementado utilizando FastAPI, lo que permite definir de forma clara los distintos endpoints, gestionar la validación de datos y documentar automáticamente la API. El código se organiza en torno a bloques funcionales bien definidos, cada uno asociado a un conjunto de endpoints con responsabilidades específicas, facilitando su extensión progresiva a lo largo del desarrollo iterativo.

Esta sección describe la implementación de los distintos conjuntos de endpoints, sus parámetros de comunicación y las integraciones con servicios externos que soportan la funcionalidad del sistema, así como la justificación de algunas decisiones de diseño tomadas. De forma más detallada, se ha incluido una descripción de la API completa como Anexo D.

4.5.1. Endpoint de adaptación de contenidos a LEdui

El endpoint de adaptación de contenidos constituye el núcleo funcional del sistema. A través del mismo se reciben los textos educativos originales y se devuelve una versión adaptada conforme a las reglas del formato Lectura Educativa Inclusiva (LEdui).

Tabla 4.3: Endpoint de adaptación de contenidos a LEdui

POST	`/adaptar_enunciado_ledu`	Adapta un enunciado educativo al formato `LEdui`.

Entre las funciones principales de este endpoint se incluyen:

- Recepción de contenidos educativos en formato textual.
- Análisis lingüístico y semántico del texto.
- Aplicación de las reglas LEdui para la adaptación del contenido.
- Alineación de las transformaciones según nivel educativo (secundaria, formación profesional o universidad).
- Generación de una versión estructurada, clara y pedagógicamente orientada del texto.
- Generación de explicación pedagógica de las reglas LEdui que se han aplicado especialmente para llevar a cabo la transformación.

Este endpoint integra servicios de inteligencia artificial accesibles mediante APIs externas, lo que permite realizar transformaciones complejas de manera automática y escalable.

4.5.1.1. Modelo de respuesta para la transformación de enunciados

El sistema implementa un **modelo de respuesta estructuralmente determinista** para la adaptación de enunciados. Esto significa que, aunque el contenido textual específico generado puede variar ligeramente entre ejecuciones debido a la naturaleza estocástica de los modelos de lenguaje (configurados con temperatura por defecto para favorecer la diversidad y naturalidad del lenguaje), la **estructura formal de la respuesta se mantiene invariable**. Cada transformación genera siempre:

- Un enunciado adaptado en formato HTML siguiendo una plantilla fija con secciones predefinidas (*¿Qué vas a aprender?*, *Pasos a seguir*, *¿Qué tienes que entregar?*, *Para reflexionar*, *Tiempo estimado*)
- Una explicación pedagógica que identifica las 2-4 reglas LEdui de mayor aplicación en la transformación concreta

- Un conjunto de pictogramas asociados a acciones clave del enunciado
- Una imagen de contexto relacionada con el contenido educativo

Este **determinismo estructural** garantiza la consistencia pedagógica del formato LEdui sin sacrificar la riqueza expresiva y adaptabilidad del contenido generado. La variabilidad controlada en el contenido textual específico resulta incluso beneficiosa, ya que permite al profesorado obtener múltiples formulaciones alternativas de un mismo enunciado educativo, facilitando la elección de aquella que mejor se ajuste al contexto específico de su aula.

4.5.1.2. Evaluación y selección de modelos de transformación de enunciados educativos

La adaptación de enunciados educativos y el análisis pedagógico se apoyan en la integración del modelo Claude 3.7 Sonnet mediante API. Este modelo se consume como servicio remoto a través de llamadas HTTP desde el *backend*, encargándose de dos funciones principales:

1. **Transformación de enunciados:** el modelo ejecuta las instrucciones proporcionadas por el sistema, que especifican de forma precisa cómo deben aplicarse las reglas del estándar LEdui y qué estructura debe adoptar la salida generada. El comportamiento del modelo se controla mediante instrucciones de sistema detalladas, reglas pedagógicas explícitas y plantillas de salida cerradas, garantizando que el resultado generado sea estructuralmente fijo y determinista.
2. **Análisis pedagógico:** además de la adaptación, el sistema utiliza el mismo servicio para generar textos explicativos dirigidos al docente usuario del aplicativo. Estos análisis identifican las reglas LEdui aplicadas, señalan las carencias del enunciado original y ofrecen orientaciones prácticas para la creación de futuros contenidos educativos. Este uso complementario refuerza el carácter formativo de la solución, trasladando parte del razonamiento pedagógico hacia el usuario final.

Se evaluaron cuatro modelos de lenguaje de gran tamaño (LLMs) como posibles opciones para **la transformación textual de enunciados educativos siguiendo las reglas LEdui.**

La Tabla 4.4 presenta una comparativa técnica exhaustiva de estos modelos, basada en pruebas empíricas realizadas durante el desarrollo del proyecto con enunciados representativos de distintos niveles educativos.

Claude 3.7 Sonnet (Anthropic) [MODELO SELECCIONADO] Ventajas:

- Capacidad excepcional para seguir instrucciones estructuradas complejas, generando outputs con el formato HTML exacto especificado en el 95 % de los casos sin necesidad de post-procesamiento.
- Tono pedagógico natural, claro y motivador sin requerir ajustes complejos en el prompt.
- Equilibrio óptimo entre calidad, velocidad y coste para el caso de uso específico de transformación educativa.
- API REST estable con SDK oficial para Python, facilitando la integración y el mantenimiento.
- Documentación exhaustiva y comunidad activa de desarrolladores.

Desventajas:

- Modelo con tiempo en funcionamiento, posibilidad de cierre por parte del proveedor.

Criterio	Claude 3.7 Sonnet	GPT-4o (OpenAI)	Gemini 1.5 Pro (Google)	Llama 3.1 70B (Meta)
Velocidad respuesta (enunciado 200 pal.)	3-5 seg	4-7 seg	5-8 seg	8-12 seg*
Coste por transformación	0.015 EUR	0.025 EUR (+67 %)	0.010 EUR (-33 %)	0 EUR**
Calidad adaptación LEdui (1-5)	Excelente 5/5	Muy buena 4/5	Buena 3/5	Aceptable 3/5
Adherencia estructura HTML fija	95 %	85 %	75 %	60 %
Tono pedagógico y claridad	Excelente (natural)	Muy bueno (formal)	Bueno (variable)	Aceptable (mejora)
Facilidad integración API	Si - Estable SDK oficial	Si - Estable SDK oficial	Si - Limitada región	No - Compleja infraest.
Soporte multilingüe (español)	Excelente	Excelente	Muy bueno	Bueno

Tabla 4.4: Comparativa técnica de modelos de lenguaje evaluados.

GPT-4o (OpenAI) Ventajas:

- Gran capacidad de razonamiento y comprensión de contexto educativo complejo.
- Documentación muy extensa y amplia comunidad de usuarios.
- Versatilidad para múltiples tareas más allá de transformación de texto.

Desventajas:

- Coste 67 % superior a Claude sin ofrecer ventajas sustanciales para este caso de uso específico.
- Tendencia a generar textos ocasionalmente excesivamente formales, requiriendo ajustes en el prompt.
- Adherencia a estructuras HTML fijas ligeramente inferior.

Gemini 1.5 Pro (Google) Ventajas:

- Coste más bajo entre las opciones comerciales (33 % inferior a Claude).
- Integración nativa con otros servicios de Google Cloud.
- Multimodalidad nativa (procesa texto e imágenes simultáneamente).

Desventajas:

- Mayor variabilidad en la calidad de los outputs, especialmente en la generación de estructuras HTML complejas (75 % de adherencia).
- Sensibilidad alta a pequeños cambios en el prompt, requiriendo mayor ingeniería de prompts.
- Limitaciones geográficas en la disponibilidad de la API en algunas regiones.

Llama 3.1 70B (Meta) Ventajas:

- Código abierto, permitiendo despliegue propio y control total sobre el modelo.

- Sin costes por uso una vez desplegado (solo costes de infraestructura). En la tabla 4.4 se han incluido arteriscos en los costes incluidos porque, aunque sea código abierto, requiere costes de hardware por una infraestructura propia (GPUs).
- Posibilidad de fine-tuning específico para el dominio educativo.

Desventajas:

- Requiere infraestructura de cómputo especializada (múltiples GPUs de alto rendimiento, gestión de modelos).
- Calidad de adaptación pedagógica inferior a los modelos propietarios evaluados.
- Complejidad operativa significativamente mayor (mantenimiento, actualizaciones, monitorización).
- Tiempo de respuesta más lento incluso con hardware dedicado.

En resumen, la elección de Claude 3.7 Sonnet como modelo principal para la transformación de textos se fundamenta en un análisis multidimensional que pondera calidad, coste, complejidad técnica y viabilidad operativa:

1. **Adherencia a instrucciones:** Claude demostró consistentemente una capacidad superior (95 %) para seguir el prompt detallado que especifica la estructura HTML fija del formato LEdui, minimizando la necesidad de post-procesamiento y validación de outputs.
2. **Equilibrio calidad-coste:** aunque Gemini presenta un coste inferior, la diferencia de calidad (20 puntos porcentuales en adherencia a estructura) y la mayor variabilidad de resultados compensan el ahorro económico. En un sistema educativo, la calidad y consistencia de los contenidos generados es prioritaria.
3. **Velocidad de respuesta:** Claude ofrece tiempos de respuesta (3-5 segundos) adecuados para uso interactivo, significativamente mejores que Gemini (5-8 seg) y Llama (8-12 seg).
4. **Viabilidad operativa:** las opciones de autohospedaje (Llama) fueron descartadas por requerir inversión en infraestructura GPU y conocimientos especializados en operación de modelos, incompatibles con los recursos disponibles para un proyecto de TFG.
5. **Tono pedagógico:** Claude genera textos con un registro naturalmente claro, motivador y empático sin necesidad de prompt engineering avanzado, mientras que GPT-4o tiende a un registro más formal que requiere ajustes.

Esta decisión técnica, fundamentada en evaluaciones empíricas con varios enunciados de prueba, ha resultado determinante para el éxito del sistema, garantizando transformaciones de alta calidad pedagógica con tiempos de respuesta y costes razonables para un entorno educativo real.

4.5.2. Endpoints de generación de imágenes de contexto y pictogramas

Este conjunto de endpoints se encarga de enriquecer los contenidos educativos adaptados mediante la incorporación de apoyos visuales y recursos multimedia, con el objetivo de mejorar la comprensión del texto y reforzar su accesibilidad desde un enfoque multimodal.

Las principales funcionalidades de este bloque de endpoints son:

- Identificación de conceptos y términos clave dentro del contenido educativo adaptado.

Tabla 4.5: Endpoints de generación de imágenes de contexto y pictogramas

POST	`/generar_imagenes_contexto`	Genera imágenes de contexto asociadas al enunciado adaptado.
POST	`/generar_pictogramas`	Genera pictogramas educativos asociados al contenido.

- Consulta a la API de ARASAAC para la obtención de pictogramas asociados semánticamente al contenido.
- Descarga, almacenamiento y gestión de los recursos gráficos obtenidos.
- Asociación coherente de pictogramas al contenido textual adaptado, garantizando consistencia semántica.
- Generación de recursos multimedia complementarios, como imágenes educativas contextualizadas a partir del contenido adaptado.
- Enriquecimiento del contenido educativo mediante soportes visuales que refuercen la comprensión y reduzcan la carga cognitiva.

4.5.2.1. Integración con Claude para análisis semántico y extracción de acciones

El modelo Claude 3.7 Sonnet identifica automáticamente las competencias clave que debe trabajar el alumnado en cada enunciado adaptado. Estas competencias se expresan mediante verbos simples de una sola palabra (por ejemplo, “analizar”, “crear”, “buscar”), lo que facilita su posterior conversión en pictogramas comprensibles.

Este proceso de simplificación es fundamental para garantizar la compatibilidad entre el contenido textual adaptado y los sistemas pictográficos de ARASAAC. El análisis se realiza mediante instrucciones precisas que aseguran resultados consistentes y reutilizables.

4.5.2.2. Integración con ARASAAC para pictogramas educativos

Para la incorporación de pictogramas educativos, el sistema se integra directamente con la API pública de ARASAAC, referente internacional en el ámbito de los sistemas aumentativos y alternativos de comunicación. A partir de las acciones simplificadas obtenidas mediante el análisis semántico previo, el sistema consulta la API de ARASAAC para localizar pictogramas reales, normalizados y ampliamente validados en contextos educativos inclusivos.

Los recursos gráficos obtenidos se descargan y gestionan automáticamente, y se componen en una imagen final que agrupa los pictogramas asociados a la actividad. Este enfoque garantiza el uso de símbolos estandarizados, evita la generación artificial de iconografía no validada y asegura el respeto a las licencias de uso de los recursos.

4.5.2.3. Evaluación y selección de modelos de generación de imágenes

Para la generación de imágenes educativas de contexto, el sistema adopta un enfoque combinado:

- **Claude 3.7 Sonnet** analiza el enunciado adaptado, identifica los conceptos clave y genera prompts detallados con alto valor pedagógico, describiendo escenas educativas claras y contextualizadas.
- **Google Gemini 2.5 Flash Image** procesa estos prompts y genera ilustraciones en formato panorámico (16:9), con estilo educativo claro y no decorativo.

Las imágenes generadas se acompañan siempre de una justificación educativa que explica al alumnado qué información aporta la imagen y por qué resulta relevante para la actividad, reforzando el papel pedagógico del recurso visual.

Durante el desarrollo se llevó a cabo una fase de experimentación con distintos servicios de generación de imágenes mediante IA, con el objetivo de seleccionar la solución más adecuada para crear recursos visuales educativos de contexto. Se evaluaron tres opciones:

- **DALL-E 3 (OpenAI):** Modelo de alta calidad para generación de imágenes, conocido por su capacidad para interpretar prompts complejos.
- **Freepik API:** Servicio especializado en imágenes de stock con capacidades de generación mediante IA.
- **Google Gemini 2.5 Flash Image (Nano Banana) [MODELO SELECCIONADO]:** Modelo de Google optimizado para generación rápida de imágenes educativas.

Las pruebas con **DALL-E 3 y Freepik** revelaron diversas limitaciones relevantes desde el punto de vista pedagógico:

- **Coherencia educativa variable:** Las imágenes generadas, aunque visualmente atractivas, no siempre reflejaban con precisión el contexto educativo específico descrito en los prompts.
- **Tiempos de generación elevados:** DALL-E 3 requería entre 8-15 segundos por imagen, mientras que Freepik presentaba tiempos aún superiores (15-20 segundos).
- **Coste:** DALL-E 3 presentaba un coste por imagen significativamente superior a las alternativas evaluadas.
- **Estilo inconsistente:** Alta variabilidad en el estilo visual entre imágenes generadas con prompts similares, dificultando mantener una identidad visual coherente en el sistema.

Google Gemini 2.5 Flash Image (Nano Banana) fue seleccionado por ofrecer el mejor equilibrio entre los criterios evaluados:

- **Coherencia con prompts educativos:** Demostró mejor capacidad para interpretar y representar fielmente contextos educativos específicos.
- **Velocidad de respuesta:** Tiempos de generación entre 4-8 segundos, permitiendo una experiencia de usuario más fluida.
- **Estabilidad y consistencia:** Mayor previsibilidad en los resultados, facilitando la generación de imágenes con estilo visual coherente.
- **Coste competitivo:** Precio inferior a DALL-E 3 con calidad comparable para el caso de uso educativo.
- **Formato nativo 16:9:** Genera imágenes directamente en formato panorámico educativo sin necesidad de post-procesamiento.

A su vez, cabe comentar que el diseño modular del sistema permite sustituir el servicio de generación de imágenes utilizado con modificaciones mínimas en el código, lo que facilitaría:

- Migración a modelos más avanzados cuando estén disponibles.
- Implementación de selección dinámica de servicio según características de la imagen requerida.
- Evaluación continua de nuevos servicios mediante pruebas.

4.5.3. Endpoints de autenticación y gestión de usuarios

Este conjunto de endpoints se encarga de la identificación y autorización de los usuarios que acceden al sistema. La autenticación se realiza mediante integración con servicios externos de Google, utilizando mecanismos basados en OAuth 2.0.

Tabla 4.6: Endpoint de autenticación y gestión de usuario

POST	`/auth/google`	Autenticación de usuarios mediante cuenta de Google (OAuth 2.0).
GET	`/auth/me`	Obtiene la información del usuario autenticado.
GET	`/auth/stats`	Devuelve estadísticas de uso asociadas al usuario.

Las principales funcionalidades ofrecidas por estos endpoints son:

- Autenticación de usuarios mediante cuenta de Google.
- Validación de tokens de acceso.
- Gestión de sesiones de usuario.
- Asignación y comprobación de roles (usuario estándar y administrador).

4.5.3.1. Integración con Google OAuth 2.0

La gestión de usuarios se ha implementado mediante la integración con APIs de Google, siguiendo un enfoque de delegación de la autenticación. Esta decisión permite apoyarse en un proveedor consolidado, reduciendo la complejidad del sistema y mejorando los niveles de seguridad y fiabilidad.

La autenticación se basa en mecanismos estándar de identidad ofrecidos por Google, que permiten verificar de forma segura la identidad del usuario sin necesidad de gestionar credenciales propias. Una vez autenticado el usuario, el sistema gestiona internamente la sesión y los permisos asociados, diferenciando entre perfiles de usuario estándar y perfiles de administración.

4.5.4. Endpoints de gestión del historial y persistencia

El sistema incluye endpoints destinados a la gestión y consulta del historial de adaptaciones realizadas por cada usuario. Estos endpoints interactúan directamente con la base de datos del sistema.

Tabla 4.7: Endpoint de gestión del historial y persistencia

GET	`/mi-historial`	Obtiene el historial de transformaciones del usuario.
GET	`/mi-historial/{transformacion_id}`	Obtiene el detalle de una transformación concreta.
GET	`/mi-historial/{transformacion_id}/download-docx`	Descarga la transformación en formato DOCX.

Entre sus funcionalidades se encuentran:

- Almacenamiento de contenidos originales y adaptados.
- Registro de metadatos asociados a cada adaptación.

- Consulta del historial de transformaciones realizadas por un usuario.
- Recuperación de adaptaciones anteriores para su reutilización.
- Exportación de transformaciones en formato DOCX.

Este bloque funcional permite dotar al sistema de trazabilidad y continuidad en el uso, aspectos especialmente relevantes en contextos educativos. El modelo de datos relacional garantiza la integridad de la información y permite consultas eficientes del historial mediante índices optimizados.

4.5.5. Endpoints de administración

Finalmente, el sistema dispone de endpoints reservados a perfiles de administración, orientados a la supervisión y gestión global del servicio.

Tabla 4.8: Endpoints de administración

GET	/admin/users	Obtiene el listado completo de usuarios del sistema.
GET	/admin/transformaciones	Obtiene todas las transformaciones realizadas.
GET	/admin/transformaciones/{transformacion_id}	Obtiene el detalle de una transformación concreta.
GET	/admin/transformaciones/{transformacion_id}/download-docx	Descarga una transformación en formato DOCX.
GET	/admin/pictogramas/{pictograma_id}/imagen	Obtiene la imagen asociada a un conjunto de pictogramas.
GET	/admin/imagenes/{imagen_id}/imagen	Obtiene una imagen de contexto generada.

Las funciones principales de estos endpoints incluyen:

- Gestión de usuarios y permisos.
- Consulta de métricas de uso del sistema.
- Monitorización del funcionamiento general del servicio.
- Análisis de patrones de uso y rendimiento.
- Acceso completo a transformaciones y recursos generados por todos los usuarios.

Estos endpoints permiten evaluar el uso real del sistema y facilitan la detección de posibles mejoras o necesidades de ajuste.

4.6. IMPLEMENTACIÓN DEL *FRONTEND*

La creación de una interfaz web, el *frontend*, ha constituido un elemento clave en el desarrollo del sistema propuesto, al permitir materializar de forma práctica el modelo de transformación del contenido educativo definido en este Trabajo Fin de Grado.

Aunque la interfaz web no formaba parte de los requisitos iniciales del producto mínimo viable, su desarrollo se identificó tempranamente como el primer requisito deseable, al resultar fundamental tanto para la validación funcional del sistema como para el desarrollo de la **Fase 5 del plan de trabajo**, centrada en la evaluación y refinamiento de la solución.

Esta interfaz web permite interactuar de manera directa con el servicio web desarrollado, facilitando la introducción de enunciados educativos y la visualización inmediata de los contenidos adaptados al formato de Lectura Educativa Inclusiva (LEdui). De este modo, actúa como un elemento mediador entre el modelo computacional de adaptación y su aplicación práctica en un contexto educativo real.

4.6.1. Experiencia de usuario en la interfaz web

La experiencia de usuario de la interfaz web desarrollada se ha diseñado a partir de una organización por pantallas, cada una de ellas asociada a un momento concreto de la interacción con el sistema a través de, principalmente, el conjunto específico de endpoints del servicio web. Este enfoque permite describir de forma clara y progresiva las funcionalidades disponibles para cada tipo de usuario, manteniendo una navegación sencilla y coherente con los principios de accesibilidad cognitiva.

4.6.1.1. Pantalla de inicio de sesión

La primera interacción del usuario con la interfaz web corresponde a la pantalla de inicio de sesión, donde se realiza el proceso de autenticación necesario para acceder al sistema. La misma se muestra en la figura 4.6. En esta pantalla, el usuario debe introducir sus credenciales

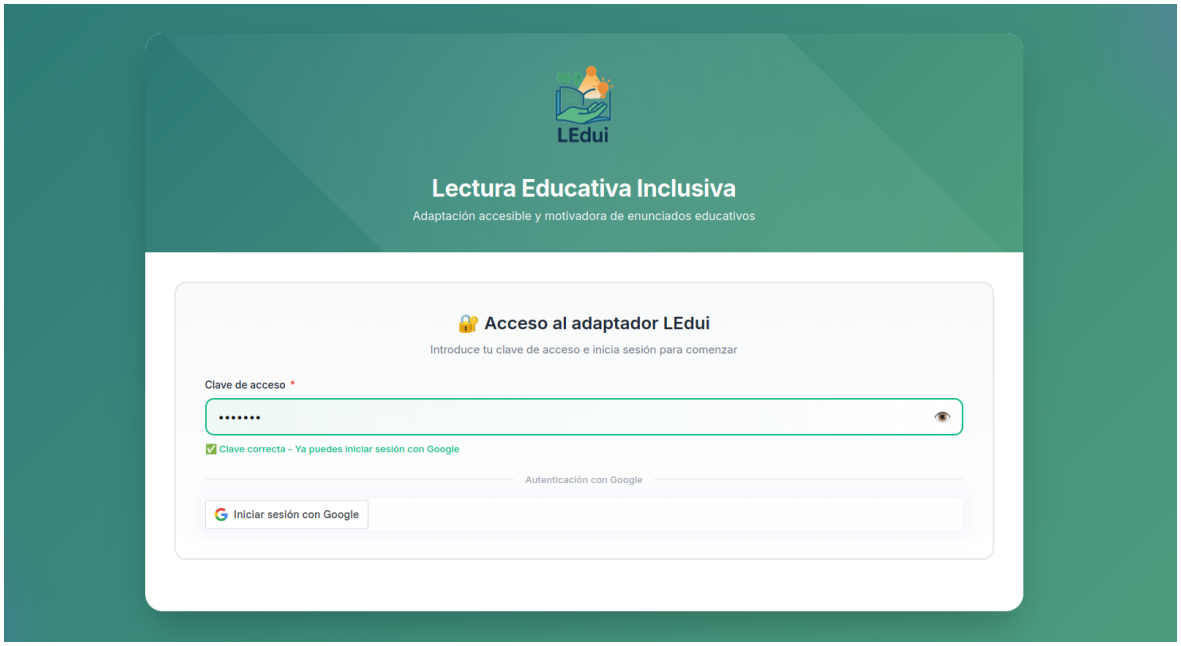


Figura 4.6: Pantalla de inicio de sesión

personales junto con una clave general de acceso asociada a su tipo de usuario. Esta clave

adicional actúa como un mecanismo complementario de control, orientado a restringir el acceso al sistema y a prevenir usos inadecuados por parte de usuarios no autorizados.

La interfaz web invoca en esta fase los endpoints de autenticación y validación, encargados de verificar tanto la identidad del usuario como la validez de la clave de acceso proporcionada. Una vez completado el proceso de autenticación, el sistema recupera la información asociada al perfil del usuario y determina su rol dentro de la plataforma.

En función de dicho rol, el usuario es redirigido automáticamente a la pantalla correspondiente, diferenciando entre usuarios con perfil de administración y usuarios con perfil general. Este enfoque permite adaptar la experiencia de uso desde el primer momento y refuerza la seguridad del sistema mediante un control de acceso basado en roles.

4.6.1.2. Pantalla de creación de una nueva transformación

La pantalla de creación de una nueva transformación constituye el punto de interacción principal de la interfaz web y el uso habitual del sistema, dado que la mayoría de los usuarios no disponen de perfil de administración y acceden a la plataforma con el objetivo de adaptar enunciados educativos concretos al formato LEdui. La misma se muestra en la figura 4.7. Desde esta pantalla, el usuario puede iniciar el proceso de transformación de forma guiada

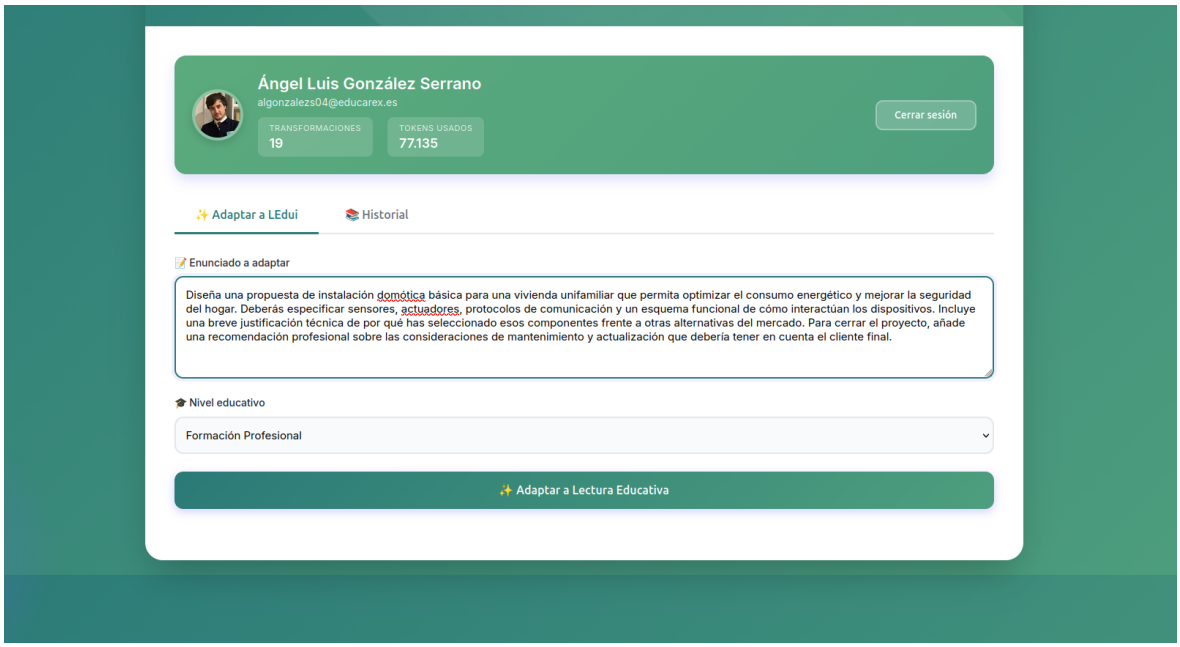


Figura 4.7: Pantalla para realizar transformación

y sencilla. En primer lugar, selecciona el nivel educativo correspondiente al contenido a adaptar y, a continuación, introduce el enunciado educativo original en formato textual. Una vez completados estos pasos, el usuario activa el proceso mediante una acción única claramente identificada en la interfaz, pulsando el botón para tal fin.

En este punto, la interfaz web invoca inicialmente al endpoint de transformación del enunciado, que constituye el núcleo funcional del sistema y aplica de forma armonizada y determinista las diez reglas del estándar LEdui. El resultado de esta llamada es un enunciado adaptado con una estructura fija, predecible y alineada con los principios de inclusión contemplados y desarrollados en este TFG. Tal y como se muestra en la figura 4.8. Con el enunciado adaptado, el sistema continúa generandode forma automática más información



Figura 4.8: Pantalla mostrando enunciado adaptado

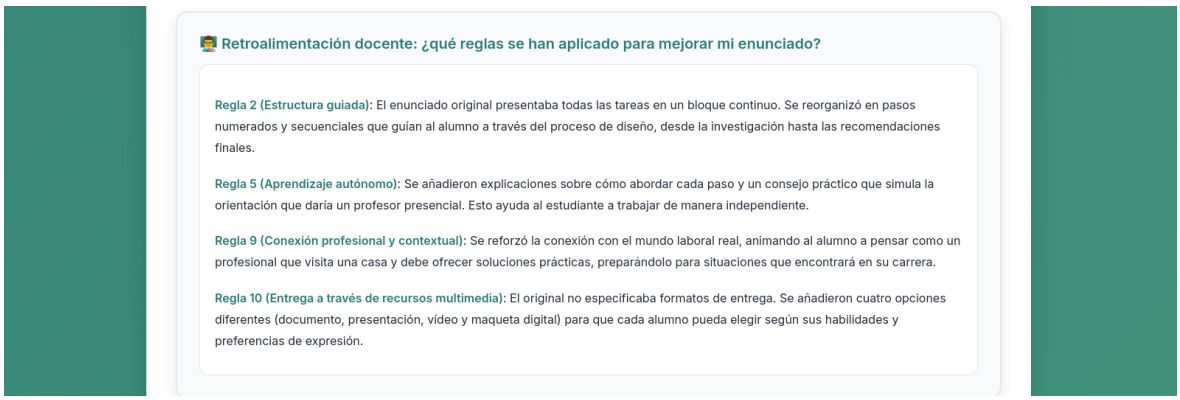


Figura 4.9: Pantalla mostrando explicación pedagógicas

clave parte de la transformación, a través de una serie de llamadas secuenciales a endpoints complementarios que enriquecen el contenido generado.

En primer lugar, muestra la explicación pedagógica dirigida al docente, que produce un texto explicativo asociado al enunciado adaptado. Esta explicación tiene como finalidad orientar al profesorado en el uso del formato LEdui, destacando aquellas reglas que han tenido mayor presencia en la adaptación realizada y ofreciendo recomendaciones para la elaboración de futuros enunciados educativos, tal y como se muestra en la figura 4.9. De este modo, la interfaz no solo ofrece el resultado final, sino que favorece un aprendizaje progresivo por parte del docente en relación con los principios del formato. A continuación, el sistema extrae las competencias implícitas en la actividad propuesta y, para cada una de ellas, consulta la API de ARASAAC con el objetivo de obtener un pictograma representativo. El resultado es un conjunto de competencias acompañadas de apoyos visuales, que se presentan de forma integrada en la interfaz y refuerzan la comprensión del propósito educativo de la actividad tanto para el docente como para el alumnado. Tal y como se muestra en la figura 4.10. Finalmente, el sistema invoca un endpoint de generación de imagen de contexto, cuyo objetivo es producir un recurso visual ilustrativo que acompañe al texto del enunciado

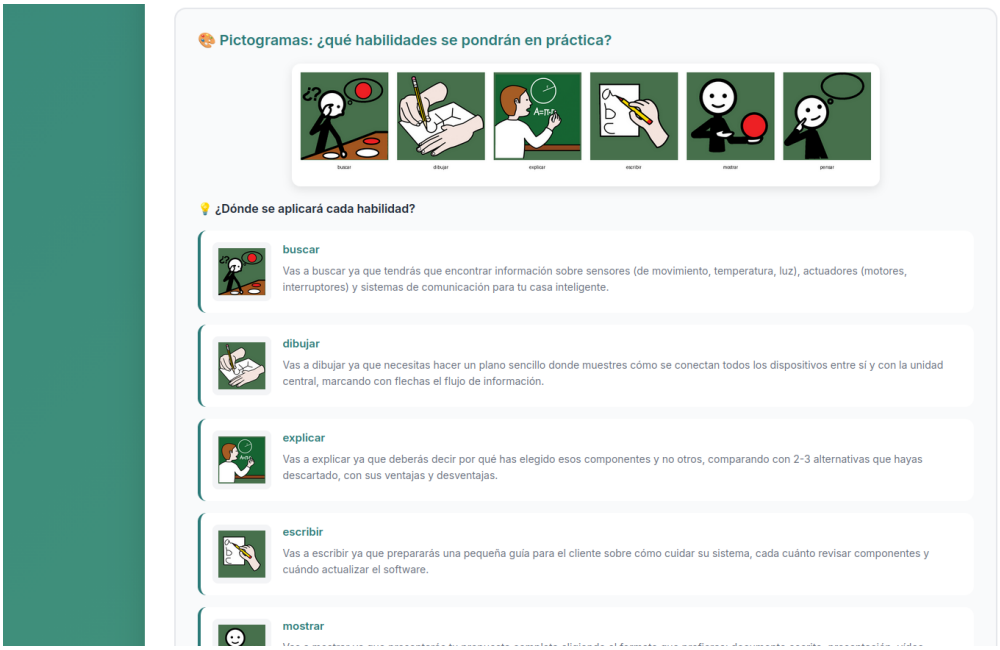


Figura 4.10: Pantalla mostrando pictogramas

adaptado. Esta imagen se genera de manera coherente con el contenido de la actividad y se presenta junto con un pie explicativo que contextualiza visualmente la tarea, tal y como se muestra en la figura 4.11. Este recurso contribuye a mejorar la comprensión global del enunciado, facilitar la anticipación de la actividad y captar la atención del alumnado. La ejecución secuencial de estos endpoints da lugar a una transformación completa del

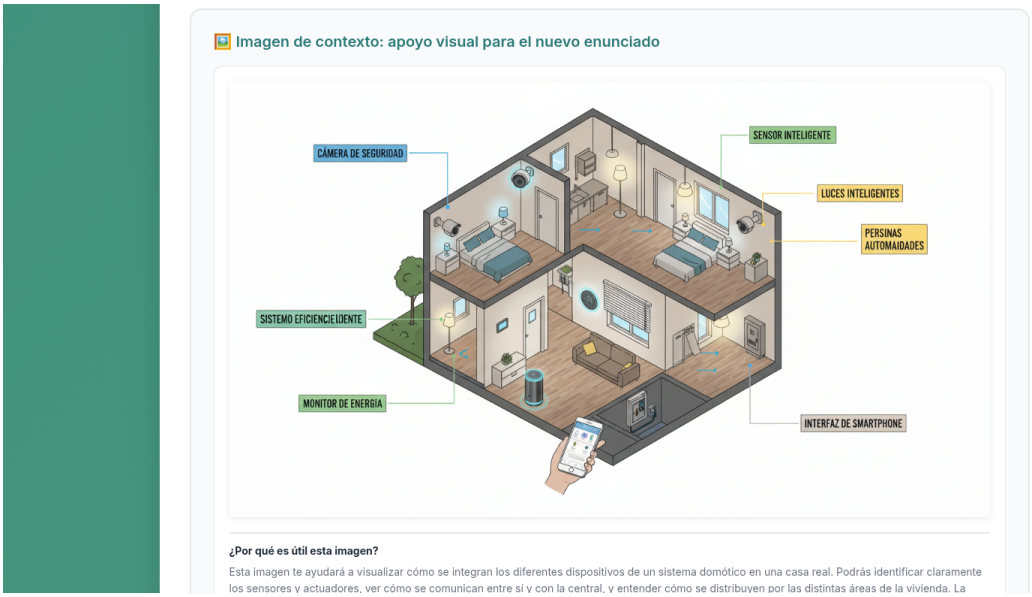


Figura 4.11: Pantalla mostrando imágenes de contexto

contenido educativo, que no se limita a la adaptación textual al formato LEdui, sino que incorpora explicaciones pedagógicas, apoyos visuales y recursos contextuales. Este enfoque modular y orquestado permite que cada componente del sistema pueda evolucionar de forma independiente, manteniendo al mismo tiempo la coherencia del resultado final y facilitando la extensibilidad futura de la solución.

4.6.1.3. Pantalla historial de transformaciones

La pantalla de historial de transformaciones permite al usuario consultar y gestionar las adaptaciones de contenidos realizadas previamente a través del sistema. Esta funcionalidad se apoya en los endpoints de persistencia, responsables de almacenar de forma estructurada la información asociada a cada transformación y de facilitar su recuperación posterior.

El acceso a esta pantalla se realiza desde una pestaña integrada en la propia pantalla de creación de nuevas transformaciones, lo que permite al usuario alternar de forma sencilla entre la generación de nuevos enunciados y la consulta de resultados previos sin abandonar el area principal de trabajo. Tal y como se muestra en la figura 4.12. En esta pantalla, la

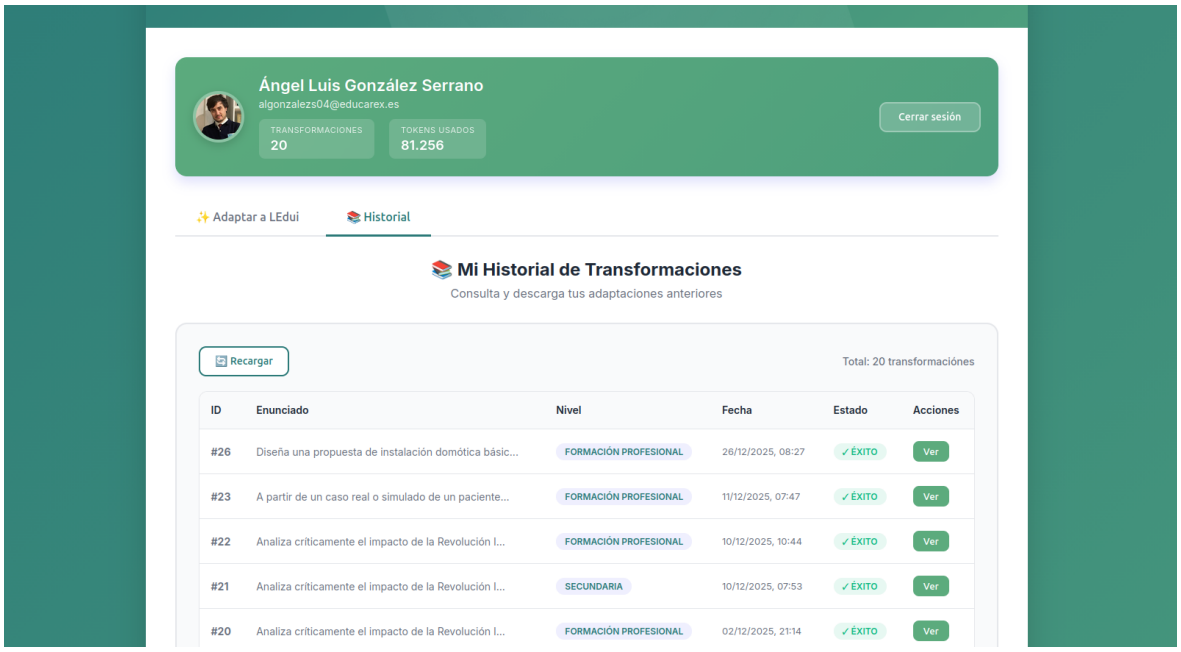


Figura 4.12: Pantalla con el historial de un usuario

interfaz web presenta una tabla con el conjunto de transformaciones realizadas por el usuario, mostrando información relevante como la fecha de creación, el nivel educativo seleccionado y el estado de la transformación. Desde esta tabla, el usuario puede seleccionar una transformación concreta para consultar en detalle el contenido adaptado, accediendo al resultado completo generado por el sistema.

Además, la pantalla ofrece la posibilidad de descargar cada transformación en un formato ofimático editable de texto, lo que permite al profesorado reutilizar y modificar los contenidos adaptados en sus propios materiales educativos. Esta funcionalidad se implementa mediante endpoints específicos de recuperación y exportación de datos, que transforman la información almacenada en la base de datos en documentos compatibles con herramientas ofimáticas habituales. Tal y como se muestra en la figura 4.13. La disponibilidad de un historial persistente de transformaciones refuerza el carácter práctico del sistema, al permitir no solo la generación puntual de contenidos adaptados, sino también su consulta, revisión y reutilización a lo largo del tiempo. De este modo, la pantalla de historial contribuye a integrar el uso del formato LEdui en la práctica docente cotidiana, apoyándose en mecanismos de persistencia que garantizan la continuidad y trazabilidad del trabajo realizado.

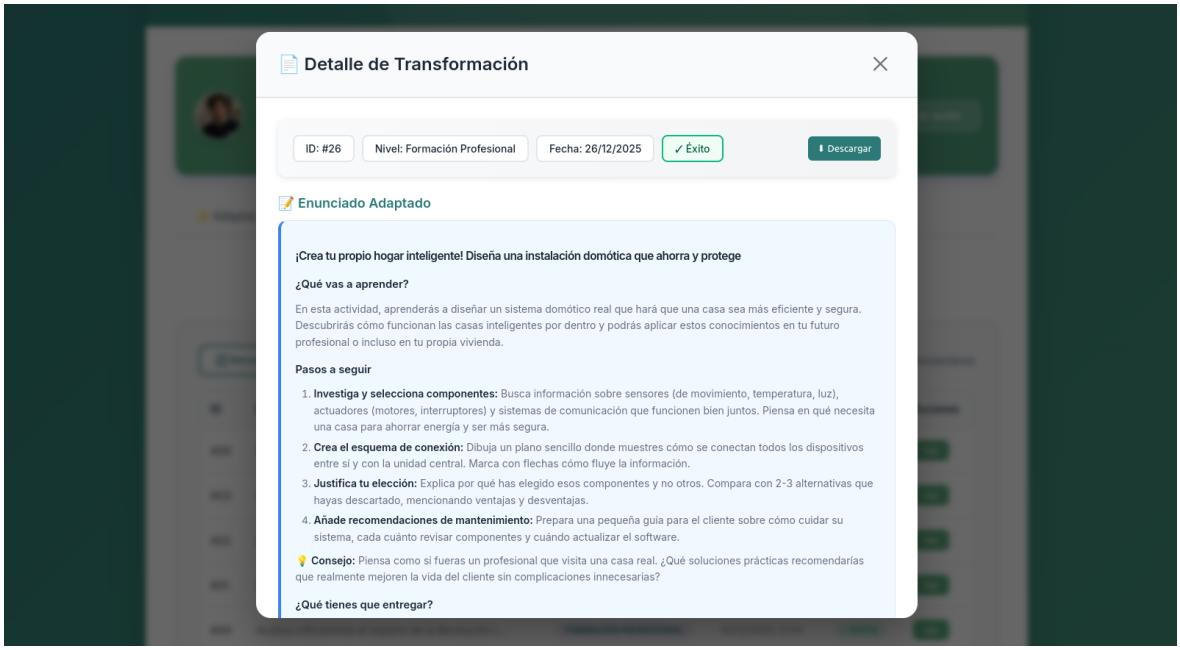


Figura 4.13: Pantalla mostrando transformación concreta y botón descargar

4.6.1.4. Pantalla de administración

La pantalla de administración está resstringida a los usuarios con dicho perfil y proporciona una visión global del funcionamiento del sistema. Esta pantalla centraliza las funcionalidades de supervisión y gestión, apoyándose de forma directa en los endpoints de persistencia, que permiten acceder a la información almacenada de manera estructurada y trazable.

Por un lado, la interfaz presenta una tabla de usuarios del sistema, obtenida mediante los endpoints de persistencia. Desde esta tabla es posible consultar los usuarios registrados, facilitando el control del uso del sistema y reforzando la seguridad de la plataforma. Tal y como se muestra en la figura 4.14. Por otro lado, la pantalla de administración muestra una tabla con el conjunto de transformaciones realizadas en el sistema, sin restricción por usuario. Esta tabla ofrece las mismas funcionalidades disponibles en la pantalla de historial de transformaciones del usuario general, consulta del contenido adaptado y descarga en formato ofimático editable, pero aplicadas de forma global. Tal y como se muestra en 4.15. Inicialmente, esta funcionalidad se concibió como exclusiva del perfil de administración. Sin embargo, tras comprobar su utilidad práctica durante el desarrollo, se decidió extenderla también a los usuarios generales, manteniendo para el administrador una visión completa de todas las transformaciones realizadas. La reutilización de los mismos mecanismos de consulta y exportación tanto para usuarios generales como para administradores ha sido posible gracias a la arquitectura basada en endpoints de persistencia, que desacoplan el almacenamiento de los datos de su presentación en la interfaz. Este enfoque ha permitido ofrecer distintas vistas de la información sin duplicar lógica en el *backend*, facilitando la mantenibilidad y evolución futura del sistema.

En conjunto, la pantalla de administración proporciona una herramienta eficaz para la gestión y supervisión del sistema, reforzando la trazabilidad de las transformaciones realizadas y permitiendo un control centralizado del uso de la plataforma.

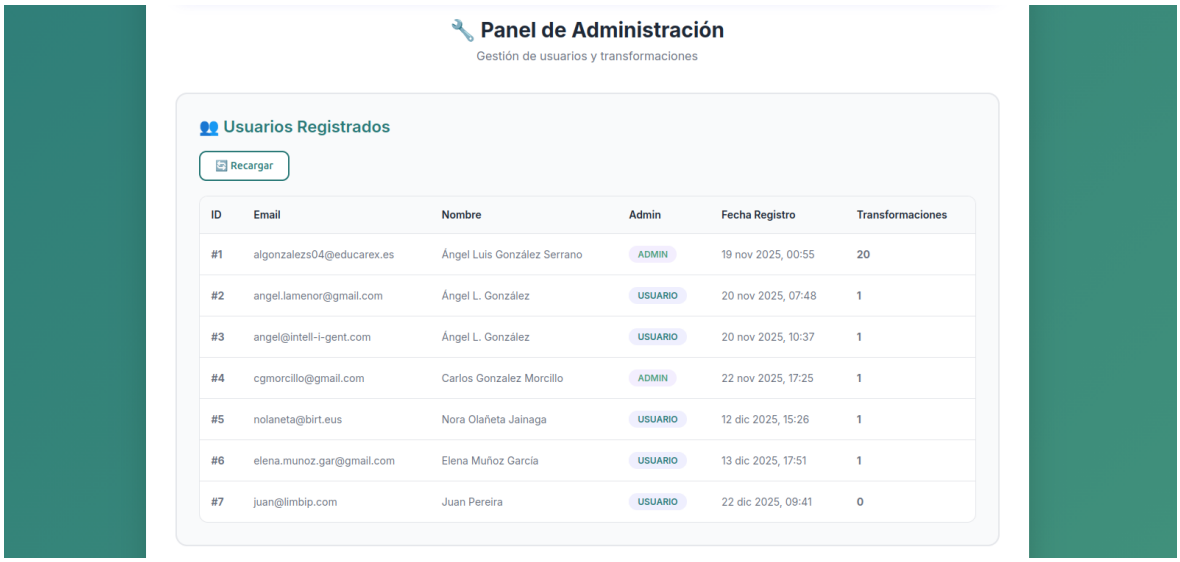


Figura 4.14: Pantalla de administración: usuarios

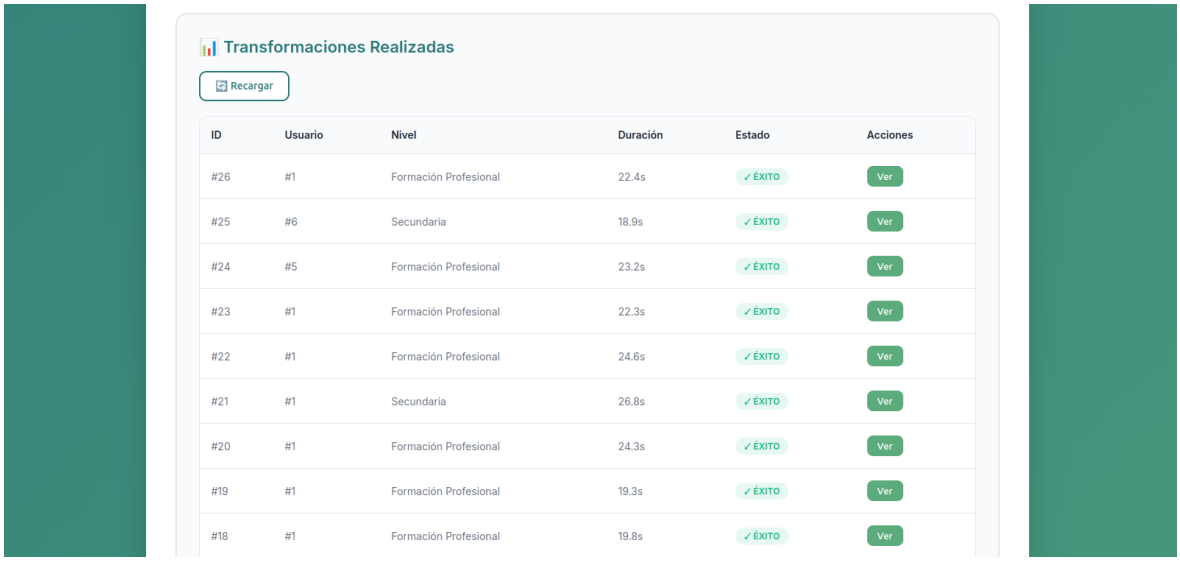


Figura 4.15: Pantalla de administración: transformaciones

4.7. DESPLIEGUE DEL SISTEMA

El despliegue del sistema constituye una fase crítica del desarrollo, en la que la solución implementada se traslada desde un entorno de desarrollo local hacia una infraestructura accesible públicamente. Este apartado detalla las decisiones arquitectónicas, la configuración técnica y el proceso seguido para el despliegue del sistema.

4.7.1. Arquitectura de despliegue

El sistema desarrollado sigue una arquitectura cliente-servidor desplegada en la nube, en la que el *backend* (API REST desarrollada con FastAPI) y la base de datos (PostgreSQL) se ejecutan en un entorno gestionado, mientras que el *frontend* (interfaz web estática) se sirve directamente desde el mismo servidor que expone la API.

Esta arquitectura de despliegue presenta las siguientes características:

- **Servidor de aplicación**: aloja el *backend* desarrollado en Python con FastAPI, responsable de procesar las peticiones HTTP, integrar los servicios externos de IA y gestionar la lógica de negocio del sistema.
- **Base de datos gestionada**: PostgreSQL desplegada como servicio gestionado, garantizando persistencia, disponibilidad y copias de seguridad automáticas.
- **Servidor web integrado**: FastAPI incluye su propio servidor web (Uvicorn), que sirve tanto los endpoints de la API como los archivos estáticos del *frontend*, simplificando la configuración de despliegue.
- **Certificado SSL/TLS**: la plataforma de despliegue proporciona certificados SSL automáticos, garantizando comunicaciones cifradas mediante HTTPS.

Desde el punto de vista de la infraestructura, el sistema se despliega como un único contenedor que agrupa *backend* y *frontend*, conectado a una base de datos PostgreSQL externa. Esta configuración simplifica la gestión del despliegue y resulta adecuada para el alcance del proyecto, aunque en un entorno de producción a gran escala sería recomendable separar *backend* y *frontend* en servicios independientes.

4.7.2. Selección de la plataforma de despliegue

Para el despliegue del sistema se ha seleccionado **Railway**, una plataforma de infraestructura como servicio (PaaS) orientada al despliegue de aplicaciones web. La elección de Railway se fundamenta en los siguientes criterios técnicos:

- Soporte nativo para aplicaciones Python con FastAPI
- Provisión automática de base de datos PostgreSQL
- Gestión automática de certificados SSL
- Integración directa con repositorios Git
- Plan gratuito adecuado para la fase de validación del proyecto

Otras alternativas evaluadas fueron servicios de infraestructura en la nube como AWS o Google Cloud Platform, descartadas por requerir mayor complejidad de configuración o presentar limitaciones en los planes gratuitos.

4.7.3. Configuración del entorno de despliegue

El despliegue del sistema requiere la configuración de un conjunto de variables de entorno que permiten al *backend* acceder a servicios externos y gestionar aspectos de seguridad. Estas variables se gestionan de forma segura a través del panel de configuración de Railway, evitando su exposición en el código fuente.

4.7.3.1. Variables de entorno obligatorias

Las siguientes variables de entorno son necesarias para el correcto funcionamiento del sistema:

Variable	Descripción
`ANTHROPIC_API_KEY`	Clave de acceso a la API de Anthropic (Claude) para transformación de contenidos
`GOOGLE_AI_API_KEY`	Clave de acceso a Google Gemini para generación de imágenes de contexto
`DATABASE_URL`	Cadena de conexión a la base de datos PostgreSQL (generada automáticamente por Railway)
`SECRET_KEY`	Clave secreta para la firma de tokens JWT (generada aleatoriamente)
`GOOGLE_CLIENT_ID`	Identificador de cliente OAuth de Google para autenticación de usuarios
`GOOGLE_CLIENT_SECRET`	Secreto de cliente OAuth de Google
`VALID_ACCESS_KEY`	Clave de acceso general para usuarios estándar
`ADMIN_ACCESS_KEY`	Clave de acceso para usuarios administradores

Tabla 4.9: Variables de entorno obligatorias del sistema

4.7.3.2. Variables de entorno opcionales

Adicionalmente, se pueden configurar las siguientes variables opcionales para ajustar el comportamiento del sistema:

Variable	Descripción
`FRONTEND_URL`	URL del *frontend* para configuración de CORS (por defecto: localhost)
`PORT`	Puerto en el que se ejecuta el servidor (Railway lo asigna automáticamente)
`ENVIRONMENT`	Entorno de ejecución: development, staging o production

Tabla 4.10: Variables de entorno opcionales

4.7.3.3. Gestión segura de credenciales

Las claves de acceso a servicios externos (APIs de Anthropic, Google) y las claves secretas (JWT, OAuth) se gestionan exclusivamente mediante variables de entorno, siguiendo las mejores prácticas de seguridad en el desarrollo de aplicaciones web:

- Las credenciales no se almacenan en el código fuente ni en el repositorio Git.
- Se utiliza un archivo `.env` local para desarrollo, incluido en `.gitignore` para evitar su subida accidental al repositorio.

- En el entorno de producción, las variables se configuran a través del panel web de la plataforma, que las almacena de forma cifrada.
- Las claves secretas se generan mediante algoritmos criptográficamente seguros (256 bits de entropía).

4.7.4. Proceso de despliegue

El despliegue del sistema en Railway se ha realizado siguiendo un proceso estructurado en varias etapas, orientado a garantizar la estabilidad y trazabilidad del sistema desplegado.

4.7.4.1. Etapa 1. Preparación del repositorio

Antes de proceder al despliegue, se han realizado las siguientes preparaciones en el repositorio de código:

- Creación del archivo `requirements.txt` con todas las dependencias necesarias del proyecto, especificando versiones exactas para garantizar reproducibilidad.
- Configuración del archivo `runtime.txt` para especificar la versión de Python requerida (Python 3.11).
- Creación de un archivo `Procfile` que indica cómo iniciar la aplicación mediante Uvicorn:

```
web: uvicorn main_v2:app --host 0.0.0.0 --port $PORT
```

- Inclusión de archivos de configuración para el correcto funcionamiento del *frontend* en el directorio `static/`.

4.7.4.2. Etapa 2. Creación del proyecto en Railway

El despliegue inicial se ha realizado mediante la interfaz web de Railway, siguiendo estos pasos:

1. Creación de un nuevo proyecto vinculado al repositorio Git del sistema.
2. Configuración del servicio de base de datos PostgreSQL.
3. Configuración de todas las variables de entorno necesarias a través del panel de configuración.
4. Activación del despliegue automático desde la rama principal del repositorio.

4.7.4.3. Etapa 3. Migración de la base de datos

Una vez desplegado el *backend* y conectada la base de datos, se ha procedido a inicializar el esquema de datos mediante SQLAlchemy:

- Creación automática de las tablas necesarias al iniciar la aplicación: `users`, `transformacion_enun` `generacion_pictogramas`, `generacion_imagenes`.
- Verificación de las relaciones entre tablas (claves foráneas) y los índices automáticos generados para optimizar consultas.

4.7.4.4. Etapa 4. Configuración de OAuth con Google

Para que el sistema de autenticación funcione correctamente en el entorno de producción, ha sido necesario configurar la aplicación OAuth en Google Cloud Console:

- Creación de un proyecto en Google Cloud Console.
- Configuración de la pantalla de consentimiento OAuth.
- Generación de credenciales OAuth 2.0 (Client ID y Client Secret).
- Configuración de las URLs autorizadas de redirección, incluyendo tanto `localhost` para desarrollo como la URL de producción.
- Registro de los orígenes JavaScript permitidos para evitar errores de CORS.

4.7.4.5. Etapa 5. Verificación del despliegue

Una vez completado el despliegue, se ha verificado el correcto funcionamiento del sistema mediante las siguientes comprobaciones:

- Acceso a la URL pública del sistema y verificación de que el *frontend* se carga correctamente.
- Prueba del flujo completo de autenticación mediante Google OAuth.
- Verificación de la conectividad con la base de datos PostgreSQL.
- Prueba de transformación completa de un enunciado, incluyendo generación de pictogramas e imágenes.
- Revisión de logs del servidor para detectar posibles errores o warnings.
- Comprobación de los tiempos de respuesta y del comportamiento del sistema bajo carga moderada.

4.7.5. Gestión de la base de datos

La base de datos PostgreSQL se ejecuta como un servicio gestionado, lo que simplifica aspectos críticos de su administración:

- **Escalabilidad de almacenamiento**: El espacio de almacenamiento asignado puede ampliarse de forma dinámica según las necesidades.
- **Monitorización de rendimiento**: La plataforma proporciona métricas básicas sobre el uso de CPU, memoria y almacenamiento.
- **Acceso seguro**: La base de datos solo es accesible desde la aplicación desplegada mediante conexión cifrada, no estando expuesta públicamente.

Durante las fases de validación, el volumen de datos almacenado ha sido moderado tal y como se expone en el plan de pruebas, sin generar problemas de rendimiento ni limitaciones de almacenamiento.

4.7.6. Monitorización y registro de eventos

Para garantizar la trazabilidad del funcionamiento del sistema en producción, se ha implementado un sistema de logging que registra eventos relevantes:

- **Logs de aplicación**: FastAPI genera logs automáticos de las peticiones HTTP recibidas, incluyendo método, endpoint, código de respuesta y tiempo de procesamiento.

- **Logs de errores**: Los errores no controlados se registran con su traza completa (stack trace), facilitando la depuración.
- **Logs de integración con APIs externas**: Se registran las llamadas a servicios de Anthropic, Google Gemini y ARASAAC, incluyendo tokens consumidos y tiempos de respuesta.
- **Acceso a logs**: Railway proporciona una interfaz web para visualizar los logs en tiempo real, así como la posibilidad de descargarlos para análisis offline.

4.7.7. Configuración de CORS y seguridad

Dado que el *frontend* y el *backend* se ejecutan en el mismo dominio, la configuración de CORS (Cross-Origin Resource Sharing) se ha simplificado. No obstante, se han configurado las siguientes políticas de seguridad:

- **Orígenes permitidos**: Se han especificado explícitamente los orígenes autorizados para realizar peticiones al *backend*, incluyendo tanto `localhost` (para desarrollo) como la URL de producción.
- **Métodos HTTP permitidos**: Se han habilitado los métodos GET, POST, PUT, DELETE y OPTIONS.
- **Headers autorizados**: Se permite el envío de los headers `Authorization` (para tokens JWT), `Content-Type` y `Accept`.
- **Gestión de credenciales**: Se ha configurado `allow_credentials=True` para permitir el envío de cookies y headers de autenticación.

Adicionalmente, se han implementado las siguientes medidas de seguridad:

- Validación estricta de tokens JWT, verificando firma, expiración y existencia del usuario.
- Protección contra inyección SQL mediante el uso de ORM (SQLAlchemy) y consultas parametrizadas.
- Limitación del tamaño máximo de las peticiones HTTP para evitar ataques de denegación de servicio.

4.7.8. Actualizaciones y despliegue continuo

El sistema se ha configurado para implementar un flujo de despliegue continuo:

- Cada vez que se realiza un commit en la rama principal del repositorio Git, Railway detecta automáticamente el cambio.
- Se ejecuta un proceso de compilación que instala las dependencias actualizadas.
- Si la compilación es exitosa, se despliega automáticamente la nueva versión del sistema.
- El despliegue se realiza sin tiempo de inactividad (zero-downtime deployment).

Este enfoque ha permitido iterar de forma ágil durante la fase de desarrollo y validación, incorporando mejoras y correcciones de forma rápida y segura.

4.7.9. Consideraciones de escalabilidad

El sistema desarrollado ha sido desplegado en un entorno de validación funcional que permite verificar su correcto funcionamiento con un volumen de usuarios limitado. No obstante, el diseño arquitectónico adoptado se ha concebido teniendo en cuenta su posible evolución

hacia un escenario de producción a mayor escala. Esta perspectiva de crecimiento progresivo ha guiado diversas decisiones técnicas tomadas durante el desarrollo, desde la elección de tecnologías hasta la organización modular del código.

En este contexto, resulta relevante analizar tanto las consideraciones de escalabilidad que permitirían adaptar el sistema a una demanda creciente, como los desafíos técnicos encontrados durante el proceso de despliegue, que han evidenciado la complejidad inherente a la transición de un entorno de desarrollo local a un entorno de producción real.

En el caso de que el sistema requiriera atender un volumen significativamente mayor de usuarios concurrentes o de transformaciones diarias, las siguientes estrategias permitirían escalar su capacidad sin comprometer la estabilidad ni los tiempos de respuesta:

- **Escalabilidad horizontal del backend**: El sistema actual se ejecuta en una única instancia del servicio FastAPI. En un escenario de alta demanda, sería necesario desplegar múltiples instancias del backend detrás de un balanceador de carga (como NGINX o servicios gestionados tipo AWS ALB), permitiendo distribuir las peticiones entrantes entre varias réplicas del servicio y aumentar la capacidad de procesamiento paralelo.
- **Sistema de caché distribuido**: Para reducir el número de llamadas a APIs externas y disminuir los tiempos de respuesta, resultaría recomendable implementar una capa de caché (mediante Redis o Memcached) que almacene transformaciones recientes. Dado que muchos enunciados educativos se reutilizan en distintos contextos, esta estrategia podría reducir significativamente los costes operativos asociados al uso de modelos de IA.
- **Limitación de tasa por usuario (*rate limiting*)**: En un entorno de producción abierto sería necesario limitar el número de peticiones que un usuario puede realizar por unidad de tiempo, evitando abusos del sistema y garantizando un acceso equitativo a los recursos. Esta limitación podría implementarse mediante middlewares como SlowAPI o servicios gestionados de limitación de tasa.
- **Procesamiento asíncrono mediante colas**: Las transformaciones que requieren múltiples llamadas a servicios externos (Claude, Gemini, ARASAAC) podrían gestionarse de forma asíncrona mediante un sistema de colas (como Celery con Redis o RabbitMQ), permitiendo desacoplar la recepción de peticiones de su procesamiento efectivo y mejorando la experiencia de usuario mediante notificaciones de progreso.
- **Optimización de la base de datos**: A medida que el volumen de transformaciones almacenadas crezca, sería necesario implementar estrategias de indexación avanzada, particionado de tablas y archivado de datos históricos para mantener el rendimiento de las consultas.
- **Monitorización y observabilidad avanzadas**: Resultaría imprescindible implementar herramientas de monitorización más sofisticadas (como Prometheus para métricas, Grafana para visualización, o servicios de APM como New Relic o Datadog) que permitan detectar cuellos de botella, analizar patrones de uso y anticipar problemas de rendimiento antes de que afecten a los usuarios.

Estas estrategias de escalabilidad no han sido implementadas en la versión actual del sistema por tratarse de un Trabajo Fin de Grado con alcance y recursos limitados, pero el diseño arquitectónico modular adoptado facilita su incorporación progresiva sin necesidad de reestructuraciones profundas del código.

A su vez, el proceso de despliegue del sistema en el entorno de producción de Railway ha permitido identificar diversos desafíos técnicos que ilustran la complejidad de la transición de un entorno de desarrollo local a un entorno de producción real. Estos desafíos, si bien han requerido esfuerzo adicional para su resolución, han constituido una experiencia de aprendizaje valiosa sobre las particularidades de los despliegues en la nube:

- **Configuración de OAuth en producción**: El ajuste de las URLs autorizadas de redirección y la gestión de certificados SSL para la autenticación con Google OAuth ha requerido varias iteraciones. Las URLs de callback deben configurarse de forma exacta en la consola de Google Cloud, y cualquier discrepancia entre las URLs configuradas y las utilizadas por la aplicación provoca fallos de autenticación difíciles de diagnosticar sin una documentación exhaustiva del proceso.
- **Compatibilidad de dependencias de Python**: Algunos paquetes Python han presentado conflictos de compatibilidad entre versiones, especialmente en lo relativo a las dependencias transitivas de bibliotecas de procesamiento de imágenes y comunicación con APIs externas. Esto ha requerido ajustes iterativos en `requirements.txt`, fijando versiones específicas de ciertos paquetes para garantizar la reproducibilidad del entorno.
- **Diferencias entre entornos de desarrollo y producción**: Aspectos que funcionaban correctamente en local (como rutas de archivos, configuración de CORS, o gestión de sesiones) han requerido ajustes para funcionar correctamente en Railway. Estas diferencias ponen de manifiesto la importancia de trabajar desde etapas tempranas con entornos de desarrollo lo más similares posible a producción, mediante el uso de contenedores Docker o herramientas de replicación de entornos.
- **Gestión segura de secretos y claves de API**: La organización y gestión segura de las múltiples claves de API necesarias (Claude, Google Gemini, Google OAuth, base de datos PostgreSQL) ha requerido especial atención. Railway facilita la gestión de variables de entorno, pero es fundamental documentar correctamente qué variables son necesarias, su formato esperado y dónde obtenerlas, para evitar errores de configuración que pueden ser difíciles de diagnosticar.
- **Tiempos de arranque y despliegue**: Los tiempos de construcción de la imagen Docker y despliegue del servicio en Railway han resultado más largos de lo inicialmente previsto, especialmente durante las primeras iteraciones con ajustes frecuentes. Esto ha evidenciado la importancia de optimizar las imágenes Docker (mediante cachés de dependencias y construcciones incrementales) para agilizar los ciclos de desarrollo-despliegue.

En síntesis, las dificultades técnicas surgidas durante el despliegue han constituido una parte relevante del proceso de aprendizaje, poniendo de manifiesto la importancia de trabajar desde fases tempranas en entornos reproducibles similares a producción, así como de documentar adecuadamente la configuración de servicios externos y la gestión de secretos. Asimismo, se ha evidenciado el valor de diseñar arquitecturas modulares y desacopladas que faciliten el despliegue y permitan una futura escalabilidad sin reestructuraciones profundas. En conjunto, el despliegue se confirma como una fase crítica del ciclo de vida del software que requiere planificación, atención específica y una clara orientación a la reproducibilidad y la documentación.

4.8. PLAN DE PRUEBAS Y VALIDACIÓN

El plan de pruebas se ha estructurado en dos vertientes complementarias: validación técnica del sistema y validación pedagógica del formato LEdui. La primera garantiza el correcto funcionamiento del software desarrollado, mientras que la segunda verifica la coherencia pedagógica y utilidad práctica de las adaptaciones generadas.

4.8.1. Validación técnica del sistema

La validación técnica se ha centrado en garantizar que todos los componentes del sistema funcionan correctamente, que la integración con servicios externos es estable y que el rendimiento es adecuado para un entorno de uso real.

4.8.1.1. Pruebas de integración con servicios externos

Se han realizado pruebas específicas de cada uno de ellos:

1. **Integración con Claude API**: se ha verificado la correcta comunicación con la API de Anthropic, validando el envío de prompts, la recepción de respuestas estructuradas y el consumo de tokens. Se ha comprobado el manejo de errores ante fallos temporales del servicio y el comportamiento del sistema con distintos tamaños de entrada.
2. **Integración con ARASAAC.** Se ha validado la búsqueda de pictogramas a partir de términos en español, el manejo de situaciones donde no existen pictogramas disponibles, la descarga y composición de imágenes, y el rendimiento ante múltiples peticiones secuenciales.
3. **Integración con Google Gemini:** se ha verificado la generación de imágenes coherentes con el contenido educativo, la interpretación correcta de los prompts generados, el formato y resolución de las imágenes (16:9), y el tiempo de generación.

Los 3 servicios se han comportado de forma más que estable en las pruebas y apenas han dado fallos.

4.8.1.2. Pruebas de persistencia y gestión de datos

Se han realizado pruebas sobre la base de datos PostgreSQL centradas en: creación de registros de usuarios mediante Google OAuth sin duplicados, almacenamiento completo de transformaciones con sus metadatos, asociación correcta entre transformaciones y recursos multimedia, recuperación eficiente del historial con paginación, y funcionamiento de las descargas en formato Word. Asimismo, se han ejecutado pruebas de concurrencia simulando acceso simultáneo de múltiples usuarios. No se han producido problemas reseñables.

4.8.1.3. Pruebas de autenticación y control de acceso

Se ha validado que solo usuarios autenticados pueden acceder a endpoints protegidos, que los tokens JWT expiran correctamente tras 24 horas, que los usuarios no pueden acceder a transformaciones ajenas, y que solo administradores pueden acceder al panel de administración. El funcionamiento ha sido el adecuado siguiendo el estándar marcado por Google.

4.8.1.4. Pruebas de estabilidad y rendimiento

Se han evaluado los tiempos de respuesta ante transformaciones de distintos tamaños, el comportamiento ante peticiones concurrentes, la gestión de recursos del servidor, y la recuperación ante fallos temporales de servicios externos. Los resultados han sido satisfactorios,

observándose tiempos de respuesta medios inferiores a 30 segundos para transformaciones completas, considerados aceptables para un sistema de este tipo.

4.8.2. Validación pedagógica del formato

La validación pedagógica se ha orientado a verificar que los contenidos generados son adecuados, útiles y alineados con los principios de inclusión educativa que fundamentan el formato LEdui.

4.8.2.1. Metodología y diseño de la validación

La validación se ha llevado a cabo mediante consulta a un panel de **cinco expertos con perfiles complementarios**:

- Tres docentes de Formación Profesional con años de experiencia en distintas especialidades.
- Un profesional del sector editorial educativo especializado en inclusión.
- Un orientador educativo especializado en atención al alumnado con necesidades educativas especiales.

El panel ha evaluado 15 enunciados representativos de distintas áreas de conocimiento y niveles educativos (ESO, FP y Universidad), distribuidos entre distintos validadores. Cada enunciado ha sido evaluado por al menos dos validadores, permitiendo contrastar percepciones. Los enunciados completos se incluyen en el Anexo B. A modo ilustrativo, a continuación se muestran varios enunciados de prueba utilizados:

Enunciado Educación Secundaria - Ciencias Sociales 4º ESO: *Examina críticamente las siguientes fuentes primarias sobre la Revolución Industrial: extractos del informe parlamentario británico sobre condiciones de trabajo infantil (1842), testimonios de obreros recogidos por Friedrich Engels y estadísticas demográficas de Manchester entre 1750 y 1850. Contrasta la información proporcionada por cada tipo de fuente, evalúa su fiabilidad considerando la autoría y el contexto histórico de producción, e infiere las transformaciones socioeconómicas que reflejan. Elabora una síntesis argumentada de 400-500 palabras que responda a la pregunta: ¿En qué medida la industrialización mejoró o empeoró las condiciones de vida de la clase trabajadora? Fundamenta tu posición con referencias específicas a las fuentes analizadas.*

Problemáticas identificadas:

- Vocabulario histórico técnico (*fuentes primarias*, *fiabilidad*, *transformaciones socioeconómicas*)
- Múltiples operaciones cognitivas sin secuenciación (examinar, contrastar, evaluar, inferir, sintetizar)
- Complejidad sintáctica elevada
- No explica cómo contrastar fuentes
- Falta estructura de pasos
- No especifica tiempo estimado
- Pregunta abierta compleja sin andamiaje

Enunciado Formación Profesional - Grado Superior Electrónica: *Proyecta un sistema domótico integral para una vivienda de 120m² que integre control de iluminación mediante*

DALI, climatización con termostatos inteligentes Zigbee, persianas motorizadas con KNX, sistema de alarma con sensores cableados e inalámbricos, y cámaras IP con almacenamiento NVR. El sistema debe centralizarse en un servidor domótico (Home Assistant o similar) accesible mediante app móvil. Elabora el proyecto técnico completo: planos de ubicación de dispositivos, cableado estructurado, cuadro de integración, configuración de automatizaciones básicas (escenas, horarios) y manual de uso para cliente. Incluye justificación de compatibilidad entre protocolos y estimación económica detallada.

Problemáticas identificadas:

- Protocolos técnicos sin explicación (*DALI*, *Zigbee*, *KNX*, *NVR*)
- Gran cantidad de elementos sin organización clara
- Múltiples entregables técnicos
- No especifica herramientas de diseño requeridas
- Falta tiempo estimado
- Complejidad técnica elevada sin gradación

Por otro lado, los **instrumentos de evaluación utilizados** (rúbrica y cuestionarios) se detallan en el Anexo C. Esta aproximación metodológica resulta coherente con la literatura en evaluación heurística de sistemas educativos, que establece que entre 3 y 5 evaluadores expertos son suficientes para identificar entre el 75 % y el 85 % de los problemas de usabilidad y coherencia pedagógica en fases iniciales de validación cualitativa [18].

4.8.2.2. Resultados de la validación

Los resultados de la validación pedagógica han sido, en términos generales, positivos. La mayoría de los enunciados adaptados han recibido valoraciones de "bueno" o "excelente" en los criterios relacionados con la claridad lingüística, la estructuración de la información y la explicitación de objetivos.

Como **aspectos positivos** destacan:

- **Coherencia estructural:** la presencia de una estructura fija y predecible facilita enormemente la comprensión de las tareas por parte del alumnado.
- **Eliminación de ambigüedades:** los enunciados adaptados reducen significativamente la inseguridad inicial que muchos estudiantes experimentan al enfrentarse a una actividad nueva.
- **Utilidad de las explicaciones pedagógicas:** valoración especialmente positiva de la identificación de las reglas LEdui aplicadas y las recomendaciones orientadas a la mejora de futuros enunciados.
- **Adecuación de los pictogramas:** representan de forma clara las acciones principales y pueden servir como apoyo visual efectivo para el alumnado con dificultades de comprensión lectora.

Como **aspectos negativos** destacan:

- **Ajuste del tono según nivel educativo:** en algunos casos, el tono resulta excesivamente cercano para Educación Universitaria, donde se esperaría un registro ligeramente más académico sin perder claridad.

- **Calidad de las imágenes de contexto:** aunque en general adecuadas, en ocasiones resultan demasiado genéricas o no aportan información adicional relevante más allá del texto adaptado.
- **Simplificación de vocabulario técnico:** en enunciados muy especializados, la simplificación puede resultar insuficiente o imprecisa, generando explicaciones que, aunque más accesibles, no transmiten con exactitud el concepto técnico original.

Como resultado de esta retroalimentación, se implementaron ajustes en los prompts del sistema para adaptar el registro lingüístico según el nivel educativo, mejorar la especificidad de las imágenes generadas y refinar la identificación de acciones clave para la generación de pictogramas.

4.8.2.3. Limitaciones de la validación

La validación realizada presenta dos limitaciones principales que delimitan el alcance de las conclusiones obtenidas.

En primer lugar, el tamaño del panel de expertos (cinco profesionales) responde a una aproximación de validación cualitativa apropiada para un TFG, pero insuficiente para generalizar resultados a escala. No obstante, la diversidad de perfiles consultados permite cubrir los principales ámbitos de aplicación del formato LEdui y detectar problemas críticos de diseño.

En segundo lugar, y de forma más relevante, la ausencia de validación empírica con alumnado real constituye la principal limitación del trabajo. Esta carencia responde a restricciones temporales y administrativas inherentes al alcance de un TFG: la validación con menores requiere aprobaciones éticas, consentimientos informados y un período mínimo de aplicación en aula que excede los plazos disponibles. La validación con expertos permite verificar la coherencia pedagógica y viabilidad del formato, pero no medir su impacto real en la comprensión, motivación y autonomía del alumnado. Esta validación empírica se plantea como línea de trabajo futuro en el capítulo 5.

4.8.3. Síntesis del plan de pruebas

El plan de pruebas realizado permite afirmar que el sistema desarrollado cumple con los requisitos técnicos y pedagógicos establecidos, demostrando su viabilidad como herramienta de apoyo al profesorado para la adaptación de contenidos educativos al formato LEdui.

La validación técnica ha confirmado la estabilidad del sistema, su capacidad para integrarse correctamente con servicios externos y su adecuado comportamiento en condiciones de uso real. Los tiempos de respuesta son aceptables para un contexto educativo, y la arquitectura modular facilita futuras ampliaciones.

La validación pedagógica ha evidenciado que los enunciados adaptados cumplen de forma efectiva con los principios de claridad, estructuración y accesibilidad cognitiva que fundamentan el formato LEdui. La retroalimentación recibida por parte de profesionales del ámbito educativo ha sido mayoritariamente positiva, destacando especialmente la utilidad práctica del sistema y su potencial para reducir barreras cognitivas en el acceso al aprendizaje.

CAPÍTULO 5
Conclusiones

Este capítulo recoge las conclusiones del Trabajo Fin de Grado, sintetizando los principales resultados obtenidos. Se evalúa el grado de cumplimiento de los objetivos y requisitos del sistema desarrollado, se justifica la aplicación de las competencias adquiridas durante su realización y se presenta una valoración personal del trabajo realizado y su aportación.

5.1. OBJETIVOS ALCANZADOS

En esta sección se presenta el grado de cumplimiento de los objetivos definidos en el Capítulo 2, analizando de forma diferenciada el objetivo general y los objetivos complementarios del Trabajo Fin de Grado.

5.1.1. Objetivo general (OG)

El objetivo general marcado es:

El objetivo general de este Trabajo Fin de Grado es el desarrollo de un servicio web orientado a la transformación automática de contenidos educativos textuales en formato de Lectura Educativa Inclusiva (LEdui), mediante el uso de inteligencia artificial.

Este objetivo se ha alcanzado completamente mediante el desarrollo de un sistema funcional que integra arquitectura cliente–servidor, API REST y orquestación de múltiples servicios de inteligencia artificial. El cumplimiento se evidencia en:

- La **formalización operativa del formato LEdui** en diez reglas aplicables computacionalmente, traduciendo principios pedagógicos abstractos en especificaciones verificables por un sistema automático.
- La **implementación de una API REST** que expone endpoints funcionales para autenticación, transformación de contenidos, gestión de usuarios y recuperación de historial, permitiendo la integración del sistema en otros entornos educativos.
- La **orquestación exitosa de servicios externos** (Claude API, Gemini API, ARASAAC) que permite generar adaptaciones textuales, imágenes de contexto y pictogramas de forma coordinada y coherente.
- El **determinismo estructural** del sistema, que garantiza que todas las transformaciones mantengan una estructura HTML fija con secciones pedagógicas predefinidas, independientemente del contenido específico generado.

La validez del sistema se ha verificado mediante pruebas funcionales y validación con profesionales de la educación, confirmando que las transformaciones generadas son coherentes con los principios del formato LEdui.

5.1.2. Objetivos complementarios

A continuación se analiza el cumplimiento de los objetivos complementarios de forma individual, explicando qué se ha logrado y cómo.

OC1. *Identificar y analizar los principales marcos normativos y estándares relacionados con la educación inclusiva y la accesibilidad cognitiva.*

Se ha realizado un análisis sistemático de cinco marcos normativos de referencia (DUA, Lectura Fácil, WCAG, UNE 71362:2020 y guías de accesibilidad cognitiva), identificando principios comunes que han servido como base conceptual para la definición de las reglas LEdui. Este análisis no solo ha compilado información, sino que ha establecido una **metodología de convergencia** que permite justificar cada regla LEdui mediante su alineación con los estándares consolidados analizados.

OC2. *Definir formalmente el formato de Lectura Educativa Inclusiva (LEdui), especificando un conjunto estructurado de reglas aplicables de manera sistemática.*

El formato LEdui se ha definido mediante **diez reglas operativas** con dos características clave que garantizan su aplicabilidad: (1) están formuladas en lenguaje comprensible tanto para docentes como para sistemas computacionales; y (2) se aplican de forma complementaria y no excluyente, evitando conflictos mediante estrategias sinérgicas. La validación inversa realizada posteriormente confirma que las reglas propuestas no contradicen ninguno de los marcos analizados, a su vez, la conexión con perfiles de necesidades educativas especiales más habituales ha demostrado que las reglas cubren transversalmente un amplio espectro de las mismas.

OC3. *Diseñar un modelo de transformación automática de contenidos educativos basado en las reglas LEdui.*

Se ha diseñado un modelo que traduce las reglas LEdui en un proceso sistemático mediante **prompting estructurado**. El modelo impone una plantilla HTML fija que garantiza la presencia de todas las secciones pedagógicas esenciales (objetivos, pasos, criterios de entrega, reflexión, tiempo estimado) sin recurrir a post-procesamiento complejo. Esta aproximación ha demostrado ser efectiva para conseguir determinismo estructural manteniendo la naturalidad expresiva del contenido generado.

OC4. *Implementar un servicio web que materialice el modelo de transformación definido, exponiendo una API funcional.*

El servicio se ha implementado mediante FastAPI siguiendo una **arquitectura modular en tres capas** (presentación, lógica de negocio, persistencia) que facilita el mantenimiento y la escalabilidad. La separación de responsabilidades permite que el sistema pueda integrarse en otros entornos (LMS, aplicaciones móviles) sin reescribir la lógica de adaptación. La decisión de exponer funcionalidad mediante API REST en lugar de una aplicación monolítica ha resultado estratégica para la flexibilidad del sistema, facilitando su conexión con terceros sistemas si fuera necesario.

OC5. *Integrar al servicio web las funcionalidades necesarias para la generación de apoyos visuales mediante la incorporación de pictogramas y recursos multimedia asociados al contenido textual adaptado, garantizando su coherencia semántica.*

Se ha logrado integrar la generación automática de pictogramas mediante ARASAAC y de imágenes de contexto mediante Gemini API. El sistema extrae automáticamente acciones clave del enunciado adaptado y busca pictogramas coherentes, mientras que genera prompts específicos para crear imágenes educativas relacionadas con el contenido.

OC6. *Desarrollar una interfaz web que permita al usuario interactuar con el sistema y visualizar los contenidos adaptados.*

Se ha desarrollado una interfaz funcional implementada en HTML, CSS y JavaScript que permite realizar transformaciones, visualizar resultados con formato pedagógico claro, gestionar historial de transformaciones y descargar contenidos en formato DOCX. El diseño prioriza la **reducción de carga cognitiva** mediante navegación clara, feedback visual inmediato y organización predecible de la información. La interfaz incluye autenticación mediante Google OAuth 2.0 y un panel de administración para gestión de usuarios y transformaciones.

OC7. *Evaluar el funcionamiento del sistema mediante casos de uso representativos.*

El sistema ha sido evaluado mediante dos aproximaciones complementarias: (1) **pruebas funcionales técnicas** que validan el correcto funcionamiento de todos los endpoints, la persistencia de datos y la integración con servicios externos; y (2) **validación pedagógica cualitativa** con cinco profesionales especializados (docentes de FP, editorial educativa y orientador) que han verificado la coherencia de las adaptaciones generadas con las necesidades reales del aula. La evaluación técnica confirma la estabilidad del sistema, mientras que la validación experta confirma su viabilidad pedagógica.

OC8. *Analizar el rendimiento y la escalabilidad del sistema desarrollado, evaluando su idoneidad para un uso real en escenarios de concurrencia y crecimiento progresivo del número de usuarios.*

Se ha evaluado el rendimiento del sistema en condiciones reales de uso, obteniendo tiempos de respuesta inferiores a 30 segundos para transformaciones completas (texto + pictogramas + imagen de contexto). Este tiempo se considera **aceptable para un entorno educativo** donde la preparación de materiales no requiere inmediatez, y es significativamente inferior al tiempo que requeriría una adaptación manual equivalente (estimado en 20-45 minutos por enunciado).

Desde el punto de vista de la escalabilidad, la arquitectura desarrollada presenta características que facilitan el crecimiento: (1) la **separación en tres capas** permite escalar horizontalmente la capa de lógica de negocio mediante despliegue de múltiples instancias del *backend*; (2) el uso de PostgreSQL como base de datos relacional garantiza capacidad de gestión de grandes volúmenes de transformaciones; (3) la integración mediante APIs externas evita cuellos de botella en la generación de contenido, delegando el procesamiento intensivo a servicios especializados.

No obstante, se han identificado dos aspectos que requerirían atención en escenarios de uso masivo: (1) el **coste económico de las APIs externas** (Claude y Gemini) crecería

linealmente con el número de transformaciones; y (2) la **dependencia de servicios externos** introduce un riesgo de disponibilidad que podría mitigarse mediante estrategias de proveedores alternativos.

5.1.3. Cumplimiento de requisitos del sistema

El análisis del cumplimiento de los objetivos permite valorar el grado de éxito del trabajo desde una perspectiva funcional y conceptual. Como complemento a dicho análisis, resulta necesario realizar un aterrizaje más técnico, evaluando el grado de cumplimiento de los requisitos del sistema definidos durante la fase de planificación, lo que permite contrastar de forma directa la solución desarrollada con las especificaciones inicialmente establecidas.

En la fase de definición de requisitos se estableció una distinción clara entre los requisitos asociados al Producto Mínimo Viable (MVP) y un conjunto de requisitos deseables, concebidos como posibles ampliaciones a abordar en iteraciones posteriores del sistema.

En relación con los **requisitos del MVP,** el grado de cumplimiento puede considerarse muy alto. El sistema desarrollado satisface plenamente los requisitos esenciales definidos para validar la viabilidad técnica y funcional del formato LEdui, incluyendo la transformación automática de contenidos educativos, la generación de una estructura fija y predecible, la exposición de una API funcional, la persistencia de resultados y la interacción básica mediante una interfaz web. Estos elementos permiten afirmar que el núcleo del sistema es plenamente operativo, estable y coherente con los objetivos técnicos planteados desde el inicio del trabajo.

Por lo que respecta a los **requisitos deseables**, el sistema ha alcanzado un grado de cumplimiento significativamente superior al mínimo exigido para un Trabajo Fin de Grado. A lo largo del desarrollo iterativo se han incorporado funcionalidades que inicialmente se consideraban opcionales o de ampliación, como la integración de apoyos visuales mediante pictogramas e imágenes de contexto, la generación de explicaciones pedagógicas complementarias para el profesorado, la gestión de usuarios y el almacenamiento del historial de transformaciones realizadas. Estas funcionalidades refuerzan el carácter aplicado, escalable y realista de la solución desarrollada.

Cabe destacar que algunos aspectos deseables de mayor complejidad, como la personalización avanzada por perfil de usuario, la validación empírica con alumnado real o la realización de auditorías formales completas de accesibilidad, no se han abordado en profundidad, al considerarse fuera del alcance temporal y metodológico de este Trabajo Fin de Grado. No obstante, su identificación permite definir de manera clara líneas de evolución futura del sistema.

En conjunto, el cumplimiento de los requisitos definidos puede considerarse muy satisfactorio, habiéndose alcanzado de forma completa los requisitos del MVP y superado parcialmente los requisitos deseables, lo que evidencia una correcta planificación, un desarrollo iterativo eficaz y una adecuada alineación entre las especificaciones técnicas iniciales y la solución final implementada.

5.2. JUSTIFICACIÓN DE COMPETENCIAS ADQUIRIDAS

De acuerdo con lo establecido en la normativa del Trabajo Fin de Grado del Grado en Ingeniería Informática, esta sección justifica las competencias específicas de la intensificación en Computación adquiridas durante el desarrollo del proyecto. En conjunto, el desarrollo de este Trabajo Fin de Grado ha permitido adquirir y aplicar de forma sólida las competencias específicas de la intensificación en **Computación**, integrando fundamentos teóricos y técnicos de la computación con una aplicación práctica orientada a la resolución de un problema real en el ámbito educativo y de la accesibilidad. El proyecto evidencia una adecuada madurez técnica y conceptual acorde con el nivel exigido para la finalización del grado.

A continuación, se detallan aquellas competencias que han sido trabajadas de forma directa y significativa en este TFG, relacionándolas con las tareas y resultados obtenidos.

CM1. *Conocimiento y aplicación de los principios y modelos de la computación.*

Esta competencia se ha adquirido mediante el análisis, diseño y creación de un nuevo formato computacional, LEdui, orientado a la adaptación de contenidos educativos. El proyecto no se limita a aplicar soluciones existentes, sino que modela un conjunto de reglas, estructuras y flujos de transformación que permiten formalizar principios pedagógicos y de accesibilidad en un sistema computable. Asimismo, se han valorado y comparado distintos estándares y modelos, seleccionando y combinando aquellos más adecuados para el problema abordado.

CM2. *Fundamentos de lenguajes y procesamiento léxico, sintáctico y semántico.*

Durante el desarrollo del sistema se ha trabajado de forma explícita con el procesamiento semántico del lenguaje natural, utilizando modelos de lenguaje para analizar, transformar y generar textos educativos. El diseño de prompts estructurados, la normalización de entradas textuales y la generación de salidas con una estructura HTML fija implican una comprensión práctica de los niveles léxico, sintáctico y semántico del lenguaje, así como de su representación computacional.

CM3. *Evaluación de la complejidad computacional y estrategias algorítmicas.*

Esta competencia se ha abordado en el diseño del flujo de transformación del contenido educativo, que se ha concebido como una orquestación secuencial de endpoints. Se han tomado decisiones orientadas a optimizar el tiempo de respuesta del sistema, el número de llamadas a servicios externos y el coste computacional asociado al uso de modelos de Inteligencia Artificial, evaluando diferentes alternativas y descartando aquellas menos eficientes.

CM4. *Fundamentos y aplicación de sistemas inteligentes.*

El proyecto hace un uso intensivo y fundamentado de técnicas propias de los sistemas inteligentes, integrando modelos de lenguaje de gran tamaño y modelos de generación de imágenes a través de APIs. Se han diseñado servicios capaces de analizar contenido, generar texto estructurado, extraer información semántica y producir recursos visuales, aplicando estos paradigmas en el ámbito educativo y de la accesibilidad.

CM5. *Representación computable del conocimiento humano.*

LEdui constituye una formalización computable de conocimiento pedagógico y normativo relacionado con la accesibilidad y la inclusión educativa. Las reglas del formato, los estándares analizados y las necesidades educativas detectadas se han traducido a estructuras, procesos y flujos de decisión que pueden ser ejecutados por un sistema informático. Esta competencia se manifiesta especialmente en la capacidad del sistema para representar objetivos educativos, competencias y apoyos visuales de forma automática.

CM6. *Desarrollo de sistemas interactivos y presentación de información compleja.*

La competencia se ha trabajado mediante el desarrollo de un *frontend* funcional que permite la interacción con el sistema y la visualización de contenidos educativos adaptados. Un proceso complejo conceptualmente, pero que se ha buscado el enfoque más comprensible y pedagógico posible a la hora de presentarlo. Además, el diseño de la interfaz y del formato de salida HTML tiene en cuenta principios de accesibilidad, claridad y usabilidad, facilitando la comprensión de información compleja por parte de distintos perfiles de usuario.

CM7. *Técnicas de aprendizaje computacional y extracción de conocimiento.*

Aunque el sistema no entrena modelos propios, se ha adquirido esta competencia mediante la integración y uso avanzado de modelos de aprendizaje computacional preentrenados. El proyecto demuestra la capacidad de seleccionar modelos adecuados, evaluar su comportamiento, comparar alternativas y utilizarlos para la extracción automática de información, generación de conocimiento y apoyo a la toma de decisiones dentro de un sistema informático.

En síntesis, el desarrollo de este Trabajo Fin de Grado ha permitido integrar de forma cohesionada las siete competencias específicas de la intensificación en Computación, demostrando que la resolución de problemas complejos en ámbitos aplicados requiere la confluencia de múltiples dimensiones de la disciplina. La capacidad de modelar conocimiento pedagógico en estructuras computables (CM1, CM5), procesarlo mediante técnicas avanzadas de lenguaje natural (CM2) e inteligencia artificial (CM4, CM7), optimizar su ejecución (CM3) y presentarlo de forma accesible e interactiva (CM6) constituye el resultado natural de una formación integral en computación. El proyecto evidencia, en consecuencia, no solo la adquisición individual de cada competencia, sino también la madurez técnica necesaria para articularlas de manera efectiva en la construcción de sistemas informáticos con impacto real en contextos profesionales y sociales.

5.3. TRABAJOS DERIVADOS Y FUTUROS

El trabajo desarrollado en este Trabajo Fin de Grado establece una base conceptual y tecnológica sólida para el formato de Lectura Educativa Inclusiva (LEdui) y para el sistema inteligente asociado. No obstante, debido a las limitaciones temporales y metodológicas propias de un TFG, existen diversas líneas de evolución y trabajo futuro que no han podido abordarse en profundidad.

1. **Evaluación empírica del formato LEdui:** diseño y ejecución de estudios académicos sistemáticos en contextos educativos reales, utilizando el sistema desarrollado como herramienta de transformación. Estos estudios permitirían analizar el impacto del formato en variables como la comprensión de enunciados, la reducción de bloqueos iniciales, la autonomía del alumnado o la percepción de claridad por parte del profesorado. Estudios que sin duda podrían ser base de una publicación científica para difundir los trabajos realizados.
2. **Validación de la comunidad educativa:** pruebas de uso prolongadas con docentes de distintos niveles y especialidades, orientadas a validar el formato LEdui en contextos reales de aula. En este sentido, se ha presentado un proyecto de innovación educativa a la Junta de Comunidades de Castilla-La Mancha (JCCM) con el objetivo de que al menos 50 docentes puedan utilizar el sistema y aportar retroalimentación estructurada para su mejora.
3. **Personalización avanzada del sistema:** desarrollo de mecanismos de adaptación del formato en función de perfiles específicos de necesidades educativas especiales, que permitan priorizar reglas manteniendo el enfoque de diseño universal.
4. **Extensión a otros niveles educativos:** adaptación del formato LEdui y del sistema de transformación automática a otros niveles educativos, como Educación Primaria o Educación Infantil, que presentan requisitos específicos en términos de lenguaje, estructura y apoyos multimodales.

Con el fin de dimensionar el alcance de las principales líneas de trabajo futuro, en la Tabla 5.1 se presenta una estimación orientativa del esfuerzo de desarrollo requerido para cada una de ellas. Estas estimaciones se basan en la complejidad técnica y metodológica de cada línea y asumen dedicación a tiempo completo.

Tabla 5.1: Resumen de estimaciones de desarrollo de trabajos futuros

Línea de trabajo futuro	Horas estimadas	Semanas (t. completo)
Estudio académico sistemático	560	8–15
Validación con comunidades educativas	225	5–6
Personalización por perfiles NEE	230	6
Adaptación a otros niveles educativos	270	6–7
TOTAL	**1285**	**25–34**

Estas líneas de trabajo futuro refuerzan el carácter de LEdui como una propuesta abierta, evolutiva y susceptible de investigación académica, en la que el sistema inteligente desarrollado actúa no solo como una herramienta tecnológica, sino como un medio para estudiar y mejorar la aplicación práctica de principios de inclusión educativa en contextos reales de enseñanza-aprendizaje. La planificación detallada de los aspectos técnicos y las estimaciones de coste evidencian que este TFG constituye una base sólida y escalable para un programa de investigación de mayor envergadura, con potencial de impacto real en la práctica educativa.

5.4. VALORACIÓN PERSONAL

Este Trabajo Fin de Grado no ha sido únicamente un ejercicio académico ni un reto tecnológico. Ha sido, ante todo, un proceso profundamente personal, en el que se han entrelazado la experiencia docente, la vocación educativa y la convicción de que la tecnología solo adquiere pleno sentido cuando se pone al servicio de las personas.

A lo largo de este trabajo se ha abordado una problemática que, aunque a menudo pasa desapercibida, condiciona de forma decisiva la experiencia educativa de muchos alumnos: la dificultad para comprender qué se les pide y cómo deben afrontar una tarea. Para el alumnado con necesidades educativas especiales, esta dificultad no constituye un obstáculo puntual, sino una barrera persistente que limita su capacidad para mostrar todo aquello que realmente sabe hacer.

Desde la experiencia cotidiana en el aula, y también desde una vivencia personal íntimamente ligada a la educación inclusiva, este trabajo ha nacido de una certeza clara: muchos bloqueos educativos no tienen su origen en la falta de capacidad, sino en la forma en que se presentan los retos. Cuando las expectativas se hacen explícitas y se ofrecen apoyos adecuados, el aprendizaje deja de ser una fuente de frustración y se convierte en una oportunidad de crecimiento. LEdui ha pretendido precisamente eso: tender puentes entre el conocimiento del alumnado y su expresión real.

El desarrollo de un sistema orientado a ayudar al profesorado a crear contenidos más inclusivos ha permitido reafirmar una convicción profunda: que la educación debe ser una auténtica palanca de oportunidades para toda la sociedad. Una educación que no limite, que no etiquete y que, en ningún caso, excluya.

Este trabajo no ha pretendido ofrecer una solución definitiva. Ha buscado, simplemente, mostrar que es posible utilizar la tecnología con responsabilidad, sensibilidad pedagógica y compromiso social. Si LEdui consigue que un alumno se enfrente a una tarea con menos miedo, con mayor confianza o con más autonomía, el esfuerzo invertido habrá merecido con creces la pena.

En última instancia, este Trabajo Fin de Grado ha representado la culminación de un camino largo y profundamente significativo: un camino que reafirma que construir un mundo educativo más justo y más humano no es solo una aspiración teórica, sino una responsabilidad compartida que también puede ejercerse desde la ingeniería y la docencia.

Bibliografía

[1] Asociación Española de Normalización. Une 153101:2018 ex. lectura fácil. pautas y recomendaciones para la elaboración de documentos. URL: https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma/?c=N0060036, 2018. Último acceso 23 dic. 2025.

[2] Asociación Española de Normalización. Une 71362:2020. calidad de los materiales educativos digitales. URL: https://www.une.org/encuentra-tu-norma/busca-tu-norma/norma?c=N0063263, 2020. Último acceso 23 dic. 2025.

[3] Aula Abierta ARASAAC. Arawrite - conocemos arawrite: Introducción, 2025. Último acceso 20 ene. 2025.

[4] Barry W. Boehm. *A Spiral Model of Software Development and Enhancement*. IEEE Computer Society, Los Alamitos, 1988. IEEE Computer.

[5] Center for Applied Special Technology. Universal design for learning guidelines version 2.2. URL: https://udlguidelines.cast.org, 2018. Último acceso 23 dic. 2025.

[6] Centro Aragonés para la Comunicación Aumentativa y Alternativa. Arasaac: Portal aragonés de la comunicación aumentativa y alternativa. URL: https://arasaac.org, 2025. Último acceso 23 dic. 2025.

[7] Comisión de Familias de Plena inclusión. Guía informativa para familiares de alumnos y alumnas con necesidades educativas especiales. URL: https://www.plenainclusion.org/wp-content/uploads/2024/11/Guia-informativa-para-familiares-Plena-inclusion.pdf, 2024. Último acceso 20 ene. 2025.

[8] Consejería de Educación, Cultura y Deportes de Castilla-La Mancha. Decreto 85/2018, de 20 de noviembre, por el que se regula la inclusión educativa del alumnado en la comunidad autónoma de castilla-la mancha. Diario Oficial de Castilla-La Mancha (DOCM), núm. 229, 23 de noviembre de 2018, 2018. Último acceso 20 ene. 2025.

[9] Cristina Dumitru, Ghaida Muttashar Abdulsahib, Osamah Ibrahim Khalaf, and Akram Bennour. Integrating artificial intelligence in supporting students with disabilities in higher education: An integrative review. *Journal of Educational Technology Systems*, 2025.

[10] Nesren S. Farhah, Asim Wadood, Ahmed Abdullah Alqarni, M. I. Uddin, and T. H. H. Aldhyani. Enhancing adaptive learning with generative ai for tailored educational support for students with disabilities. *Journal of Disability Research*, June 2025. Sistema ALGA-Ed para educación adaptativa.

[11] Ricardo Fitas. Inclusive education with ai: supporting special needs and tackling language barriers. *AI and Ethics*, September 2025.

[12] Google Education Blog. Learn your way from google adapts educational content using ai, September 2025. Último acceso 20 ene. 2025.

[13] Google Research. Learn your way: Reimagining textbooks with generative ai, September 2025. Último acceso 20 ene. 2025.

[14] Wayne Holmes, Maya Bialik, and Charles Fadel. *Artificial Intelligence in Education: Promises and Implications for Teaching and Learning*. Center for Curriculum Redesign, Boston, 2019. ISBN: 9780997252922.

[15] Rose Luckin, Wayne Holmes, Mark Griffiths, and Laurie Forcier. *Intelligence Unleashed: An Argument for AI in Education*. Pearson Education, London, 2016. ISBN: 9781292065365.

[16] Richard E. Mayer. *Multimedia Learning*. Cambridge University Press, Cambridge, second edition, 2009. ISBN: 9780521735355.

[17] Verónica-Alexandra Melo-López, Andrea Basantes-Andrade, Carla-Belén Gudiño-Mejía, and Evelyn Hernández-Martínez. The impact of artificial intelligence on inclusive education: A systematic review. *Education Sciences*, 15(5):539, April 2025.

[18] Jakob Nielsen and Thomas K. Landauer. A mathematical model of the finding of usability problems. *Proceedings of the INTERACT'93 and CHI'93 Conference on Human Factors in Computing Systems*, pages 206–213, 1994.

[19] Organisation for Economic Co-operation and Development. Oecd digital education outlook 2021: Pushing the frontiers with ai, blockchain and robots. URL: https://www.oecd.org/education/digital-education-outlook-2021/, 2021. Último acceso 23 dic. 2025.

[20] Allan Paivio. *Mental Representations: A Dual Coding Approach*. Oxford University Press, New York, 1986. ISBN: 9780195066661.

[21] Plena Inclusión España. ¿qué es la lectura fácil? URL: https://www.plenainclusion.org/discapacidad-intelectual/recurso/lectura-facil/, 2018. Último acceso 23 dice. 2025.

[22] Plena Inclusión España. Accesibilidad cognitiva: hacia una comprensión universal. URL: https://accesibilidadcognitiva.org/documentos/accesibilidad-cognitiva-hacia-una-comprension-universal/, 2021. Último acceso 23 dic. 2025.

[23] Portal Aragonés de la Comunicación Aumentativa y Alternativa. Arawrite - escritura con pictogramas arasaac, 2025. Accedido: 2-febrero-2026.

[24] Roger S. Pressman and Bruce R. Maxim. *Software Engineering: A Practitioner's Approach*. McGraw-Hill Education, New York, 9th edition, 2020. ISBN: 9781259872976.

[25] Red Española de Aprendizaje-Servicio. Aprendizaje-servicio: Portal de la red española de aprendizaje-servicio. URL: https://www.aprendizajeservicio.net/, 2025. Último acceso 20 ene. 2025.

[26] Rewordify. Rewordify.com: Understand what you read, 2024. Accedido: 2-febrero-2026.

[27] David H. Rose and Anne Meyer. *Teaching Every Student in the Digital Age: Universal Design for Learning*. Association for Supervision and Curriculum Development, Alexandria, VA, 2002. ISBN: 9780871205990.

[28] Catherine E. Snow and Paola Uccelli. The challenge of academic language. *The Cambridge Handbook of Literacy*, pages 112–133, 2009.

[29] Ian Sommerville. *Software Engineering*. Pearson Education, Boston, 10th edition, 2016. ISBN: 9780133943030.

[30] United Nations Educational, Scientific and Cultural Organization. Guidance for generative ai in education and research. URL: https://www.unesco.org/en/articles/guidance-generative-ai-education-and-research, 2023. Último acceso 23 dic. 2025.

[31] United Nations Educational, Scientific and Cultural Organization. Global education monitoring report 2024/25: Leadership in education: Lead for learning. URL: https://www.unesco.org/gem-report/en, 2024. Último acceso 23 dic. 2025.

[32] World Wide Web Consortium. Understanding wcag 2.2. URL: https://www.w3.org/WAI/WCAG22/Understanding/, 2023. Último acceso 23 dic. 2025.

[33] World Wide Web Consortium. Web content accessibility guidelines (wcag) 2.2. URL: https://www.w3.org/TR/WCAG22/, 2023. Último acceso 23 dic. 2025.

ANEXOS

ANEXO A

Anexo A. Descripción detallada de las reglas LEdui

Este anexo presenta una descripción exhaustiva de las 10 reglas que conforman el formato de Lectura Educativa Inclusiva (LEdui). Para cada regla se incluye su fundamentación pedagógica, ejemplos de aplicación práctica, conexión con marcos normativos de referencia y consideraciones específicas para su implementación tanto manual como computacional.

A.1. REGLA 1. CLARIDAD LINGÜÍSTICA

Los textos deben redactarse utilizando frases cortas, voz activa y vocabulario cotidiano. Los tecnicismos, siempre que sean necesarios, deben explicarse mediante ejemplos cercanos y comprensibles. El objetivo es facilitar un acceso directo al significado del texto, evitando barreras derivadas de un lenguaje excesivamente complejo o técnico.

A.1.1. Fundamentación pedagógica

La claridad lingüística constituye el pilar básico de cualquier estrategia de accesibilidad cognitiva. Los marcos de inclusión educativa más habituales exponen que la complejidad sintáctica y léxica puede actuar como barrera significativa para el alumnado con dificultades de comprensión lectora, trastornos del lenguaje o procesamiento cognitivo reducido. Usar frases cortas (idealmente de 15-20 palabras como máximo) reduce la carga de la memoria de trabajo, mientras que la voz activa facilita la identificación del agente, la acción y el objeto de forma inmediata.

El vocabulario cotidiano no implica simplificación conceptual, sino accesibilidad comunicativa. Se trata de expresar ideas complejas mediante un lenguaje que resulte familiar al alumnado, reservando los tecnicismos únicamente para aquellos casos en que resulten imprescindibles por razones disciplinares o profesionales. En estos casos, la estrategia del *tecnicismo explicado* (término técnico + paráfrasis sencilla entre paréntesis o inmediatamente después) permite mantener el rigor conceptual sin comprometer la comprensión.

A.1.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

> "Elabora un análisis crítico de las implicaciones epistemológicas que se derivan de la implementación de metodologías ágiles en contextos organizacionales caracterizados por estructuras jerárquicas tradicionales."

Después de aplicar la regla:

"Analiza qué problemas pueden aparecer cuando una empresa con una estructura tradicional (con jefes y niveles jerárquicos claros) intenta usar metodologías ágiles (formas de trabajar más flexibles y en equipo). Reflexiona sobre si estas dos formas de trabajar son compatibles o si generan conflictos."

Ejemplo 2 – Antes de aplicar la regla:

"Configura el enrutador mediante la interfaz CLI (Command Line Interface) ejecutando los comandos pertinentes para establecer el direccionamiento IP de las interfaces correspondientes."

Después de aplicar la regla:

"Configura el router usando la CLI (línea de comandos, la pantalla negra donde escribes instrucciones). Escribe los comandos necesarios para asignar una dirección IP a cada interfaz del router."

A.2. REGLA 2. ESTRUCTURA GUIADA

Los contenidos deben presentarse de forma ordenada y secuencial, preferentemente mediante pasos claros o listas numeradas. Esta estructura guiada permite que cada acción quede claramente identificada, reduciendo la sobrecarga cognitiva y facilitando el seguimiento de las instrucciones por parte del alumnado.

A.2.1. Fundamentación pedagógica

La estructuración secuencial de la información responde a principios de la psicología cognitiva relacionados con la segmentación de tareas complejas (*chunking*). Cuando una actividad se presenta como un bloque de texto continuo, el alumnado debe realizar un esfuerzo cognitivo adicional para identificar qué debe hacer, en qué orden y cómo saber si ha completado cada paso. La presentación mediante pasos numerados o listas explicita esta estructura, externalizando parte del proceso de planificación y permitiendo que el alumnado centre sus recursos cognitivos en la ejecución de la tarea más que en su decodificación.

La estructuración guiada es especialmente beneficiosa para alumnado con dificultades de función ejecutiva, TDAH o trastornos del espectro autista, ya que proporciona predictibilidad y reduce la incertidumbre. Asimismo, facilita la autorregulación al permitir al alumnado monitorizar su propio progreso ("ya he completado el paso 2 de 5").

A.2.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Para crear el informe, primero tendrás que investigar las fuentes proporcionadas, luego redactar un borrador con tus conclusiones, después revisarlo para corregir errores y finalmente entregarlo en formato PDF antes del viernes."

Después de aplicar la regla:

"Sigue estos pasos para crear el informe:

Paso 1: Lee las fuentes que te he proporcionado y toma notas sobre las ideas principales.

Paso 2: Escribe un borrador con tus conclusiones. No te preocupes por los errores en esta fase.

Paso 3: Revisa tu borrador y corrige errores de ortografía, gramática y coherencia.

Paso 4: Convierte tu informe a formato PDF.

Paso 5: Entrégalo antes del viernes a las 23:59h."

Ejemplo 2 – Antes de aplicar la regla:

"Diseña una base de datos relacional considerando las necesidades del cliente, normaliza las tablas hasta la tercera forma normal, implementa las relaciones mediante claves primarias y foráneas, y documenta el modelo mediante un diagrama entidad-relación."

Después de aplicar la regla:

"Diseña una base de datos siguiendo estos pasos:

Paso 1: Analiza las necesidades del cliente e identifica qué información debe almacenar la base de datos.

Paso 2: Diseña las tablas necesarias y normalízalas hasta la tercera forma normal (3FN).

Paso 3: Establece las relaciones entre tablas usando claves primarias y claves foráneas.

Paso 4: Crea un diagrama entidad-relación (E-R) que muestre gráficamente tu diseño."

A.3. REGLA 3. COHERENCIA EXPOSITIVA

El enunciado debe organizarse siguiendo una estructura lógica que incluya introducción, desarrollo y cierre. Asimismo, se recomienda resumir ideas complejas y utilizar conectores textuales que ayuden a mantener la coherencia y la continuidad del discurso, facilitando la retención y comprensión de la información.

A.3.1. Fundamentación pedagógica

La coherencia expositiva responde a principios de la lingüística textual y la teoría de los esquemas cognitivos. Los textos bien estructurados activan esquemas mentales que facilitan la comprensión y retención de la información. Una introducción clara establece expectativas, un desarrollo ordenado permite seguir la progresión temática, y un cierre proporciona síntesis y clausura cognitiva.

Los conectores textuales ("en primer lugar", "por otro lado", "en conclusión") funcionan como señales que guían la lectura, explicitando las relaciones lógicas entre ideas (adición, contraste, causalidad, consecuencia). Para el alumnado con dificultades de comprensión lectora, estos conectores son especialmente valiosos porque reducen la necesidad de inferir relaciones implícitas.

El resumen de ideas complejas al final de secciones extensas facilita la consolidación del aprendizaje y permite al alumnado verificar su comprensión antes de avanzar.

A.3.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Los circuitos serie y paralelo funcionan diferente. En serie, la corriente es la misma. En paralelo, el voltaje es el mismo. Calcula la resistencia total del circuito adjunto."

Después de aplicar la regla:

"**Introducción:** En esta actividad vas a trabajar con circuitos eléctricos. Necesitas conocer la diferencia entre circuitos en serie y en paralelo.

Conceptos clave:

- *Circuito en serie:* La corriente eléctrica es la misma en todos los componentes.
- *Circuito en paralelo:* El voltaje es el mismo en todos los componentes.

Tarea: Observa el circuito adjunto y calcula su resistencia total.

Recuerda: Identifica primero si es un circuito en serie o en paralelo, porque la fórmula de cálculo es diferente en cada caso."

Ejemplo 2 – Antes de aplicar la regla:

"Analiza el poema. Identifica recursos estilísticos. Relaciona con el contexto histórico. Redacta conclusiones."

Después de aplicar la regla:

"**Contexto:** Vas a analizar un poema del Romanticismo español. El Romanticismo fue un movimiento cultural del siglo XIX que valoraba la libertad, la emoción y la naturaleza.

Paso 1: Lee el poema con atención e identifica recursos estilísticos (metáforas, personificaciones, hipérboles).

Paso 2: Relaciona el contenido del poema con las características del Romanticismo que hemos estudiado.

Paso 3: Redacta tus conclusiones: ¿por qué este poema es representativo del Romanticismo? ¿Qué emociones transmite?

Reflexiona: ¿Crees que los temas del Romanticismo (libertad, naturaleza, emociones intensas) siguen siendo relevantes hoy en día?"

A.4. REGLA 4. TONO Y MOTIVACIÓN

Esta regla promueve el uso de un tono positivo, empático y cercano en la redacción de los contenidos educativos. Esta regla busca reforzar la confianza del alumnado y fomentar su implicación en la tarea, reduciendo el miedo al error y la sensación de frustración ante actividades percibidas como complejas.

A.4.1. Fundamentación pedagógica

La dimensión afectiva del aprendizaje tiene un impacto significativo en el rendimiento académico y en la persistencia del alumnado ante las dificultades, estando demostrado que el lenguaje utilizado en los enunciados educativos puede influir en la percepción de autoeficacia del alumnado.

Un tono imperativo y distante ("Realice el ejercicio", "Identifique los errores") puede generar ansiedad y desconexión emocional, especialmente en alumnado con historial de fracaso escolar o baja autoestima académica. Por el contrario, un tono cercano y empático

("Vamos a trabajar juntos en esta actividad", "No te preocupes si al principio te resulta difícil") transmite apoyo y reduce la percepción de amenaza.

El lenguaje positivo no implica trivialización de las exigencias académicas, sino reformulación de las consignas de modo que el error se presente como parte natural del proceso de aprendizaje y no como fracaso personal.

A.4.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Identifique los errores de su código. Corríjalos."

Después de aplicar la regla:

"Revisa tu código con calma y busca posibles errores. No te preocupes si encuentras varios: es completamente normal. Los programadores profesionales pasan mucho tiempo depurando código. Una vez localizados los errores, corrígelos y vuelve a probar el programa."

Ejemplo 2 – Antes de aplicar la regla:

"Resuelva el problema. Compruebe que el resultado es correcto. Si es incorrecto, repita el procedimiento."

Después de aplicar la regla:

"Ahora te toca resolver el problema. Tómate tu tiempo para pensar en la solución. Una vez que tengas el resultado, compruébalo: ¿tiene sentido? Si no estás seguro, revisa cada paso con calma. Recuerda que equivocarse forma parte del aprendizaje."

A.5. REGLA 5. APRENDIZAJE AUTÓNOMO

Los textos deben redactarse como si el alumnado aprendiera de forma autónoma, incorporando aclaraciones, ejemplos o metáforas sencillas que puedan sustituir la figura del tutor en el momento inicial de la comprensión. Esta regla pretende favorecer la autorregulación y el aprendizaje a ritmo propio.

A.5.1. Fundamentación pedagógica

El aprendizaje autónomo es una competencia clave en la educación contemporánea y un requisito fundamental para el aprendizaje a lo largo de la vida. Sin embargo, muchos enunciados educativos presuponen la presencia de un docente que aclare dudas, proporcione ejemplos adicionales o reformule consignas ambiguas. Esta dependencia del apoyo docente puede actuar como barrera para el alumnado con menor confianza académica o en contextos de aprendizaje no presencial.

La incorporación de aclaraciones preventivas ("esto significa que...", "por ejemplo..."), definiciones integradas y metáforas explicativas reduce la necesidad de intervención docente sin eliminarla. El objetivo no es sustituir al docente, sino proporcionar al alumnado herramientas para avanzar de forma autónoma en las fases iniciales de comprensión de la tarea.

A.5.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Realiza un diagrama de Gantt para planificar tu proyecto."

Después de aplicar la regla:

"Realiza un diagrama de Gantt para planificar tu proyecto. Un diagrama de Gantt es una herramienta visual que muestra las tareas de un proyecto en forma de barras horizontales a lo largo del tiempo. Cada barra representa una tarea, y su longitud indica cuánto durará. Por ejemplo, si tu proyecto incluye las tareas 'Investigación' (2 semanas) y 'Redacción' (1 semana), dibujarías dos barras: una más larga para investigación y otra más corta para redacción. Puedes hacerlo en papel, con Excel o con herramientas online como TeamGantt."

Ejemplo 2 – Antes de aplicar la regla:

"Aplica el teorema de Pitágoras para resolver el problema."

Después de aplicar la regla:

"Aplica el teorema de Pitágoras para resolver el problema. Recuerda que el teorema de Pitágoras dice que, en un triángulo rectángulo (el que tiene un ángulo de 90°), el cuadrado de la hipotenusa (el lado más largo) es igual a la suma de los cuadrados de los otros dos lados. La fórmula es: $a^2 + b^2 = c^2$, donde c es la hipotenusa. Por ejemplo, si los lados son 3 y 4, entonces $3^2 + 4^2 = 9 + 16 = 25$, y la hipotenusa será $\sqrt{25} = 5$."

A.6. REGLA 6. INTENCIÓN PEDAGÓGICA

Todo enunciado educativo debe indicar de forma explícita los objetivos de la actividad, los criterios de entrega y, cuando sea posible, el tiempo estimado de trabajo. Esta explicitación permite al alumnado comprender qué se espera de él y planificar adecuadamente la realización de la tarea.

A.6.1. Fundamentación pedagógica

La claridad en las metas de aprendizaje mejora significativamente el rendimiento y la motivación del alumnado. Cuando los objetivos son explícitos, el alumnado puede:

- Entender el propósito pedagógico de la actividad ("¿qué voy a aprender?")
- Autoevaluar su progreso ("¿estoy cumpliendo lo que se espera?")
- Priorizar esfuerzos ("¿qué aspectos son más importantes?")
- Gestionar el tiempo ("¿cuánto tardaré en completar esto?")

La ausencia de objetivos explícitos genera incertidumbre y dificulta la autorregulación. El alumnado con necesidades educativas especiales, especialmente aquellos con dificultades de función ejecutiva, se beneficia especialmente de esta explicitación.

Los criterios de entrega claros (formato, extensión, modo de envío) reducen la ansiedad y evitan malentendidos. La estimación temporal ayuda al alumnado a gestionar su tiempo y a evitar la procrastinación.

A.6.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Investiga sobre las energías renovables y presenta tus conclusiones."

Después de aplicar la regla:

"**¿Qué vas a aprender?**

En esta actividad vas a conocer los principales tipos de energías renovables y sus ventajas e inconvenientes. Al finalizar, serás capaz de explicar por qué son importantes para el futuro del planeta.

¿Qué debes entregar?

Un documento en formato PDF de 2-3 páginas con tus conclusiones sobre las energías renovables.

¿Cómo se evaluará?

Se valorará que expliques al menos 3 tipos de energías renovables, que identifiques sus ventajas e inconvenientes, y que redactes con claridad.

¿Cuánto tiempo necesitarás?

Aproximadamente 2 horas (1 hora de investigación + 1 hora de redacción)."

Ejemplo 2 – Antes de aplicar la regla:

"Desarrolla una aplicación web con formulario de registro."

Después de aplicar la regla:

"**Objetivo de aprendizaje:**

Practicar la creación de formularios HTML y la validación de datos con JavaScript.

Tarea:

Desarrolla una aplicación web que incluya un formulario de registro de usuarios con los campos: nombre, email, contraseña y confirmación de contraseña. El formulario debe validar que todos los campos estén completos y que las contraseñas coincidan.

Criterios de entrega:

- Archivo HTML con el formulario
- Archivo JavaScript con la validación
- Archivo CSS para dar estilo (opcional)
- Comprimido en un archivo .zip

Plazo: 1 semana. Tiempo estimado de trabajo: 3-4 horas."

A.7. REGLA 7. APRENDIZAJE COLABORATIVO

LEdui fomenta la inclusión de propuestas que favorezcan la cooperación, la ayuda mutua y la evaluación entre iguales. Esta regla busca potenciar las habilidades sociales y comunicativas del alumnado, así como promover dinámicas de aprendizaje compartido.

A.7.1. Fundamentación pedagógica

El aprendizaje colaborativo demuestra que la interacción entre iguales puede tener efectos positivos tanto en el rendimiento académico como en el desarrollo de competencias sociales.

Cuando el alumnado trabaja colaborativamente:

- Verbaliza su pensamiento, lo que refuerza la comprensión.
- Recibe retroalimentación inmediata de sus compañeros.

- Desarrolla habilidades de comunicación y trabajo en equipo.
- Se beneficia de perspectivas diversas sobre un mismo problema.

Para el alumnado con necesidades educativas especiales, el aprendizaje colaborativo puede actuar como andamiaje natural, siempre que se estructuren adecuadamente los roles y se garantice la interdependencia positiva.

Esta regla no implica que todas las actividades deban ser colaborativas, sino que cuando sea pedagógicamente apropiado, se incluyan opciones o sugerencias de trabajo en equipo.

A.7.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Lee el texto y responde a las preguntas."

Después de aplicar la regla:

"Lee el texto y responde a las preguntas.

Opción de trabajo colaborativo: Si lo prefieres, puedes trabajar esta actividad en pareja. En ese caso, leed el texto juntos, discutid las respuestas y redactad vuestras conclusiones de forma conjunta. Ambos debéis participar activamente."

Ejemplo 2 – Antes de aplicar la regla:

"Diseña un logotipo para la empresa ficticia."

Después de aplicar la regla:

"Diseña un logotipo para la empresa ficticia.

Fase de brainstorming colaborativo: Antes de empezar, reúnete con 2-3 compañeros y compartid ideas sobre cómo podría ser el logotipo. Cada uno puede hacer bocetos rápidos y explicar su propuesta. Después de esta fase, cada uno desarrollará su propio diseño final, pero habrás enriquecido tu creatividad con las ideas del grupo.

Evaluación entre iguales: Cuando termines tu logotipo, intercámbialo con un compañero y pedid feedback mutuo: ¿el logotipo representa bien la empresa? ¿Es visualmente atractivo? ¿Qué aspectos mejorarías?"

A.8. REGLA 8. PENSAMIENTO CRÍTICO Y REFLEXIÓN

El formato incluye la incorporación de preguntas abiertas o dilemas breves que inviten al análisis personal y a la reflexión, sin imponer una respuesta cerrada. Esta regla favorece la profundización en los contenidos y estimula el pensamiento crítico del alumnado.

A.8.1. Fundamentación pedagógica

El pensamiento crítico, entendido como la capacidad de analizar información, evaluar argumentos y formular juicios fundamentados, constituye una competencia transversal clave en la educación contemporánea. La taxonomía de Bloom sitúa el análisis, la evaluación y la creación en los niveles cognitivos superiores, por encima de la mera memorización o comprensión.

Las preguntas de reflexión al final de una actividad cumplen varias funciones pedagógicas:

- Fomentan la metacognición ("¿qué he aprendido? ¿cómo lo he aprendido?")

- Promueven la transferencia del aprendizaje a contextos reales
- Estimulan el pensamiento divergente y la creatividad
- Permiten al alumnado posicionarse críticamente sobre los contenidos

Las preguntas reflexivas no tienen "respuestas correctas" únicas, lo que reduce la ansiedad evaluativa y permite que todo el alumnado, independientemente de su nivel de competencia, pueda participar aportando su perspectiva personal.

A.8.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Calcula el área del triángulo siguiendo la fórmula estudiada."

Después de aplicar la regla:

"Calcula el área del triángulo siguiendo la fórmula estudiada.

Reflexiona: ¿En qué situaciones de la vida real podría ser útil calcular el área de un triángulo? Piensa en profesiones u oficios donde esta habilidad matemática sea necesaria."

Ejemplo 2 – Antes de aplicar la regla:

"Lee el artículo sobre cambio climático y resume las ideas principales."

Después de aplicar la regla:

"Lee el artículo sobre cambio climático y resume las ideas principales.

Pregunta de reflexión crítica:

Según el artículo, algunas soluciones al cambio climático requieren cambios en nuestros hábitos de consumo. ¿Crees que los ciudadanos individuales tienen responsabilidad en este problema, o la responsabilidad es principalmente de gobiernos y grandes empresas? Justifica tu opinión con argumentos."

A.9. REGLA 9. CONEXIÓN PROFESIONAL Y CONTEXTUAL

Los contenidos deben relacionarse con situaciones reales del entorno cotidiano o profesional del alumnado. Esta conexión contextual facilita la transferencia del aprendizaje y ayuda a comprender la utilidad práctica de los conocimientos trabajados.

A.9.1. Fundamentación pedagógica

Es fundamental que el conocimiento no puede desvincularse del contexto en el que se adquiere y aplica. El aprendizaje abstracto y descontextualizado dificulta la transferencia a situaciones reales y reduce la motivación intrínseca del alumnado.

La conexión con contextos profesionales o cotidianos cumple varias funciones:

- Responde a la pregunta "¿para qué sirve esto?" que todo alumnado se plantea.
- Facilita la comprensión de conceptos abstractos mediante ejemplos concretos.
- Aumenta la motivación al evidenciar la relevancia del aprendizaje.
- Prepara para la aplicación del conocimiento en contextos laborales reales.

En Formación Profesional, esta regla es especialmente crítica, ya que el objetivo educativo es precisamente la adquisición de competencias profesionales aplicables en entornos laborales específicos.

A.9.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Resuelve la ecuación de segundo grado siguiente: $x^2 - 5x + 6 = 0$."

Después de aplicar la regla:

"**Contexto real:** Un ingeniero civil necesita calcular la trayectoria de un arco estructural. La altura del arco en cada punto se puede modelar mediante una ecuación de segundo grado.

Problema: La ecuación que describe el arco es $x^2 - 5x + 6 = 0$. Resuelve esta ecuación para determinar en qué puntos el arco toca el suelo (altura = 0).

Reflexiona: ¿En qué otras profesiones u oficios crees que se utilizan ecuaciones de segundo grado?"

Ejemplo 2 – Antes de aplicar la regla:

"Escribe un programa en Python que ordene una lista de números."

Después de aplicar la regla:

"**Situación profesional real:**

Trabajas como desarrollador en una empresa de e-commerce. El equipo de ventas necesita un informe de los productos más vendidos ordenados de mayor a menor número de ventas.

Tarea:

Escribe un programa en Python que reciba una lista de productos con sus ventas (por ejemplo: [("Portátil", 245), (Ratón", 1205), ("Teclado", 890)]) y devuelva la lista ordenada de mayor a menor número de ventas.

Pista: Investiga sobre el método `sort()` o la función `sorted()` de Python."

A.10. REGLA 10. RECURSOS MULTIMEDIA Y MULTIMODALES

Contemplar la posibilidad de trabajar y entregar contenidos mediante distintos formatos, como vídeo, audio, infografías o podcasts, adaptando el canal de comunicación a los diferentes estilos de aprendizaje y necesidades del alumnado.

A.10.1. Fundamentación pedagógica

Los marcos inclusivos trabajados sugieren que el alumnado procesa y retiene mejor la información cuando se presenta a través de canales variados. Aunque la evidencia sobre "estilos de aprendizaje" individuales es controvertida, sí está consolidada la idea de que la **multimodalidad** (presentar información a través de múltiples canales simultáneamente) refuerza la comprensión y la retención.

La teoría cognitiva del aprendizaje multimedia de Mayer establece que las personas aprenden mejor cuando se combina texto con imágenes o diagramas, siempre que exista coherencia semántica entre ambos. Los apoyos visuales reducen la carga cognitiva verbal y facilitan la construcción de representaciones mentales integradas.

Para el alumnado con necesidades educativas especiales, los recursos multimodales son especialmente valiosos. Por ejemplo:

- Alumnado con dislexia: Beneficio de contenidos en audio o vídeo.

- Alumnado con discapacidad auditiva: Beneficio de transcripciones y apoyos visuales.
- Alumnado con TDAH: Beneficio de contenidos dinámicos y visuales.

A.10.2. Ejemplos de aplicación

Ejemplo 1 – Antes de aplicar la regla:

"Entrega un informe escrito sobre el tema estudiado."

Después de aplicar la regla:

"Entrega tu trabajo sobre el tema estudiado. Puedes elegir el formato que prefieras:

- **Informe escrito:** Documento PDF de 3-4 páginas.
- **Vídeo explicativo:** Grábate explicando el tema (5-7 minutos).
- **Infografía:** Crea un póster visual con las ideas principales.
- **Podcast:** Graba un audio explicando el tema como si fuera un programa de radio.

Todos los formatos se valorarán igual. Elige el que mejor se adapte a tu forma de trabajar."

Ejemplo 2 – Antes de aplicar la regla:

"Lee el documento adjunto y responde a las preguntas."

Después de aplicar la regla:

"Estudia el contenido sobre fotosíntesis. Puedes hacerlo de diferentes formas:

- **Leer el documento PDF** adjunto (si prefieres leer)
- **Ver el vídeo explicativo** disponible en este enlace (si prefieres contenido audiovisual)
- **Escuchar el podcast** sobre el tema (si prefieres audio)

Después, responde a las preguntas. Todas las fuentes contienen la misma información, solo cambia el formato."

ANEXO B

Anexo B. Enunciados utilizados en la validación pedagógica

En este anexo se presentan los quince enunciados educativos sometidos a validación. Estos enunciados corresponden a distintos niveles formativos y familias profesionales, reflejando la diversidad de contextos educativos para los que se ha diseñado el formato LEdui.

Los quince enunciados utilizados en la validación se distribuyeron de la siguiente manera:

- **Educación Secundaria Obligatoria (3 enunciados):** Actividades de Lengua Castellana y Literatura (análisis de textos), Ciencias Sociales (comentario de fuentes históricas) y Matemáticas (resolución de problemas contextualizados).
- **Formación Profesional (9 enunciados):** Tareas de módulos técnicos de las familias de Informática y Comunicaciones (diseño de redes, programación, instalaciones domóticas), Administración y Gestión (elaboración de informes, gestión documental), Sanidad (análisis de sistemas sanitarios, gestión de documentación clínica) y Electricidad y Electrónica (instalaciones domóticas, sistemas de control).
- **Universidad (3 enunciados):** Actividades de asignaturas de grado en Historia (análisis historiográfico), Informática (desarrollo de software, análisis de algoritmos) y Educación (diseño de unidades didácticas).

Los quince enunciados presentados ilustran de forma transversal las principales carencias que el formato LEdui pretende subsanar en contextos educativos diversos:

- **Complejidad lingüística elevada:** uso extensivo de terminología técnica, académica o sectorial sin explicaciones ni glosarios de apoyo.
- **Ausencia de estructura guiada:** presentación en bloques textuales continuos que dificultan la identificación de las tareas requeridas y su secuenciación.
- **Falta de explicitación de expectativas:** no se especifican criterios de evaluación claros, extensión esperada ni tiempo estimado de dedicación.
- **Carencia de elementos motivacionales:** tono académico distante que no refuerza la confianza ni la autonomía del alumnado.
- **Ausencia de apoyos al aprendizaje autónomo:** no se ofrecen consejos, ejemplos, recursos de consulta ni formatos alternativos de entrega.
- **Sobrecarga cognitiva:** acumulación de múltiples tareas complejas sin gradación ni andamiaje que facilite su abordaje progresivo.

A continuación se incluyen estos enunciados. Para cada uno de ellos se ha incluido el nivel educativo, el texto original y las principales problemáticas identificadas en relación con las reglas del formato LEdui.

B.1. EDUCACIÓN SECUNDARIA OBLIGATORIA

En esta sección se incluyen 3 enunciados.

B.1.1. Enunciado ESO-1: Lengua Castellana y Literatura (4º ESO)

Enunciado original: *Lee detenidamente el fragmento del capítulo VII de La Regenta de Leopoldo Alas «Clarín» que se adjunta. Identifica y clasifica los recursos estilísticos presentes en el texto (metáforas, símiles, hipérboles, personificaciones) y analiza su función expresiva en el contexto de la narración. Posteriormente, interpreta el simbolismo del espacio descrito (la catedral de Vetusta) en relación con el estado emocional de la protagonista. Redacta un comentario crítico de entre 300 y 400 palabras en el que valores la relevancia del fragmento dentro de la novela naturalista española del siglo XIX, estableciendo conexiones con otros autores y obras del período estudiados en clase.*

Problemáticas identificadas:

- Vocabulario literario especializado sin explicaciones (*símiles*, *hipérboles*, *simbolismo*, *naturalista*)
- Múltiples tareas sin diferenciación clara (identificar, clasificar, analizar, interpretar, valorar)
- Ausencia de estructura: párrafo continuo sin pasos
- Verbos académicos complejos sin concretar acciones
- No especifica criterios de evaluación
- Falta tiempo estimado
- Tono académico distante

B.1.2. Enunciado ESO-2: Ciencias Sociales (4º ESO)

Enunciado original: *Examina críticamente las siguientes fuentes primarias sobre la Revolución Industrial: extractos del informe parlamentario británico sobre condiciones de trabajo infantil (1842), testimonios de obreros recogidos por Friedrich Engels y estadísticas demográficas de Manchester entre 1750 y 1850. Contrasta la información proporcionada por cada tipo de fuente, evalúa su fiabilidad considerando la autoría y el contexto histórico de producción, e infiere las transformaciones socioeconómicas que reflejan. Elabora una síntesis argumentada de 400-500 palabras que responda a la pregunta: ¿En qué medida la industrialización mejoró o empeoró las condiciones de vida de la clase trabajadora? Fundamenta tu posición con referencias específicas a las fuentes analizadas.*

Problemáticas identificadas:

- Vocabulario histórico técnico (*fuentes primarias*, *fiabilidad*, *transformaciones socioeconómicas*)
- Múltiples operaciones cognitivas sin secuenciación (examinar, contrastar, evaluar, inferir, sintetizar)
- Complejidad sintáctica elevada

- No explica cómo contrastar fuentes
- Falta estructura de pasos
- No especifica tiempo estimado
- Pregunta abierta compleja sin andamiaje

B.1.3. Enunciado ESO-3: Matemáticas (3º ESO)

Enunciado original: *Una empresa de telecomunicaciones ofrece dos tarifas para teléfono móvil. La Tarifa A tiene un coste fijo mensual de 15 euros y cada minuto de llamada cuesta 0,10 euros. La Tarifa B no tiene coste fijo pero cada minuto cuesta 0,25 euros. Determina algebraicamente el número de minutos a partir del cual resulta más económica la Tarifa A, plantea y resuelve la ecuación correspondiente, representa gráficamente ambas funciones de coste en el mismo sistema de coordenadas cartesianas e interpreta el significado del punto de corte en el contexto del problema. Justifica razonadamente qué tarifa recomendarías a un usuario que realiza una media de 80 minutos mensuales de llamadas.*

Problemáticas identificadas:

- Vocabulario matemático sin explicaciones (*algebraicamente*, *sistema de coordenadas cartesianas*, *punto de corte*)
- Múltiples tareas (determinar, plantear, resolver, representar, interpretar, justificar) sin estructura
- Enunciado denso en un solo párrafo
- No especifica formato esperado de respuesta
- Falta indicación de tiempo estimado
- No ofrece ayudas ni ejemplos

B.2. FORMACIÓN PROFESIONAL

En esta sección se incluyen 9 enunciados.

B.2.1. Enunciado FP-1: Diseño de redes (Grado Medio Informática)

Enunciado original: *Diseña la topología de red para una empresa con tres plantas (50 equipos en planta baja, 30 en primera planta, 20 en segunda planta) que requiere segmentación por departamentos, acceso a Internet redundante y servidor local para gestión documental. Especifica el direccionamiento IP utilizando subredes VLSM, justifica la elección del equipamiento activo (switches, routers, firewall) considerando throughput y escalabilidad, elabora el esquema lógico de la red diferenciando DMZ, LAN interna y conexión WAN, y documenta las políticas de seguridad básicas a implementar (ACLs, autenticación 802.1X). Presenta el proyecto en formato técnico profesional incluyendo presupuesto estimado.*

Problemáticas identificadas:

- Alta densidad de términos técnicos sin glosario (*VLSM*, *throughput*, *DMZ*, *ACLs*, *802.1X*)
- Múltiples entregas sin diferenciación (topología, direccionamiento, esquema, políticas, presupuesto)
- Ausencia de pasos numerados
- No especifica extensión esperada

- Falta tiempo estimado
- No menciona herramientas recomendadas

B.2.2. Enunciado FP-2: Programación (Grado Superior Informática)

Enunciado original: *Desarrolla una aplicación de escritorio en Java que implemente un sistema CRUD completo para gestión de inventario de una tienda, utilizando arquitectura MVC, persistencia mediante JDBC conectado a MySQL, validación de datos en todas las capas y manejo robusto de excepciones. La interfaz gráfica debe construirse con JavaFX siguiendo principios de usabilidad, incorporar búsqueda y filtrado avanzado de productos, generar informes exportables a PDF y cumplir con el patrón Singleton para la conexión a base de datos. Documenta el código siguiendo estándares JavaDoc, incluye diagramas UML de clases y secuencia, y prepara un manual de usuario básico.*

Problemáticas identificadas:

- Acrónimos técnicos sin explicar (*CRUD*, *MVC*, *JDBC*, *UML*)
- Requerimientos técnicos complejos sin priorización
- Múltiples entregables (código, diagramas, documentación, manual)
- No especifica criterios de evaluación
- Falta desglose temporal
- No ofrece recursos de apoyo

B.2.3. Enunciado FP-3: Instalaciones domóticas (Grado Superior Electrónica)

Enunciado original: *Proyecta un sistema domótico integral para una vivienda de 120m² que integre control de iluminación mediante DALI, climatización con termostatos inteligentes Zigbee, persianas motorizadas con KNX, sistema de alarma con sensores cableados e inalámbricos, y cámaras IP con almacenamiento NVR. El sistema debe centralizarse en un servidor domótico (Home Assistant o similar) accesible mediante app móvil. Elabora el proyecto técnico completo: planos de ubicación de dispositivos, cableado estructurado, cuadro de integración, configuración de automatizaciones básicas (escenas, horarios) y manual de uso para cliente. Incluye justificación de compatibilidad entre protocolos y estimación económica detallada.*

Problemáticas identificadas:

- Protocolos técnicos sin explicación (*DALI*, *Zigbee*, *KNX*, *NVR*)
- Gran cantidad de elementos sin organización clara
- Múltiples entregables técnicos
- No especifica herramientas de diseño requeridas
- Falta tiempo estimado
- Complejidad técnica elevada sin gradación

B.2.4. Enunciado FP-4: Gestión documental (Grado Superior Administración)

Enunciado original: *La dirección de tu empresa necesita implementar un sistema de gestión documental que cumpla con la normativa de protección de datos (RGPD) y permita clasificar, archivar y recuperar eficientemente documentación administrativa y comercial. Analiza las necesidades específicas de la organización, compara al menos tres soluciones software del mercado (considerando coste, funcionalidades, escalabilidad), propón un plan de migración*

desde el sistema actual en papel y carpetas compartidas, diseña el esquema de clasificación documental siguiendo criterios profesionales, establece niveles de acceso según roles y elabora un protocolo de archivo y destrucción de documentos conforme a los plazos legales. Presenta tu propuesta en formato informe ejecutivo de 6-8 páginas con índice, introducción, desarrollo y conclusiones.

Problemáticas identificadas:

- Siglas sin explicar (*RGPD*)
- Múltiples tareas complejas (analizar, comparar, proponer, diseñar, establecer)
- Ausencia de secuencia clara
- No especifica criterios de comparación de software
- Extensión definida pero sin tiempo estimado
- Falta indicaciones sobre fuentes de información

B.2.5. Enunciado FP-5: Análisis sanitario (Grado Superior Enfermería)

Enunciado original: *Analiza críticamente la evolución de los modelos de atención sanitaria basados en la Atención Primaria y la incorporación progresiva de tecnologías digitales (historia clínica electrónica, teleasistencia, monitorización remota). Establece relaciones entre estos avances, los cambios organizativos en los centros de salud y su impacto en la calidad asistencial, la accesibilidad y la protección de datos del paciente. Elabora un informe analítico de 800-1000 palabras, sustentado en legislación vigente, guías clínicas y literatura científica, citado en formato APA, en el que reflexiones sobre los desafíos éticos y profesionales que estos sistemas plantean y su proyección en el futuro del sector sociosanitario.*

Problemáticas identificadas:

- Vocabulario técnico denso sin explicaciones (*teleasistencia*, *monitorización remota*, *formato APA*)
- Ausencia de estructura: bloque textual continuo sin pasos claros
- Verbos académicos complejos: *analiza críticamente*, *establece relaciones*, *reflexiones*
- No especifica tiempo estimado de trabajo
- Falta tono motivador y cercano
- No ofrece formatos alternativos de entrega

B.2.6. Enunciado FP-6: Sistema Administrativo (Grado Superior Administración y Finanzas

Enunciado original: *Una PYME local desea modernizar su sistema administrativo y te solicita un informe ejecutivo que compare tres alternativas: Implantar un ERP básico, Mantener su sistema actual incorporando automatización contable, Externalizar parte de sus procesos administrativos. Evalúa cada opción considerando coste, riesgos, impacto en la productividad y requerimientos de formación. No es obligatorio incluir citas académicas, pero sí justificar cada recomendación con argumentos económicos o de gestión. Concluye con una propuesta concreta sobre la alternativa más adecuada y una breve hoja de ruta de implementación.*

Problemáticas identificadas:

- Uso de siglas sin explicar (*PYME*, *ERP*)

- Estructura más clara que otros enunciados, pero sin pasos numerados
- Falta explicitación de extensión esperada del informe
- No especifica tiempo estimado
- Carece de consejos o apoyos para el alumno que no sepa por dónde empezar
- No menciona posibilidad de formatos alternativos

B.2.7. Enunciado FP-7: Consumo energético (Grado Superior en Automatización)

Enunciado original: *Diseña una propuesta de instalación domótica básica para una vivienda unifamiliar que permita optimizar el consumo energético y mejorar la seguridad del hogar. Deberás especificar sensores, actuadores, protocolos de comunicación y un esquema funcional de cómo interactúan los dispositivos. Incluye una breve justificación técnica de por qué has seleccionado esos componentes frente a otras alternativas del mercado. Para cerrar el proyecto, añade una recomendación profesional sobre las consideraciones de mantenimiento y actualización que debería tener en cuenta el cliente final.*

Problemáticas identificadas:

- Vocabulario técnico específico sin definiciones (*domótica*, *actuadores*, *protocolos de comunicación*)
- Estructura narrativa sin pasos diferenciados
- Múltiples elementos a entregar sin desglose claro (esquema, justificación, recomendación)
- No especifica criterios de evaluación ni extensión
- Falta tiempo estimado
- No ofrece alternativas de formato de entrega

B.2.8. Enunciado FP-8: Análisis hotelero (Grado Superior en Alojamientos Turísticos)

Enunciado original: *Un establecimiento hotelero de categoría 4 estrellas ha detectado una disminución del 15 % en su índice de ocupación durante la temporada media y desea implementar estrategias de revenue management para optimizar ingresos. Analiza los datos históricos de ocupación y tarifas que se adjuntan (últimos 24 meses), identifica patrones estacionales y variables que afectan a la demanda, propón una estrategia de segmentación de mercado diferenciando clientes corporativos, leisure y grupos, diseña una política de tarifas dinámicas considerando yield management, establece indicadores KPI para medir resultados (ADR, RevPAR, ocupación) y elabora un plan de acción trimestral con acciones concretas de comercialización. Presenta el proyecto como informe de consultoría profesional.*

Problemáticas identificadas:

- Terminología sectorial anglosajona sin traducción (*revenue management*, *yield management*, *leisure*, *ADR*, *RevPAR*)
- Múltiples análisis y propuestas sin gradación
- Requiere interpretación de datos no incluidos en el enunciado
- No especifica extensión ni formato exacto
- Falta tiempo estimado

- Complejidad conceptual elevada

B.2.9. Enunciado FP-9: Imagen Personal (Grado Superior en Estilismo y Peluquería)

Enunciado original: *Desarrolla un proyecto de asesoría de imagen completa para un cliente ficticio que se incorpora a un puesto directivo en una multinacional. Partiendo del análisis morfológico (visagismo, estudio cromático personal, proporciones corporales), diseña propuestas de cambio de look que incluyan: corte y color de cabello adaptado a fisionomía facial y tendencias actuales, maquillaje profesional considerando características dermatológicas, recomendaciones de vestuario según dress code corporativo y ocasiones (formal, business casual, eventos), y pautas de mantenimiento y cuidado personal. Elabora un dossier visual profesional con bocetos, muestras de color, referencias fotográficas de inspiración y justificación técnica de cada decisión estética. Presenta el proyecto como si fuese para un cliente real.*

Problemáticas identificadas:

- Vocabulario técnico estético (*visagismo*, *estudio cromático*, *dress code*, *business casual*)
- Múltiples ámbitos de intervención sin estructura clara
- Requiere cliente ficticio pero no define características
- No especifica formato ni extensión del dossier
- Falta tiempo estimado
- No indica recursos técnicos necesarios

B.3. UNIVERSIDAD

En esta sección se incluyen 3 enunciados.

B.3.1. Enunciado UNI-1: Análisis historiográfico (Grado en Historia)

Enunciado original: *Elabora un análisis historiográfico comparativo sobre el debate en torno a la transición del feudalismo al capitalismo, contrastando las tesis de la escuela marxista británica (Dobb, Hilton, Brenner) con las interpretaciones de la escuela de los Annales (Braudel, Wallerstein) y las aproximaciones neoinstitucionalistas de North y Thomas. El trabajo debe contextualizar epistemológicamente cada corriente historiográfica, identificar sus marcos teórico-metodológicos, analizar sus divergencias en torno a la periodización, causalidad y protagonismo de los agentes históricos, y valorar críticamente la vigencia de sus planteamientos a la luz de la historiografía reciente. Incorpora el análisis de fuentes primarias que evidencien transformaciones estructurales en las relaciones de producción (documentación señorial, contratos agrarios, ordenanzas gremiales), aplicando técnicas de crítica externa e interna según la metodología de Langlois-Seignobos. Entrega: ensayo académico de 8000-10000 palabras con estructura IMRyD adaptada a humanidades, aparato crítico en sistema Harvard, y presentación oral defendiendo tu posicionamiento historiográfico ante un tribunal.*

Problemáticas identificadas:

- Altísima densidad de terminología especializada (*historiográfico*, *epistemológicamente*, *neoinstitucionalistas*, *IMRyD*, *crítica externa e interna*)
- Referencias a múltiples autores y escuelas sin contexto previo (Dobb, Hilton, Brenner, Braudel, Wallerstein, North, Thomas, Langlois-Seignobos)
- Complejidad metodológica elevada sin gradación por fases

- Múltiples tareas mezcladas (contextualizar, identificar, analizar, valorar, incorporar fuentes)
- Requerimientos sobre fuentes primarias sin orientación sobre acceso o selección
- No especifica tiempo de desarrollo ni fases de trabajo
- Vocabulario académico denso (*periodización*, *causalidad*, *agentes históricos*, *aparato crítico*)
- Falta orientación sobre priorización de contenidos
- Extensión elevada (8000-10000 palabras) sin desglose

B.3.2. Enunciado UNI-2: Análisis de algoritmos (Grado de Ingeniería Informática)

Enunciado original: *Analiza formal y empíricamente la eficiencia de tres algoritmos de ordenación (QuickSort, MergeSort, HeapSort) implementados en lenguaje C. Determina su complejidad temporal en notación Big-O para los casos mejor, promedio y peor, demostrando matemáticamente mediante recurrencias y aplicando el Teorema Maestro cuando proceda. Implementa los tres algoritmos optimizados, diseña un conjunto de benchmarks que evalúen su rendimiento con datasets de tamaños crecientes (*10^3, 10^4, 10^5, 10^6 *elementos) y diferentes distribuciones (ordenados, aleatorios, inversamente ordenados), mide tiempos de ejecución y consumo de memoria utilizando herramientas de profiling, y representa gráficamente los resultados comparando predicción teórica con comportamiento empírico.*

Problemáticas identificadas:

- Terminología matemática-computacional compleja (*Big-O*, *recurrencias*, *Teorema Maestro*, *profiling*)
- Doble enfoque (teórico y empírico) sin integración clara
- Múltiples fases (demostración, implementación, medición, análisis) sin cronograma
- Especifica formato IEEE pero no lo explica
- Extensión definida pero sin tiempo estimado
- Requiere conocimientos avanzados de matemáticas sin andamiaje

B.3.3. Enunciado UNI-3: Diseño de unidades didácticas (Grado en Educación Primaria)

Enunciado original: *Diseña una unidad didáctica completa para el área de Ciencias Naturales de 5º de Primaria sobre el sistema circulatorio humano, fundamentada en el marco teórico del constructivismo social de Vygotsky y los principios del Diseño Universal para el Aprendizaje (DUA). La unidad debe incluir: contextualización curricular alineada con Real Decreto 157/2022, análisis de competencias clave y saberes básicos asociados, justificación metodológica argumentando la selección de estrategias didácticas, secuencia de 8-10 sesiones detalladas con objetivos específicos, actividades diferenciadas por niveles de desempeño, recursos didácticos especificando TIC empleadas, instrumentos de evaluación criterial con rúbricas analíticas, y propuesta de atención a la diversidad contemplando alumnado NEAE. Entrega en formato documento profesional de 25-30 páginas con referencias bibliográficas en APA 7ª edición y anexos con materiales didácticos elaborados.*

Problemáticas identificadas:

- Vocabulario pedagógico especializado (*constructivismo social*, *competencias clave*, *rúbricas analíticas*, *NEAE*)
- Múltiples elementos curriculares sin priorización
- Referencias normativas específicas (RD 157/2022)
- Gran extensión (25-30 páginas) sin desglose temporal
- Requiere conocimiento de marcos teóricos previos
- No especifica tiempo de desarrollo del trabajo

ANEXO C

Anexo C. Intrumentos de validación pedagógica

Este anexo recoge los instrumentos de evaluación utilizados en el proceso de validación pedagógica del formato LEdui descrito en el capítulo 4 del documento principal. La validación se llevó a cabo mediante consulta a un panel de cinco profesionales especializados en educación inclusiva, quienes evaluaron quince enunciados educativos transformados al formato LEdui utilizando los instrumentos que se presentan a continuación.

Cada evaluador recibió un conjunto de tres a seis enunciados transformados correspondientes a diferentes niveles educativos y áreas de conocimiento, junto con los enunciados originales para su comparación. Los instrumentos se presentaron en formato digital editable para facilitar su cumplimentación.

C.1. INSTRUCCIONES PARA EVALUADORES

En este apartado se incluyen las instrucciones que recibe el evaluador.

C.1.1. Contexto de la evaluación

Este proceso de validación forma parte del Trabajo Fin de Grado titulado *"Servicio web para la generación automática de contenidos en formato de Lectura Educativa Inclusiva (LEdui)"*, desarrollado en el marco del Grado en Ingeniería Informática.

El objetivo de esta validación es evaluar la utilidad del formato LEdui desde la perspectiva de profesionales especializados en educación. Su participación como experto/a resulta fundamental para verificar que el formato propuesto responde efectivamente a necesidades reales identificadas en contextos educativos.

C.1.2. Formato de Lectura Educativa Inclusiva (LEdui)

LEdui es un formato de adaptación de contenidos educativos fundamentado en la convergencia de cinco marcos normativos de referencia: Diseño Universal para el Aprendizaje (DUA), Lectura Fácil (UNE 153101:2018), accesibilidad web (WCAG 2.1), norma de calidad de materiales educativos digitales (UNE 71362:2020) y guías de accesibilidad cognitiva.

El formato se articula en torno a **diez reglas operativas** que traducen principios pedagógicos abstractos en especificaciones concretas aplicables tanto manualmente por docentes como automáticamente mediante sistemas inteligentes:

1. **Claridad lingüística:** uso de vocabulario cotidiano, frases cortas y voz activa. Tecnicismos necesarios siempre explicados.

2. **Estructura guiada:** organización en pasos claros y secuenciales.
3. **Coherencia expositiva:** secuencia lógica con introducción, desarrollo y cierre. Uso de conectores.
4. **Tono y motivación:** lenguaje cercano, positivo y estimulante que refuerce la confianza.
5. **Aprendizaje autónomo:** aclaraciones, ejemplos y guías que sustituyan la intervención docente directa.
6. **Intención pedagógica:** explicitación de objetivos, criterios de entrega y tiempo estimado.
7. **Aprendizaje colaborativo:** propuestas que favorezcan la cooperación y evaluación entre iguales.
8. **Pensamiento crítico:** preguntas abiertas que inviten al análisis personal.
9. **Conexión profesional y contextual:** relación con situaciones reales del ámbito laboral o cotidiano.
10. **Recursos multimedia y multimodales:** integración de formatos alternativos de entrega (vídeo, audio, infografías).

C.1.3. Material proporcionado

Recibirá lo siguientes documentos:

- **Enunciado original:** texto tal como se plantearía habitualmente al alumnado, sin adaptaciones específicas.
- **Enunciado transformado:** misma actividad adaptada automáticamente al formato LEdui mediante el sistema desarrollado.

Se solicita que evalúe únicamente los enunciados transformados utilizando los instrumentos proporcionados (rúbrica de evaluación y cuestionario cualitativo).

C.1.4. Procedimiento de evaluación

1. Lea atentamente cada pareja enunciado original / enunciado transformado.
2. Cumplimente la **Rúbrica de Evaluación** para cada enunciado transformado, valorando el grado de cumplimiento de cada regla LEdui según la escala proporcionada.
3. Responda al **Cuestionario Cualitativo** tras evaluar los tres enunciados, aportando su valoración global sobre el formato LEdui y su utilidad práctica.
4. El tiempo estimado de evaluación es de aproximadamente 45-60 minutos.

C.1.5. Confidencialidad y uso de datos

Los datos recogidos se utilizarán exclusivamente con fines académicos en el marco de este Trabajo Fin de Grado. Las valoraciones individuales se tratarán de forma agregada y anónima en el análisis de resultados. Si desea recibir información sobre los resultados de la validación, puede indicarlo al final del cuestionario.

C.2. RÚBRICA DE EVALUACIÓN DE ENUNCIADOS TRANSFORMADOS

La tabla a continuación muestra la rúbrica utilizada.

EVALUACIÓN DE ENUNCIADO TRANSFORMADO			
Evaluador/a: _____ **Enunciado:** _____			
Nº	**Regla LEdui**	**Indicadores de cumplimiento**	**Valoración**
1	Claridad lingüística	• Vocabulario sencillo y cotidiano • Frases cortas (<20 palabras) • Tecnicismos explicados • Voz activa predominante	0-1-2-3-4 / N/A
2	Estructura guiada	• Organización en pasos numerados • Orden lógico y secuencial • Acciones claramente identificadas	0-1-2-3-4 / N/A
3	Coherencia expositiva	• Introducción, desarrollo, cierre • Uso de conectores textuales • Resumen de ideas complejas	0-1-2-3-4 / N/A
4	Tono y motivación	• Lenguaje cercano y positivo • Mensajes de ánimo • Refuerzo de la confianza	0-1-2-3-4 / N/A
5	Aprendizaje autónomo	• Aclaraciones y ejemplos integrados • Guías que sustituyan al tutor • Favorece autorregulación	0-1-2-3-4 / N/A
6	Intención pedagógica	• Objetivos explícitos • Criterios de entrega claros • Tiempo estimado especificado	0-1-2-3-4 / N/A
7	Aprendizaje colaborativo	• Propuestas de cooperación • Evaluación entre iguales • Trabajo conjunto	0-1-2-3-4 / N/A
8	Pensamiento crítico	• Preguntas abiertas incluidas • Invitación al análisis personal • Sin respuesta única impuesta	0-1-2-3-4 / N/A
9	Conexión contextual	• Relación con vida real/laboral • Ejemplos del entorno cercano • Transferencia del aprendizaje	0-1-2-3-4 / N/A
10	Multimodalidad	• Opciones de formato de entrega • Atención a estilos de aprendizaje • Explicación de alternativas	0-1-2-3-4 / N/A
Valoración global del enunciado (0-10):			_____

C.2.1. Instrucciones de cumplimentación de la rúbrica

Para cada uno de los enunciados transformados que se le proporcionan, valore el grado de cumplimiento de cada regla LEdui utilizando la siguiente escala:

- **0 - No cumple:** La regla no se aplica o se aplica de forma inadecuada.
- **1 - Cumplimiento bajo:** La regla se aplica parcialmente pero presenta deficiencias significativas.
- **2 - Cumplimiento medio:** La regla se cumple de forma aceptable pero podría mejorarse.
- **3 - Cumplimiento alto:** La regla se cumple adecuadamente.
- **4 - Cumplimiento excelente:** La regla se cumple de forma ejemplar, sin aspectos mejorables.
- **N/A - No aplicable:** La regla no es pertinente para este enunciado concreto.

C.3. CUESTIONARIO CUALITATIVO

Una vez evaluados los tres enunciados mediante la rúbrica, responda a las siguientes preguntas abiertas que permiten recoger su valoración global sobre el formato LEdui y su utilidad práctica. No hay extensión mínima ni máxima para las respuestas; exprese sus opiniones con la libertad y profundidad que considere apropiada.

C.3.1. Preguntas sobre el formato LEdui

Pregunta 1. Valoración general: tras evaluar los enunciados transformados, ¿cuál es su valoración general del formato LEdui? ¿Considera que las diez reglas propuestas son coherentes con principios pedagógicos de inclusión educativa?

Pregunta 2. Utilidad práctica: desde su experiencia profesional, ¿cree que el formato LEdui podría resultar útil para facilitar la comprensión de enunciados educativos al alumnado con necesidades educativas especiales? ¿Y al conjunto del alumnado?

Pregunta 3. Viabilidad de aplicación: ¿Considera viable la aplicación del formato LEdui en contextos reales de aula? ¿Qué dificultades o barreras identifica para su adopción por parte del profesorado?

Pregunta 4. Comparación con enunciados originales: al comparar los enunciados originales con los transformados, ¿qué aspectos considera que mejoran significativamente? ¿Identifica algún aspecto que podría haberse perdido o empeorado en la transformación?

Pregunta 5. Equilibrio entre accesibilidad y rigor: una preocupación habitual al adaptar contenidos educativos es la posible pérdida de rigor académico o simplificación excesiva. ¿Considera que los enunciados transformados mantienen un nivel de exigencia adecuado al nivel educativo correspondiente?

Pregunta 6. Reglas LEdui: relevancia y priorización: de las diez reglas LEdui, ¿cuáles considera que aportan mayor valor para la accesibilidad cognitiva de los enunciados? ¿Hay alguna regla que considere menos relevante o que podría reformularse?

Pregunta 7. Perfiles de alumnado: ¿Para qué perfiles de alumnado con necesidades educativas especiales considera que el formato LEdui resultaría especialmente beneficioso? ¿Identifica algún perfil para el que podría no ser adecuado?

Pregunta 8. Automatización mediante IA: los enunciados evaluados han sido transformados automáticamente mediante un sistema de inteligencia artificial. ¿Considera que la calidad de las adaptaciones es comparable a lo que conseguiría un docente especializado adaptando manualmente? ¿Qué ventajas o limitaciones identifica en la automatización?

Pregunta 9. Sugerencias de mejora: ¿Qué modificaciones o ampliaciones propondría para mejorar el formato LEdui? ¿Qué elementos adicionales deberían incorporarse?

Pregunta 10. Valoración final: Si tuviese que resumir en una frase su opinión sobre el formato LEdui y su potencial para mejorar la accesibilidad de contenidos educativos, ¿cuál sería?

ANEXO D

Anexo D. Documentación de la API REST

D.1. INTRODUCCIÓN

Este anexo presenta la documentación técnica completa de la API REST desarrollada para el sistema de transformación de contenidos educativos al formato LEdui. La API sigue los principios arquitectónicos REST y expone un conjunto de endpoints organizados por funcionalidad, diseñados para permitir tanto la transformación automática de enunciados educativos como la gestión integral del ciclo de vida de dichas transformaciones.

D.1.1. Información general del servicio

- **Título**: API Lectura Educativa
- **Versión**: 11.0.0
- **Framework**: FastAPI (Python)
- **Arquitectura**: Cliente-servidor (REST)
- **Base de datos**: PostgreSQL
- **Autenticación**: JWT + Google OAuth 2.0

D.1.2. URL base y versionado

La API se despliega en un entorno de producción accesible mediante HTTPS. La URL base depende del entorno de despliegue:

- **Desarrollo local**: `http://localhost:8000`
- **Producción**: URL proporcionada por el servicio Railway

El versionado de la API se gestiona mediante un número de versión semántico incluido en los metadatos del servicio, accesible a través del endpoint `/salud`.

D.1.3. Esquema de autenticación

La API utiliza un sistema de autenticación de dos niveles:

1. **Autenticación externa**: Mediante Google OAuth 2.0, que permite a los usuarios identificarse con su cuenta de Google.
2. **Autenticación interna**: Mediante tokens JWT (JSON Web Tokens) emitidos por la propia API tras validar el token de Google.

Todos los endpoints protegidos requieren incluir el token JWT en el header de autorización:

```
Authorization: Bearer <token_jwt>
```

La duración de los tokens JWT es de 24 horas desde su emisión.

D.1.4. Códigos de estado HTTP

La API utiliza los siguientes códigos de estado HTTP estándar:

Código	Significado
200	Petición exitosa
201	Recurso creado exitosamente
400	Petición mal formada o parámetros inválidos
401	No autenticado (token ausente o inválido)
403	Sin permisos suficientes (requiere rol admin)
404	Recurso no encontrado
500	Error interno del servidor
503	Servicio no disponible (configuración incompleta)

Tabla D.1: Códigos de estado HTTP utilizados por la API

D.1.5. Formato de las peticiones y respuestas

Todas las peticiones y respuestas utilizan el formato JSON (*application/json*), con la excepción de los endpoints de descarga de archivos que devuelven contenido binario o HTML.

Las fechas se representan en formato ISO 8601, y todos los textos utilizan codificación UTF-8.

D.2. AUTENTICACIÓN Y GESTIÓN DE USUARIOS

Este bloque de endpoints gestiona la autenticación de usuarios mediante Google OAuth y proporciona información sobre el usuario autenticado y sus estadísticas de uso.

D.2.1. POST /auth/google

Endpoint para autenticación de usuarios mediante token de Google OAuth 2.0.

Descripción: Recibe un token de autenticación de Google, lo valida con los servicios de Google, crea o recupera el usuario en la base de datos y devuelve un token JWT para uso en la API.

Autenticación requerida: No (este es el endpoint de autenticación inicial)

Parámetros de entrada (cuerpo JSON):

Campo	Tipo	Obligatorio	Descripción
token	string	Sí	Token de Google OAuth 2.0 obtenido del frontend

Ejemplo de petición:

```
POST /auth/google
Content-Type: application/json

{
  "token": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjZm..."
}
```

Respuesta exitosa (200):

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "bearer",
  "user": {
    "id": 1,
    "email": "usuario@ejemplo.com",
    "name": "Juan Pérez",
    "picture": "https://lh3.googleusercontent.com/...",
    "is_admin": false
  }
}
```

Posibles errores:

- **401**: Token de Google inválido o expirado
- **500**: Error al comunicar con servicios de Google o error interno

D.2.2. GET /auth/me

Obtiene la información completa del usuario actualmente autenticado.

Descripción: Devuelve los datos del perfil del usuario asociado al token JWT proporcionado.

Autenticación requerida: Sí (token JWT válido)

Parámetros: Ninguno (el usuario se identifica mediante el token)

Ejemplo de petición:

```
GET /auth/me
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Respuesta exitosa (200):

```
{
  "id": 1,
  "email": "usuario@ejemplo.com",
  "name": "Juan Pérez",
  "picture": "https://lh3.googleusercontent.com/...",
  "is_admin": false
}
```

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado

D.2.3. GET /auth/stats

Obtiene estadísticas de uso del usuario actual.

Descripción: Calcula y devuelve métricas agregadas sobre el uso del sistema por parte del usuario autenticado, incluyendo número de transformaciones, tokens consumidos y recursos generados.

Autenticación requerida: Sí (token JWT válido)

Parámetros: Ninguno

Ejemplo de petición:

```
GET /auth/stats
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Respuesta exitosa (200):

```
{
  "transformaciones_totales": 20,
  "transformaciones_exitosas": 18,
  "tokens_usados": 45320,
  "pictogramas_generados": 15,
  "imagenes_generadas": 12
}
```

Descripción de los campos de respuesta:

Campo	Descripción
transformaciones_totales	Número total de transformaciones iniciadas
transformaciones_exitosas	Número de transformaciones completadas sin errores
tokens_usados	Suma total de tokens de entrada y salida consumidos en llamadas a modelos de IA
pictogramas_generados	Número de conjuntos de pictogramas generados
imagenes_generadas	Número de imágenes de contexto generadas

Tabla D.2: Campos de respuesta del endpoint de estadísticas

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **500**: Error al consultar la base de datos

D.3. TRANSFORMACIÓN DE CONTENIDOS EDUCATIVOS

Este bloque agrupa los endpoints responsables de la transformación automática de enunciados educativos al formato LEdui y la generación de recursos complementarios (explicaciones pedagógicas, pictogramas e imágenes de contexto).

D.3.1. POST /transformar/adaptar

Endpoint principal para la adaptación de enunciados educativos al formato LEdui.

Descripción: Recibe un enunciado educativo en formato textual y lo transforma aplicando las 10 reglas del formato LEdui de manera armonizada y determinista. Genera una estructura HTML fija que incluye objetivos, pasos, criterios de entrega, preguntas de reflexión y estimación temporal.

Autenticación requerida: Sí (token JWT válido)

Parámetros de entrada (cuerpo JSON):

Campo	Tipo	Obligatorio	Descripción
enunciado	string	Sí	Texto del enunciado original a adaptar
nivel	string	No	Nivel educativo: "secundaria"(por defecto), "formacion_profesional", üniversidad"
max_palabras	integer	No	Límite aproximado de palabras para la adaptación (por defecto: 350)

Ejemplo de petición:

```
POST /transformar/adaptar
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado": "Diseña una propuesta de instalación domótica
               básica para una vivienda unifamiliar que permita
               optimizar el consumo energético y mejorar la
               seguridad del hogar.",
  "nivel": "formacion_profesional",
  "max_palabras": 300
}
```

Respuesta exitosa (200):

```
{
  "exito": true,
  "enunciado_adaptado": "<div class='contenido-ledui'>
    <h2>¡Crea tu propio hogar inteligente! Diseña una
        instalación domótica que ahorra y protege</h2>
    <h3>¿Qué vas a aprender?</h3>
    <p>En esta actividad aprenderás a diseñar un sistema
       domótico real que hará que una casa sea más
       eficiente y segura...</p>
    ...",
  "nivel": "formacion_profesional",
  "tokens_entrada": 125,
  "tokens_salida": 487,
  "duracion_segundos": 4.3,
  "transformacion_id": 42
}
```

Descripción de los campos de respuesta:

Posibles errores:

- **400**: Enunciado vacío o parámetros inválidos
- **401**: Token JWT ausente, inválido o expirado

Campo	Descripción
exito	Booleano que indica si la transformación se completó correctamente
enunciado_adaptado	Texto HTML con el enunciado adaptado al formato LEdui
nivel	Nivel educativo aplicado en la transformación
tokens_entrada	Número de tokens consumidos en la entrada al modelo de IA
tokens_salida	Número de tokens generados en la salida del modelo de IA
duracion_segundos	Tiempo de procesamiento de la transformación
transformacion_id	Identificador único de la transformación en la base de datos

- **503**: Servicio de IA no disponible (falta configuración de API key)
- **500**: Error al procesar la transformación o al guardar en base de datos

D.3.2. POST /transformar/explicar

Genera una explicación pedagógica sobre las reglas LEdui aplicadas en una transformación.

Descripción: Analiza un enunciado adaptado y genera un texto explicativo dirigido al profesorado que identifica las reglas LEdui más relevantes aplicadas, señala carencias del enunciado original y ofrece orientaciones para la creación de futuros contenidos educativos.

Autenticación requerida: Sí (token JWT válido)

Parámetros de entrada (cuerpo JSON):

Campo	Tipo	Obligatorio	Descripción
enunciado_adaptado	string	Sí	Texto del enunciado ya adaptado a LEdui
transformacion_id	integer	No	ID de la transformación para asociar la explicación

Ejemplo de petición:

```
POST /transformar/explicar
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado_adaptado": "<div>...</div>",
  "transformacion_id": 42
}
```

Respuesta exitosa (200):

```
{
  "exito": true,
  "explicacion_pedagogica": "Regla 2 (Estructura guiada):
    El enunciado original presentaba todas las tareas en un
    único bloque continuo. Se reorganizó en pasos numerados...\n\n
    Regla 6 (Aprendizaje autónomo): Se añadieron explicaciones
    sobre cómo abordar cada tarea sin esperar la intervención
    del docente...",
```

```
  "tokens_entrada": 156,
  "tokens_salida": 312,
  "duracion_segundos": 3.1
}
```

Posibles errores:

- **400**: Enunciado adaptado vacío
- **401**: Token JWT ausente, inválido o expirado
- **503**: Servicio de IA no disponible
- **500**: Error al generar la explicación

D.3.3. POST /transformar/pictogramas

Genera pictogramas educativos asociados a las competencias presentes en un enunciado adaptado.

Descripción: Analiza el enunciado adaptado, extrae las acciones y competencias clave, las simplifica a verbos básicos, consulta la API de ARASAAC para obtener pictogramas representativos, los descarga y los compone en una imagen unificada con explicaciones textuales.

Autenticación requerida: Sí (token JWT válido)

Parámetros de entrada (cuerpo JSON):

Campo	Tipo	Obligatorio	Descripción
enunciado	string	Sí	Enunciado adaptado del que extraer competencias
transformacion_id	integer	No	ID para asociar los pictogramas a una transformación

Ejemplo de petición:

```
POST /transformar/pictogramas
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado": "Pasos a seguir:\n1. Investiga y selecciona...",
  "transformacion_id": 42
}
```

Respuesta exitosa (200):

```
{
  "exito": true,
  "pictogramas_generados": 6,
  "acciones_encontradas": [
    {
      "accion_educativa": "Investigar y seleccionar componentes",
      "accion_simplificada": "buscar",
      "explicacion": "Vas a buscar y elegir qué sensores..."
    },
```

```
    ...
  ],
  "explicacion_habilidades": "buscar: Vas a buscar y elegir
    qué sensores usar...\ndibujar: Vas a dibujar un plano
    sencillo...",
  "tiene_imagen": true,
  "imagen_ancho": 1800,
  "imagen_alto": 300
}
```

Posibles errores:

- **400**: Enunciado vacío
- **401**: Token JWT ausente, inválido o expirado
- **503**: Servicio de IA o API de ARASAAC no disponible
- **500**: Error al generar pictogramas o componer imagen

D.3.4. POST /transformar/imagenes-contexto

Genera imágenes educativas de contexto asociadas a un enunciado adaptado.

Descripción: Analiza el enunciado, genera prompts detallados con valor pedagógico, utiliza Google Gemini para crear imágenes ilustrativas en formato 16:9, y proporciona justificaciones educativas de cada imagen generada.

Autenticación requerida: Sí (token JWT válido)

Parámetros de entrada (cuerpo JSON):

Campo	Tipo	Obligatorio	Descripción
enunciado	string	Sí	Enunciado adaptado del que generar imágenes
num_imagenes	integer	No	Número de imágenes a generar (1-3, por defecto 1)
estilo	string	No	Estilo: .[ed]ucativo"(defecto), ïlustrativo", realista", .[es]quematico"
transformacion_id	integer	No	ID para asociar las imágenes

Ejemplo de petición:

```
POST /transformar/imagenes-contexto
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado": "¡Crea tu propio hogar inteligente!...",
  "num_imagenes": 1,
  "estilo": "educativo",
  "transformacion_id": 42
}
```

Respuesta exitosa (200):

```
{
  "exito": true,
  "imagenes_generadas": 1,
  "imagenes": [
    {
      "id": 78,
      "prompt_utilizado": "Educational illustration showing
        a cross-section view of a modern smart home...",
      "justificacion_educativa": "Esta imagen muestra de
        manera visual y comprensible cómo se integran los
        diferentes componentes...",
      "texto_explicativo": "Esta imagen te ayudará a
        visualizar cómo se integran diferentes sensores...",
      "tiene_imagen": true,
      "ancho": 1792,
      "alto": 1024
    }
  ]
}
```


Posibles errores:

- **400**: Enunciado vacío o número de imágenes fuera de rango
- **401**: Token JWT ausente, inválido o expirado
- **503**: Servicio de IA o Google Gemini no disponible
- **500**: Error al generar imágenes

D.4. GESTIÓN DEL HISTORIAL DEL USUARIO

Este bloque de endpoints permite a los usuarios autenticados consultar, visualizar y descargar sus transformaciones anteriores.

D.4.1. GET /mi-historial

Obtiene el listado paginado de transformaciones del usuario actual.

Descripción: Devuelve todas las transformaciones realizadas por el usuario autenticado, ordenadas por fecha de creación descendente.

Autenticación requerida: Sí (token JWT válido)

Parámetros de consulta (query):

Parámetro	Tipo	Obligatorio	Descripción
limit	integer	No	Número máximo de resultados (por defecto: 50)
offset	integer	No	Número de resultados a saltar (por defecto: 0)

Ejemplo de petición:

```
GET /mi-historial?limit=10&offset=0
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Respuesta exitosa (200):

```
{
  "total": 20,
  "transformaciones": [
    {
      "id": 42,
      "enunciado_original": "Diseña una propuesta de...",
      "nivel": "formacion_profesional",
      "exito": true,
      "created_at": "2026-01-15T10:30:00",
      "duracion_segundos": 4.3
    },
    ...
  ]
}
```

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **500**: Error al consultar la base de datos

D.4.2. GET /mi-historial/{transformacion_id}

Obtiene el detalle completo de una transformación específica del usuario.

Descripción: Recupera todos los datos asociados a una transformación concreta: enunciado adaptado, explicaciones pedagógicas, pictogramas e imágenes de contexto.

Autenticación requerida: Sí (token JWT válido)

Parámetros de ruta:

Parámetro	Tipo	Descripción
transformacion_id	integer	ID de la transformación a consultar

Ejemplo de petición:

```
GET /mi-historial/42
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Respuesta exitosa (200):

```
{
  "id": 42,
  "enunciado_original": "Diseña una propuesta de...",
  "enunciado_adaptado": "<div>...</div>",
  "nivel": "formacion_profesional",
  "explicacion_pedagogica": "Regla 2 (Estructura guiada)...",
  "created_at": "2026-01-15T10:30:00",
  "pictogramas": {
    "id": 15,
    "pictogramas_generados": 6,
    "explicacion_habilidades": "buscar: Vas a buscar...",
    "tiene_imagen": true
```

```
  },
  "imagenes_contexto": [
    {
      "id": 78,
      "justificacion_educativa": "Esta imagen muestra...",
      "texto_explicativo": "Esta imagen te ayudará...",
      "tiene_imagen": true
    }
  ]
}
```

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **403**: La transformación no pertenece al usuario actual
- **404**: Transformación no encontrada
- **500**: Error al consultar la base de datos

D.4.3. GET /mi-historial/{transformacion_id}/download-docx

Descarga una transformación completa en formato Word editable.

Descripción: Genera un documento HTML/DOCX que incluye el enunciado adaptado, pictogramas (como imágenes embebidas) e imágenes de contexto con sus justificaciones educativas.

Autenticación requerida: Sí (token JWT válido)

Parámetros de ruta:

Parámetro	Tipo	Descripción
transformacion_id	integer	ID de la transformación a descargar

Ejemplo de petición:

```
GET /mi-historial/42/download-docx
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Respuesta exitosa (200):

Archivo binario con headers:

```
Content-Type: application/vnd.openxmlformats-officedocument
              .wordprocessingml.document
Content-Disposition: attachment;
  filename="transformacion_42_usuario_LEdu.docx"
```

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **403**: La transformación no pertenece al usuario actual
- **404**: Transformación no encontrada
- **500**: Error al generar el documento

D.5. PANEL DE ADMINISTRACIÓN

Este bloque agrupa endpoints exclusivos para usuarios con rol de administrador, permitiendo una visión global del sistema y la gestión de usuarios y transformaciones.

D.5.1. GET /admin/users

Obtiene el listado completo de usuarios registrados en el sistema.

Descripción: Devuelve todos los usuarios con su información básica y estadísticas de uso.

Autenticación requerida: Sí (token JWT válido + rol admin)

Ejemplo de petición:

```
GET /admin/users
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Respuesta exitosa (200):

```
{
  "total": 8,
  "users": [
    {
      "id": 1,
      "email": "usuario@ejemplo.com",
      "name": "Juan Pérez",
      "is_admin": false,
      "created_at": "2025-12-01T09:00:00",
      "transformaciones_count": 20
    },
    ...
  ]
}
```

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **403**: Usuario sin permisos de administrador
- **500**: Error al consultar la base de datos

D.5.2. GET /admin/transformaciones

Obtiene el listado global de todas las transformaciones del sistema.

Descripción: Similar al endpoint de historial de usuario, pero sin restricción por usuario. Permite filtrar y paginar.

Autenticación requerida: Sí (token JWT válido + rol admin)

Parámetros de consulta:

Ejemplo de petición:

```
GET /admin/transformaciones?limit=20&nivel=formacion_profesional
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

Parámetro	Tipo	Obligatorio	Descripción
limit	integer	No	Límite de resultados (defecto: 50)
offset	integer	No	Offset para paginación (defecto: 0)
user_id	integer	No	Filtrar por ID de usuario específico
nivel	string	No	Filtrar por nivel educativo

Respuesta exitosa (200): Similar a `/mi-historial`, pero incluyendo el `user_id` en cada transformación.

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **403**: Usuario sin permisos de administrador
- **500**: Error al consultar la base de datos

D.5.3. GET /admin/transformaciones/{transformacion_id}

Obtiene el detalle de cualquier transformación del sistema.

Descripción: Idéntico a `/mi-historial/{id}` pero sin restricción de propiedad.

Autenticación requerida: Sí (token JWT válido + rol admin)

Respuesta: Ver sección 4.2 (estructura idéntica).

D.5.4. GET /admin/transformaciones/{transformacion_id}/download-docx

Descarga cualquier transformación en formato Word.

Descripción: Idéntico a `/mi-historial/{id}/download-docx` pero sin restricción de propiedad.

Autenticación requerida: Sí (token JWT válido + rol admin)

Respuesta: Ver sección 4.3 (formato idéntico).

D.5.5. GET /admin/pictogramas/{pictogramas_id}/imagen

Descarga la imagen compuesta de pictogramas de una transformación.

Descripción: Devuelve el archivo de imagen PNG con los pictogramas unidos.

Autenticación requerida: Sí (token JWT válido + rol admin)

Parámetros de ruta:

Parámetro	Tipo	Descripción
pictogramas_id	integer	ID del registro de pictogramas

Respuesta exitosa (200):

Archivo binario PNG con headers:

```
Content-Type: image/png
Content-Disposition: attachment;
  filename="pictogramas_15.png"
```

Posibles errores:

- **401**: Token JWT ausente, inválido o expirado
- **403**: Usuario sin permisos de administrador
- **404**: Imagen no encontrada

D.5.6. GET /admin/imagenes/{imagen_id}/imagen

Descarga una imagen de contexto específica.

Descripción: Devuelve el archivo de imagen PNG generado.

Autenticación requerida: Sí (token JWT válido + rol admin)

Parámetros de ruta:

Parámetro	Tipo	Descripción
imagen_id	integer	ID de la imagen de contexto

Respuesta: Similar a `/admin/pictogramas/{id}/imagen`, devuelve imagen PNG.

Posibles errores: Mismos que el endpoint anterior.

D.6. MODELOS DE DATOS (PYDANTIC SCHEMAS)

La API utiliza modelos Pydantic para la validación automática de datos de entrada y la serialización de respuestas. A continuación se detallan los principales esquemas utilizados.

D.6.1. AdaptarEntrada

Modelo para la entrada del endpoint de adaptación de enunciados.

Campo	Tipo	Requerido	Descripción
enunciado	string	Sí	Enunciado original
nivel	string	No	Nivel educativo (defecto: "secundaria")
max_palabras	integer	No	Límite de palabras (defecto: 350)

D.6.2. ImagenContextoEntrada

Modelo para la generación de imágenes de contexto.

Campo	Tipo	Requerido	Descripción
enunciado	string	Sí	Enunciado del que generar imágenes
num_imagenes	integer	No	Cantidad (1-3, defecto: 1)
estilo	string	No	Estilo visual (defecto: .^{ed}ucativo")
transformacion_id	integer	No	ID para asociar las imágenes

D.6.3. PictogramasEntrada

Modelo para la generación de pictogramas.

D.6.4. GoogleTokenInput

Modelo para la autenticación con Google.

Campo	Tipo	Requerido	Descripción
enunciado	string	Sí	Enunciado estructurado del que extraer acciones
transformacion_id	integer	No	ID para asociar los pictogramas

Campo	Tipo	Requerido	Descripción
token	string	Sí	Token de Google OAuth 2.0

D.6.5. TokenResponse

Modelo de respuesta tras autenticación exitosa.

Campo	Tipo	Descripción
access_token	string	Token JWT para uso en la API
token_type	string	Tipo de token (siempre "bearer")
user	object	Objeto con datos del usuario (ver UserResponse)

D.6.6. UserResponse

Modelo de datos del usuario.

D.7. EJEMPLOS DE USO COMPLETOS

Esta sección presenta flujos de trabajo completos que ilustran el uso secuencial de múltiples endpoints para casos de uso reales.

D.7.1. Flujo 1: Autenticación y transformación completa

Objetivo: Un usuario se autentica y realiza una transformación completa de un enunciado.

Paso 1: Autenticación con Google

```
POST /auth/google
Content-Type: application/json

{
  "token": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjZm..."
}

RESPUESTA:
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "bearer",
  "user": {
    "id": 5,
    "email": "docente@escuela.es",
    "name": "María García",
    "is_admin": false
  }
}
```

Paso 2: Adaptar enunciado

Campo	Tipo	Descripción
id	integer	Identificador único del usuario
email	string	Correo electrónico
name	string (opcional)	Nombre completo
picture	string (opcional)	URL de la foto de perfil
is_admin	boolean	Indica si tiene rol de administrador

```
POST /transformar/adaptar
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado": "Analiza críticamente el impacto de la
               Revolución Industrial en las estructuras
               sociales europeas del siglo XIX",
  "nivel": "secundaria",
  "max_palabras": 300
}

RESPUESTA:
{
  "exito": true,
  "enunciado_adaptado": "<div>...</div>",
  "transformacion_id": 87
}
```

Paso 3: Generar explicación pedagógica

```
POST /transformar/explicar
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado_adaptado": "<div>...</div>",
  "transformacion_id": 87
}

RESPUESTA:
{
  "exito": true,
  "explicacion_pedagogica": "Regla 1 (Claridad lingüística)..."
}
```

Paso 4: Generar pictogramas

```
POST /transformar/pictogramas
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado": "<div>...</div>",
```

```
  "transformacion_id": 87
}

RESPUESTA:
{
  "exito": true,
  "pictogramas_generados": 5,
  "tiene_imagen": true
}
```

Paso 5: Generar imagen de contexto

```
POST /transformar/imagenes-contexto
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

{
  "enunciado": "<div>...</div>",
  "num_imagenes": 1,
  "estilo": "educativo",
  "transformacion_id": 87
}

RESPUESTA:
{
  "exito": true,
  "imagenes_generadas": 1
}
```

D.7.2. Flujo 2: Consulta y descarga del historial

Objetivo: Un usuario consulta sus transformaciones anteriores y descarga una en formato Word.

Paso 1: Listar historial

```
GET /mi-historial?limit=10&offset=0
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

RESPUESTA:
{
  "total": 15,
  "transformaciones": [
    { "id": 87, "nivel": "secundaria", ... },
    { "id": 86, "nivel": "formacion_profesional", ... },
    ...
  ]
}
```

Paso 2: Obtener detalle de una transformación

```
GET /mi-historial/87
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
```

```
RESPUESTA:
{
  "id": 87,
  "enunciado_adaptado": "<div>...</div>",
  "explicacion_pedagogica": "...",
  "pictogramas": { ... },
  "imagenes_contexto": [ ... ]
}
```

Paso 3: Descargar en Word

```
GET /mi-historial/87/download-docx
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

RESPUESTA: Archivo .docx descargado
```

D.7.3. Flujo 3: Administración del sistema

Objetivo: Un administrador consulta usuarios y transformaciones globales.

Paso 1: Listar usuarios

```
GET /admin/users
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

RESPUESTA:
{
  "total": 8,
  "users": [
    {
      "id": 5,
      "email": "docente@escuela.es",
      "name": "María García",
      "is_admin": false,
      "transformaciones_count": 15
    },
    ...
  ]
}
```

Paso 2: Ver transformaciones de un usuario específico

```
GET /admin/transformaciones?user_id=5&limit=5
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

RESPUESTA:
{
  "total": 15,
  "transformaciones": [ ... ]
}
```

Paso 3: Descargar imagen de pictogramas

```
GET /admin/pictogramas/23/imagen
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...

RESPUESTA: Archivo PNG descargado
```

D.8. MANEJO DE ERRORES Y CÓDIGOS DE ESTADO

La API implementa un sistema consistente de manejo de errores mediante códigos de estado HTTP estándar y mensajes descriptivos en formato JSON.

D.8.1. Estructura de respuesta de error

Todos los errores devuelven un objeto JSON con el siguiente formato:

```
{
  "detail": "Descripción legible del error"
}
```

En algunos casos, especialmente para errores de validación (400), el campo `detail` puede ser un array con múltiples errores.

D.8.2. Catálogo completo de códigos de error

Código	Situación	Ejemplo de mensaje
400	Parámetros inválidos o petición mal formada	."El enunciado no puede estar vacío"
401	Token JWT ausente, inválido o expirado	"Token inválido o expirado"
403	Usuario sin permisos suficientes	"No tienes permisos de administrador"
404	Recurso no encontrado	"Transformación no encontrada"
500	Error interno del servidor	."Error al procesar la transformación"
503	Servicio externo no disponible	"Servicio no disponible: falta configurar ANTHROPIC_API_KEY"

Tabla D.3: Códigos de estado HTTP y situaciones de error

D.8.3. Errores de autenticación (401)

Los errores 401 ocurren cuando:

- No se proporciona el header `Authorization`
- El token JWT ha expirado (más de 24 horas)
- El token JWT no puede ser validado (firma incorrecta)
- El usuario asociado al token ya no existe

Ejemplo:

```
{
  "detail": "Token inválido o expirado"
}
```

D.8.4. Errores de permisos (403)

Los errores 403 ocurren cuando:

- Se intenta acceder a un endpoint de administración sin ser admin
- Se intenta acceder a una transformación de otro usuario

Ejemplo:

```
{
  "detail": "No tienes permisos de administrador"
}
```

D.8.5. Errores de validación (400)

Los errores 400 ocurren cuando los datos de entrada no cumplen las restricciones definidas en los modelos Pydantic.

Ejemplo:

```
{
  "detail": [
    {
      "loc": ["body", "num_imagenes"],
      "msg": "ensure this value is less than or equal to 3",
      "type": "value_error.number.not_le"
    }
  ]
}
```

D.8.6. Errores de servicio no disponible (503)

Estos errores indican problemas de configuración o indisponibilidad de servicios externos:

- Falta la API key de Anthropic (Claude)
- Falta la API key de Google Gemini
- El servicio de ARASAAC no está accesible

Ejemplo:

```
{
  "detail": "Servicio no disponible: falta configurar
             ANTHROPIC_API_KEY"
}
```

D.9. CONSIDERACIONES TÉCNICAS ADICIONALES

D.9.1. Límites de uso y quotas

El sistema no implementa límites de rate limiting en la API, pero está sujeto a las quotas de los servicios externos integrados:

- **Anthropic Claude**: Límites según plan contratado
- **Google Gemini**: Límites según configuración de proyecto
- **ARASAAC**: Sin límites conocidos (servicio público)

D.9.2. Formato de fechas

Todas las fechas se devuelven en formato ISO 8601 con zona horaria UTC:

`"2026-01-15T10:30:00.123456+00:00"`

D.9.3. Codificación de caracteres

Todas las peticiones y respuestas utilizan codificación UTF-8. Los caracteres especiales españoles (ñ, acentos) están completamente soportados.

D.9.4. Tamaño máximo de peticiones

El tamaño máximo de los enunciados de entrada está limitado a:

- **Enunciados para adaptar**: 5000 caracteres aproximadamente
- **Enunciados para pictogramas**: Sin límite explícito (limitado por procesamiento de IA)
- **Enunciados para imágenes**: Sin límite explícito

D.9.5. Concurrencia y procesamiento asíncrono

La API está implementada sobre FastAPI, que soporta procesamiento asíncrono. Sin embargo, las llamadas a servicios de IA externos son secuenciales por diseño, para garantizar el determinismo del resultado.

D.9.6. Versionado de la API

La versión actual de la API es **11.0.0**. Aunque actualmente no existe un esquema de versionado en las URLs (`/v1/`, `/v2/`), se recomienda verificar la versión mediante el endpoint `/salud` antes de realizar integraciones.

D.9.7. Documentación interactiva

FastAPI genera automáticamente documentación interactiva en dos formatos:

- **Swagger UI**: Disponible en `/docs`
- **ReDoc**: Disponible en `/redoc`

Estas interfaces permiten probar los endpoints directamente desde el navegador, incluyendo la autenticación mediante tokens JWT.

D.10. RESUMEN DE ENDPOINTS

La siguiente tabla presenta un resumen completo de todos los endpoints disponibles en la API, organizados por categoría funcional.

D.10.1. Endpoints de autenticación

Método	Endpoint	Descripción	Auth
POST	/auth/google	Autenticación con Google OAuth	No
GET	/auth/me	Información del usuario actual	Sí
GET	/auth/stats	Estadísticas de uso del usuario	Sí

Tabla D.4: Endpoints de autenticación y gestión de usuarios

D.10.2. Endpoints de transformación

Método	Endpoint	Descripción	Auth
POST	/transformar/adaptar	Adaptación de enunciado a LEdui	Sí
POST	/transformar/explicar	Generación de explicación pedagógica	Sí
POST	/transformar/pictogramas	Generación de pictogramas	Sí
POST	/transformar/imagenes-contexto	Generación de imágenes de contexto	Sí

Tabla D.5: Endpoints de transformación de contenidos

D.10.3. Endpoints de historial de usuario

Método	Endpoint	Descripción	Auth
GET	/mi-historial	Listado de transformaciones del usuario	Sí
GET	/mi-historial/{id}	Detalle de transformación del usuario	Sí
GET	/mi-historial/{id}/download-docx	Descarga transformación en Word	Sí

Tabla D.6: Endpoints de gestión del historial del usuario

D.10.4. Endpoints de administración

Método	Endpoint	Descripción	Auth
GET	/admin/users	Listado de usuarios	Admin
GET	/admin/transformaciones	Listado global de transformaciones	Admin
GET	/admin/transformaciones/{id}	Detalle de cualquier transformación	Admin
GET	/admin/transformaciones/{id}/download-docx	Descarga cualquier transformación	Admin
GET	/admin/pictogramas/{id}/imagen	Descarga imagen de pictogramas	Admin
GET	/admin/imagenes/{id}/imagen	Descarga imagen de contexto	Admin

Tabla D.7: Endpoints del panel de administración