

UAX FP
Universidad
Alfonso X el Sabio

Desarrollador de Aplicaciones Multiplataforma

Segundo / DAM

Diego Fabian Ordoñez Ontaneda

Tutor del TFG: Jorge Sanchez Rubio

DEDICATORIA

A mi madre, mis hermanos y mi pareja,

Por haber sido mi mayor sostén durante este camino.

Gracias por vuestro apoyo incondicional, tanto económico como emocional,

por cada palabra de ánimo, por cada consejo,

y por impulsarme a seguir adelante en este reto que ha sido cursar este grado superior.

En momentos de incertidumbre, nunca dudaron en recordarme que aún con la edad o las
circunstancias,

mis ganas de aprender a programar eran válidas y valiosas.

Gracias por estar ahí, por creer en mí incluso cuando yo dudaba,

y por hacerme sentir que, si en algún momento me faltaba algo,

nunca me faltaría vuestra mano tendida.

Este proyecto también es vuestro.

ÍNDICES

Abstract del Proyecto (Español – Inglés)	4
Justificación del Proyecto	5
Introducción	6
Objetivos	7
Descripción	12
Flujo de la Arquitectura de Solución	13
Diagrama Global de Casos de Uso	17
Diagrama de Clases	35
Diagrama de Clases – Modelos	36
Tabla de Vistas y Funciones	37
Diagrama E/R (Entidad - Relación)	38
Diagrama de Base de Datos	39
Diagrama de Flujo de Navegación	40
Interfaces de la Aplicación	41
Diagrama de Red	54
Tecnologías	55
Metodología Aplicada	59
Diagrama de Gantt	62
Presupuesto del Proyecto	63
README y Control de Versiones	65
Contacto	67
Trabajos Futuros	68
Conclusiones	68
Referencias	69

ABSTRACT (ESPAÑOL)

EdukAI

Es una plataforma educativa inteligente desarrollada con Django y OpenAI, diseñada para ayudar a estudiantes a comprender y reforzar sus conocimientos a través del procesamiento automatizado de apuntes. El usuario puede subir archivos PDF o escribir texto manualmente, y la inteligencia artificial se encarga de generar resúmenes automáticos, proponer categorías educativas relevantes y elaborar cuestionarios tipo test personalizados. El sistema permite también visualizar la puntuación obtenida, comparar mejoras en el rendimiento, descargar apuntes y tests en distintos formatos, y acceder a todas las funcionalidades desde un entorno visual intuitivo y responsive.

Además, el acceso como invitado permite probar funciones clave sin registro, mientras que los usuarios autenticados disponen de un perfil con avatar, historial completo de actividades y gestión personalizada de apuntes.

La organización automática de documentos por temáticas como Matemáticas, Historia o Ciencias facilita un acceso estructurado al conocimiento.

EdukAI está pensado para ser escalable y flexible, con posibilidad de implementación tanto en servidores locales como en la nube (por ejemplo, AWS). El proyecto se ha estructurado en fases: ideación, diseño, desarrollo, integración de IA y herramientas de exportación.

ABSTRACT (ENGLISH)

EdukAI

Is an intelligent educational platform built with Django and OpenAI, designed to assist students in understanding and reinforcing their knowledge through automated note processing. Users can upload PDF files or enter text manually, and the AI system will generate automatic summaries, suggest appropriate educational categories, and create customized multiple-choice quizzes.

The platform allows users to review their scores, track performance over time, download summaries and quizzes in various formats, and interact with a modern, responsive user

interface.

A guest mode is available to test key features without registration, while registered users have access to a personalized profile with avatar, activity history, and organized note management.

Automatic classification of documents by subjects like Math, History, or Science provides structured access to learning materials.

EdukAI is designed for scalability and flexibility, with deployment options ranging from local servers to cloud platforms such as AWS. The project is organized into key phases: ideation, design, development, AI integration, and export functionality.

JUSTIFICACIÓN DEL PROYECTO

La idea de EdukAI nace de una necesidad real vivida como estudiante: el poco tiempo para repasar los apuntes, la sobrecarga de información, y la dificultad de preparar exámenes de forma efectiva, especialmente en entornos de estudio online. En muchos casos, nos encontramos con textos largos, mal organizados o poco claros, y sin una forma sencilla de convertirlos en materiales prácticos de repaso, como resúmenes o preguntas tipo test.

Durante el curso, varios compañeros intentaban prepararse utilizando páginas web donde se copiaban tests preexistentes o generados manualmente, lo que en ocasiones no se adaptaba realmente al contenido que estábamos estudiando. De ahí surge la motivación de crear una herramienta más personalizada y automatizada, que se encargue de transformar apuntes en recursos de estudio útiles: resúmenes claros y tests dinámicos.

Aunque existen herramientas como Quizlet, Notion AI o herramientas de IA integradas en plataformas educativas, la mayoría no están pensadas específicamente para estudiantes que quieren automatizar el proceso desde sus propios apuntes. EdukAI se diferencia en que combina:

- Resumen automático con inteligencia artificial (OpenAI)
- Generación aleatoria de preguntas tipo test con opciones personalizadas
- Registro y comparación de resultados en test
- Exportación de apuntes y tests en formatos reutilizables (PDF)

- Experiencia sencilla, visual y accesible incluso sin registrarse

EdukAI está dirigido a estudiantes de secundaria, bachillerato, formación profesional o universitarios que buscan optimizar su tiempo de estudio. También puede ser útil para opositores o personas que siguen cursos online y necesitan reforzar lo aprendido. La idea es ofrecer una ayuda real para el día a día del estudiante, sin complicaciones técnicas y con el respaldo de la inteligencia artificial.

INTRODUCCIÓN

EdukAI es una aplicación pensada para ayudar a estudiantes que quieren aprovechar mejor sus apuntes, ahorrar tiempo y preparar exámenes de forma más práctica. Muchas veces nos encontramos con montones de información sin resumir, textos largos o apuntes que no están del todo claros. Además, preparar preguntas para repasar puede ser algo que lleva mucho tiempo, y no siempre tenemos a alguien que nos evalúe o nos prepare tests.

Esta plataforma intenta resolver eso. EdukAI permite subir apuntes (en formato PDF o escribiéndolos directamente), resumirlos con ayuda de una IA, generar preguntas tipo test automáticamente y organizar todo en un entorno visual e intuitivo. Todo se guarda por categorías, y se puede ver un historial con los resúmenes, las preguntas y la puntuación de los tests realizados, la plataforma ofrece un consejo motivador diario generado por inteligencia artificial, adaptado al perfil del estudiante, que se muestra directamente en el panel principal.

Además, se puede usar como invitado sin necesidad de registrarse, aunque para guardar el progreso se recomienda crear una cuenta. La interfaz es sencilla, moderna y adaptable a distintos dispositivos.

Funciones principales que ofrece EdukAI:

- Subir apuntes o escribir texto para generar resúmenes con IA.
- Generación automática de preguntas tipo test (con 4 opciones).
- Guardado de resúmenes y tests asociados a cada apunte.

- Registro de puntuaciones e historial por usuario.
- Exportación de apuntes y tests en PDF.
- Uso como invitado o con cuenta personalizada.

Requisitos principales que debe cumplir el proyecto:

- Ser accesible para cualquier estudiante, sin conocimientos técnicos.
- Usar inteligencia artificial (OpenAI) para automatizar tareas.
- Permitir crear cuentas, iniciar sesión y gestionar perfil.
- Contar con un diseño responsive y visualmente atractivo.
- Almacenar los apuntes, resúmenes y resultados de forma segura.
- Ofrecer una navegación fluida y clara desde el dashboard.
- Poder usar funcionalidades básicas sin necesidad de registrarse.

OBJETIVOS

EdukAI tiene como objetivo principal facilitar a los estudiantes el procesamiento de apuntes mediante inteligencia artificial. Permite transformar textos en resúmenes, generar preguntas tipo test, organizar apuntes por categorías y registrar puntuaciones. Además, ofrece la posibilidad de usar la aplicación sin estar registrado, conservando funcionalidades clave.

A continuación se presenta el **RFTP inicial** que acompaña esta propuesta.

R01 – El programa debe permitir el acceso únicamente a usuarios registrados.

- **R01F01** – El usuario debe poder registrarse en el sistema mediante un enlace registro en el panel principal de la aplicación.
 - R01F01T01 – Crear una tabla usuarios en la base de datos.

- R01F01T01P01 – Introducir un usuario de prueba y comprobar almacenamiento.
- R01F01T02 – Diseñar un formulario de registro visual con validaciones.
- R01F01T02P01 – Visualizar register.html y enviar datos (nombre, correo, que estudio realiza, contraseña).
- **R01F02** – El usuario debe poder iniciar sesión con nombre de usuario o email y contraseña.
 - R01F02T01 – Crear la vista CustomLoginView con validación de usuario activo.
 - R01F02T01P01 – Probar inicio de sesión con un usuario activo/inactivo.
 - R01F02T01P02 – Probar que, si el usuario no está activo por caducidad del token de registro o por olvido de contraseña, el sistema muestre un mensaje adecuado y permita reenviar el enlace correspondiente.
 -
- **R01F03** – El usuario podrá recuperar su contraseña mediante un correo.
 - R01F03T01 – Implementar sistema de recuperación con token.
 - R01F03T01P01 – Enviar un email real y restablecer contraseña desde enlace.
 - R01F03T01P02 – Si el token esta caducado se solicita nuevo token recogiendo el email automáticamente desde la pantalla login

R02 – El programa debe permitir a los usuarios gestionar su perfil personal.

- **R02F01** – El usuario podrá acceder a su perfil desde el menú lateral y modificar su información personal.
 - **R02F01T01** – Crear la vista perfil_view protegida por login.
 - **R02F01T01P01** – Comprobar acceso a la página /perfil/ solo para usuarios autenticados.

- **R02F01T02** – Permitir editar nombre de usuario, email (solo lectura), y estudios actuales.
- **R02F01T02P01** – Mostrar los campos en un formulario con los datos precargados.
- **R02F01T02P02** – Guardar los cambios en el modelo PerfilUsuario y el modelo User.
- **R02F02** – El usuario podrá cambiar su contraseña directamente desde el perfil.
 - **R02F02T01** – Añadir campo opcional para introducir nueva contraseña en el formulario de perfil.
 - **R02F02T01P01** – Validar seguridad de la contraseña usando validate_password.
 - **R02F02T01P02** – Si la contraseña cambia, actualizar la sesión y enviar un email de notificación al usuario.
- **R02F03** – El usuario podrá seleccionar un avatar predeterminado o subir una imagen personalizada.
 - **R02F03T01** – Crear lista de avatares prediseñados con nombres asociados.
 - **R02F03T01P01** – Mostrar en el formulario un select para elegir un avatar y/o subir una imagen.
 - **R02F03T01P02** – Previsualizar la imagen de perfil antes de guardar los cambios.
 - **R02F03T01P03** – Si el usuario sube una nueva imagen, eliminar la anterior del sistema de archivos.
 - **R02F03T01P04** – Si se elige un avatar, asignar la ruta del avatar y eliminar la imagen personalizada si existía.
- **R02F04** – El usuario podrá eliminar su imagen de perfil actual desde un checkbox.
 - **R02F04T01** – Agregar campo eliminar_imagen en el formulario.

- **R02F04T01P01** – Si se marca la casilla, eliminar el archivo del sistema de archivos y limpiar el campo imagen en el modelo y asignar un avatar predeterminado al usuario.

R03 – El usuario podrá subir apuntes y obtener resúmenes generados por IA.

- **R03F01** – El usuario puede subir un archivo PDF o escribir texto (switch).
 - R03F01T01 – Crear modelo Apunte con campos archivo/texto fecha de creacion usuario y resumen generado.
 - R03F01T01P01 – Probar subida manual y mediante archivo.
- **R03F02** – El sistema debe extraer texto y resumirlo con OpenAI y agregarlo al modelo Apunte que le corresponde.
 - R03F02T01 – Integrar la API de OpenAI (resumir_texto_con_ia()).
 - R03F02T01P01 – Verificar resumen generado desde texto de prueba.
- **R03F03** – El usuario puede optar por personalizar el título y la categoría o dejarlo en manos de la IA.
 - R03F03T01 – Añadir switch en subir_apunte.html para alternar.
 - R03F03T01P01 – Verificar ambos modos funcionando.

R04 – El sistema debe organizar los apuntes por categorías.

- **R04F01** – Las categorías pueden ser elegidas por el usuario o sugeridas automáticamente.
 - R04F01T01 – Crear modelo Categoria y asociarlo al apunte.
 - R04F01T01P01 – Verificar selección o creación dinámica de categorías.
- **R04F02** – Las categorías se listarán en formularios y filtros.
 - R04F02T01 – Crear desplegable con select de categorías.
 - R04F02T01P01 – Comprobar filtrado de apuntes por categoría.

R05 – El usuario podrá generar preguntas tipo test a partir de un apunte.

- **R05F01** – Generar preguntas automáticamente con IA.
 - R05F01T01 – Crear función generar_preguntas_con_ia().
 - R05F01T01P01 – Comprobar formato JSON válido con 15 preguntas y 4 opciones de respuesta.
- **R05F02** – Guardar preguntas asociadas al apunte.
 - R05F02T01 – Crear modelo Pregunta y asociarlo con el apunte y usuario.
 - R05F02T01P01 – Insertar preguntas y verificarlas en admin.
- **R05F03** – Mostrar preguntas asociadas al apunte.
 - R05F03T01 – Crear vista dentro de Pregunta para mostrar las preguntas generadas con su respuesta.
 - R05F03T01P01 – Insertar preguntas y verificarlas su visualizacion tras presionar boton ver preguntas.

R06 – El usuario podrá realizar tests sobre las preguntas generadas.

- **R06F01** – Mostrar preguntas de forma secuencial y recoger respuestas.
 - R06F01T01 – Diseñar vista interactiva con JS y feedback inmediato.
 - R06F01T01P01 – Probar recorrido completo del test.
- **R06F02** – Al finalizar el test, mostrar puntuación y guardar resultado.
 - R06F02T01 – Crear modelo ResultadoTest este se asociara al modelo Pregunta.
 - R06F02T01P01 – Probar guardado de resultados y cálculo de mejora.

R07 – Mostrar contenido motivador generado por IA en el dashboard

- **R07F01** – Generar consejo diario personalizado según el estudio del usuario.
 - R07F01T01 – Llamar a la API de OpenAI y usar caché para evitar repeticiones.

- R07F01T01P01 – Verificar que el consejo se muestra correctamente y cambia cada día.

R08 – El sistema permitirá uso como invitado (sin cuenta).

- **R08F01** – El invitado podrá subir apuntes y generar resúmenes.
 - R08F01T01 – Desactivar funcionalidades de guardado para invitados solo podra copiar los textos resumidos generados.
 - R08F01T01P01 – Verificar vista limitada sin acceso a perfil , tests interactivos o almacenamiento de apuntes.

R09 – El sistema debe permitir descargar apuntes y tests en PDF.

- **R09F01** – Los apuntes subidos o escritos podrán exportarse en PDF para usuarios registrados para usuarios invitados solo se puede copiar el resumen generado de los apuntes.
- **R09F02** – Las preguntas y respuestas se podrán descargar como test imprimible para usuarios registrados y no registrados.

DESCRIPCIÓN

En este apartado se detallan los componentes principales que conforman el funcionamiento de la aplicación EdukAI. A través de distintos diagramas y explicaciones técnicas, se muestran las funcionalidades clave del sistema, los distintos tipos de usuarios, los procesos de autenticación, el uso de inteligencia artificial, y la estructura de almacenamiento de datos.

Cada sección está pensada para ofrecer una visión clara y ordenada del desarrollo propuesto, facilitando la comprensión del flujo de información y las decisiones técnicas adoptadas en el diseño de la solución.aplicación, y la interacción entre las distintas capas del sistema.



Explicación del Flujo de la Arquitectura de Solución – EdukAI

El siguiente diagrama representa el flujo lógico y funcional de la plataforma web **EdukAI**, desde el registro del usuario hasta el procesamiento de apuntes mediante inteligencia artificial, mostrando cómo se gestionan los datos y cómo interactúan los distintos componentes del sistema.

1. Registro y Autenticación

- El usuario se registra en la plataforma ingresando su **nombre de usuario, email, estudios y contraseña**.
- Se genera un **token de verificación** que se envía por correo electrónico para confirmar la cuenta.
- Tras la validación del token, se guarda el usuario en la **base de datos** y se permite el acceso a la autenticación mediante **Django**.

1.1 Recuperación de Contraseña / Token Caducado

- Si un usuario ha perdido el acceso o su token de activación ha expirado, puede solicitar:
 - **Reenvío del token de verificación.**
 - **Restablecimiento de la contraseña.**
- El sistema verifica el email, genera un nuevo token (si aplica) y envía un enlace por correo.
- El usuario accede a un formulario donde puede introducir una nueva contraseña.
- Tras validar el token, la nueva contraseña se **actualiza en la base de datos**.
- Este flujo asegura la recuperación de cuentas sin comprometer la seguridad.

2. Inicio de Sesión y Acceso

- Desde el módulo central **EdukAI**, el usuario puede iniciar sesión, registrarse o acceder como **invitado**.
- La autenticación se gestiona mediante **Django**, generando un token para mantener la sesión activa.

3. Usuario Invitado

- Tiene acceso limitado a funcionalidades:
 - Puede **subir archivos PDF**.
 - Puede **generar resúmenes y preguntas tipo test** mediante IA.
 - Puede **descargar preguntas**, pero **no se guarda su progreso ni historial ni puede visualizarlas preguntas**.

4. Usuario Registrado

- Tiene acceso completo a todas las funcionalidades:
 - Subir archivos PDF y generar resúmenes con OpenAI.
 - Crear preguntas de estudio.
 - Visualizar contenido generado (resúmenes y preguntas) ademas de los apuntes subidos a EdukAI.
 - Realizar **tests interactivos** y ver puntuaciones.
 - Descargar los resultados y consultar su **historial de puntuaciones**.
 - Clasificar automáticamente sus apuntes mediante categorización temática.
- Todo el progreso se **almacena en la base de datos** asociada al perfil del usuario.

5. Perfil de Usuario

- Muestra y permite editar información personal como:
 - Nombre, estudios, contraseña, avatar o foto de perfil.
- Cualquier cambio realizado se actualiza en la **base de datos**.

6. Inteligencia Artificial – OpenAI

- La IA se utiliza para procesar el contenido de los apuntes mediante distintos tipos de **prompts**, como:
 - Generación de resúmenes.
 - Creación de preguntas.
 - Generación de títulos y categorías.
- EdukAI convierte el PDF a texto y lo envía como prompt a **OpenAI**, quien devuelve los resultados que se integran en la plataforma.

7. Base de Datos

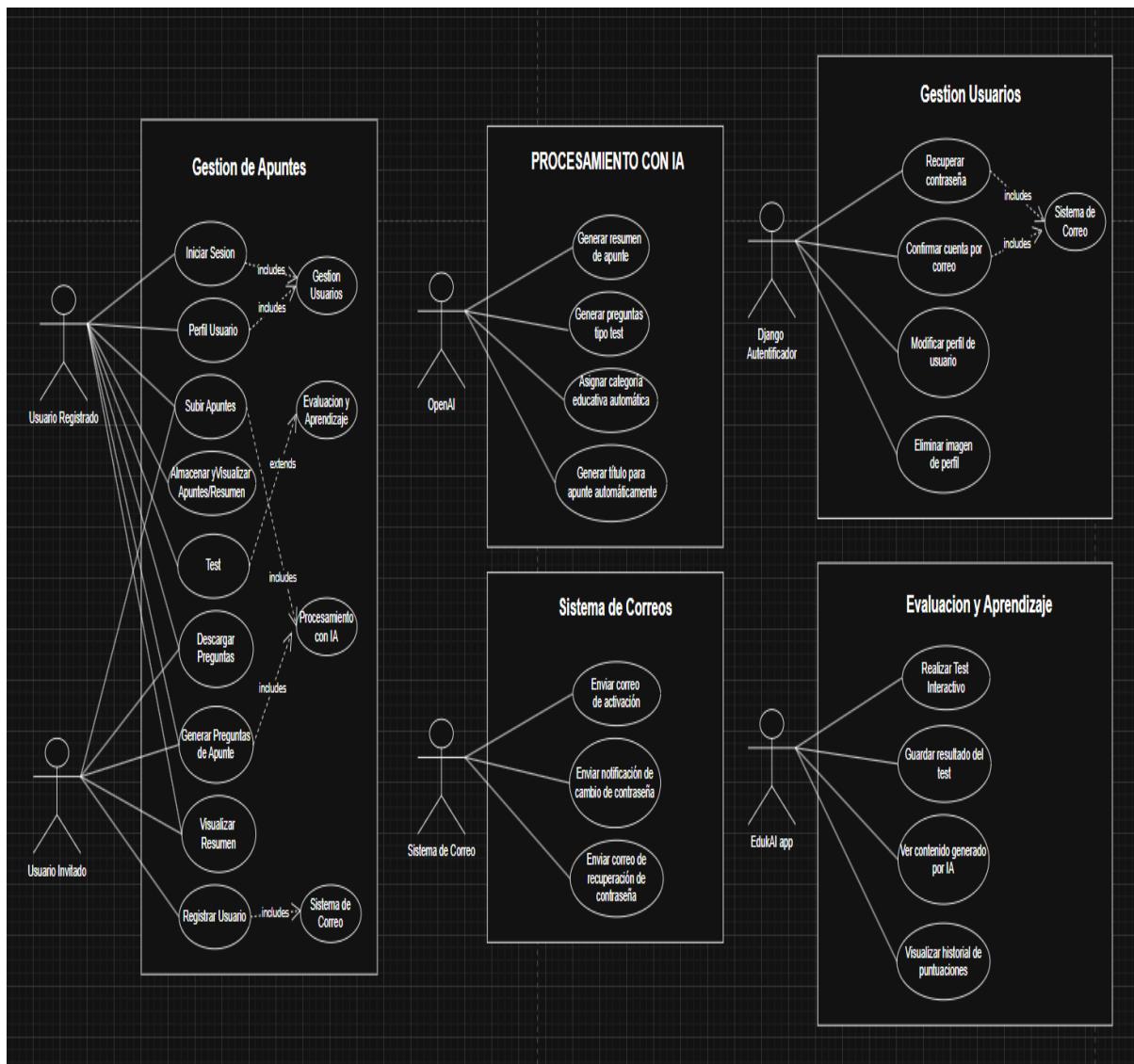
- Centraliza toda la información de la plataforma:
 - Usuarios y perfiles.
 - Archivos PDF subidos.
 - Resúmenes y preguntas generadas.
 - Puntuaciones obtenidas en los tests.

Casos de uso. Incluye diagrama y tabla con:

- Descripción.
- Precondiciones
- Postcondiciones
- Datos de entrada
- Datos de salida
- Tablas
- Clases
- Interfaces

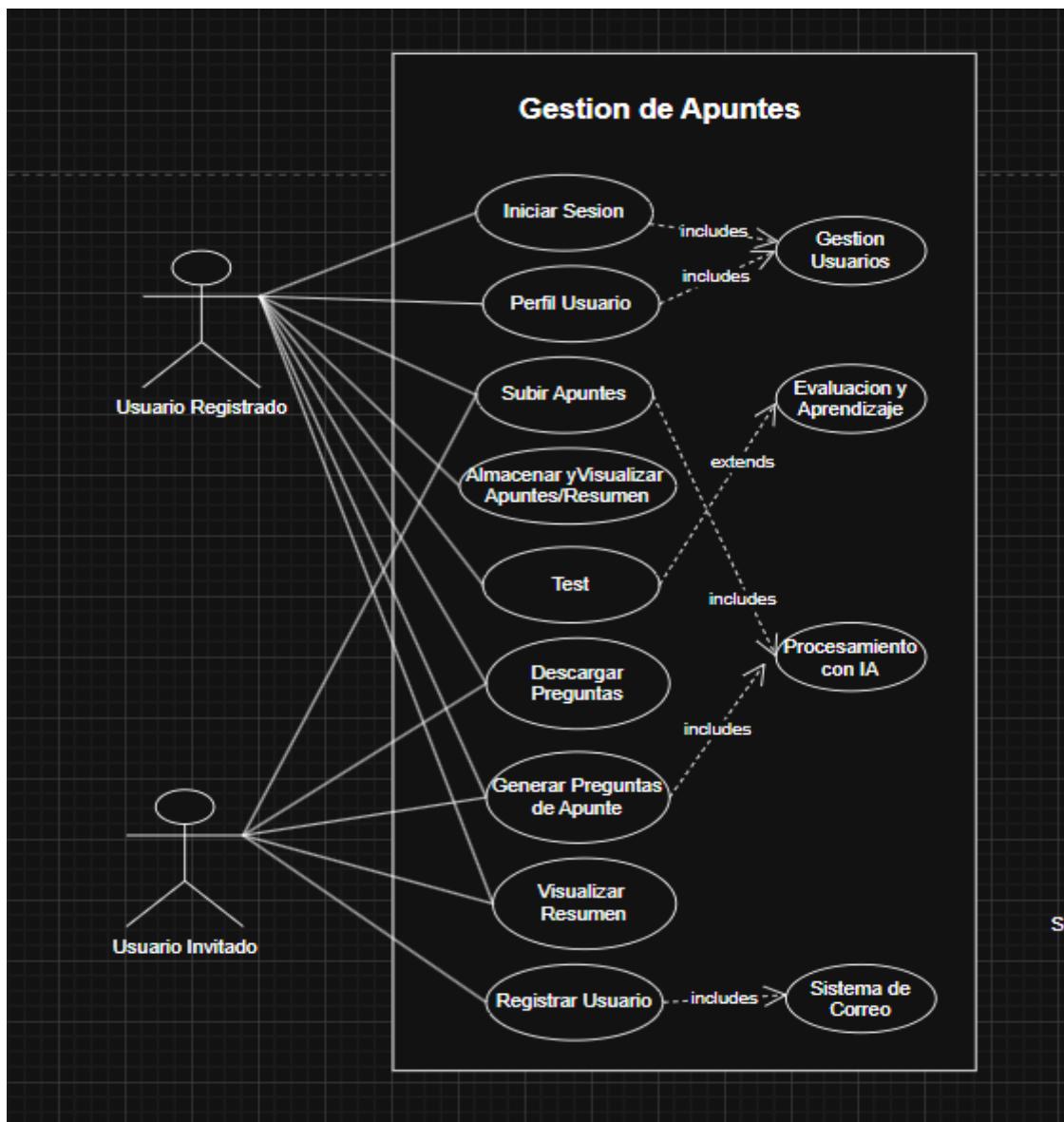
Diagrama Global de Casos de Uso

El siguiente diagrama global representa una visión general de todos los casos de uso contemplados en la plataforma EdukAI. En él se muestran los principales actores del sistema y las funcionalidades clave que pueden ejecutar, organizadas por módulos como gestión de usuarios, apuntes, procesamiento con IA, evaluación y sistema de correos. Este diagrama permite comprender de forma rápida y estructurada el alcance funcional del proyecto antes de profundizar en cada caso de uso específico.



Gestión de Apuntes

En la siguiente sección, se presenta el **diagrama del caso de uso “Gestión de Apuntes”**, donde se ilustran claramente los **dos actores principales: Usuario Registrado y Usuario Invitado**, junto con los distintos casos de uso que cada uno puede ejecutar dentro del sistema. Este módulo incluye operaciones como subir documentos, visualizar contenido, generar preguntas, iniciar tests y descargar información relevante. Es el núcleo operativo de la aplicación, ya que conecta directamente con los procesos de inteligencia artificial y aprendizaje automático implementados.



Actor: Usuario Registrado

La siguiente tabla recoge los casos de uso asociados al actor **Usuario Registrado** dentro del módulo *Gestión de Apuntes* de EdukAI. Estos casos de uso representan las acciones principales que puede realizar un usuario autenticado, como registrar una cuenta, iniciar sesión, subir y visualizar apuntes, realizar tests, generar preguntas mediante IA y descargar contenido. Todas estas funcionalidades quedan asociadas a su perfil personal, lo que permite guardar el historial y mantener persistencia de datos en la plataforma. A continuación detallamos los diferentes casos de uso encontrados.

CASO DE USO: Iniciar Sesión	
DESCRIPCIÓN: El usuario accede al sistema mediante sus credenciales.	
PRECONDICIONES: Usuario registrado y cuenta activa.	POSTCONDICIONES: Sesión iniciada con token de autenticación.
DATOS ENTRADA Email, contraseña.	DATOS SALIDA Token de acceso.
TABLAS: user	CLASES: Usuario, Autenticador
INTERFACES: Login.html	

CASO DE USO: Perfil Usuario	
DESCRIPCIÓN: Permite modificar nombre, avatar, contraseña u otros datos del perfil.	
PRECONDICIONES: Usuario autenticado.	POSTCONDICIONES: Perfil actualizado en base de datos.
DATOS ENTRADA ID usuario, nuevos datos.	DATOS SALIDA Confirmación de actualización.
TABLAS:	CLASES:

user	Usuario, Perfil
INTERFACES: Perfil.html	

CASO DE USO: Subir Apuntes	
DESCRIPCIÓN: El usuario sube un apunte en PDF o texto para almacenarlo y procesarlo	
PRECONDICIONES: Usuario autenticado o invitado.	POSTCONDICIONES: El apunte se guarda en la base de datos.
DATOS ENTRADA Archivo PDF o texto, título.	DATOS SALIDA Confirmación, ID apunte, resumen.
TABLAS: apuntes	CLASES: Apunte, Categoría
INTERFACES: Subir_Apunte.html	

CASO DE USO: Almacenar y Visualizar Apuntes/Resumen	
DESCRIPCIÓN: Muestra apuntes subidos y permite almacenar o ver resumen generado	
PRECONDICIONES: Apunte o resumen debe existir.	POSTCONDICIONES: Visualización correcta del apunte y resumen.
DATOS ENTRADA ID user, ID de apunte o resumen.	DATOS SALIDA Contenido PDF o texto generado.
TABLAS: apuntes, categoría	CLASES: Apunte, Categoría, User
INTERFACES: Mis_apuntes.html	

CASO DE USO: Test	
DESCRIPCIÓN: El usuario inicia un test basado en las preguntas generadas.	
PRECONDICIONES: Preguntas disponibles para un apunte.	POSTCONDICIONES: Inicio del test interactivo.
DATOS ENTRADA ID usuario, ID apunte.	DATOS SALIDA Inicio del test interactivo.
TABLAS: preguntas, apuntes	CLASES: Apunte,Pregunta
INTERFACES: test_interactivo.html	

CASO DE USO: Descargar Preguntas	
DESCRIPCIÓN: Permite al usuario descargar las preguntas generadas en PDF o similar.	
PRECONDICIONES: Preguntas generadas previamente.	POSTCONDICIONES: Archivo descargado por el usuario.
DATOS ENTRADA ID apunte, ID pregunta.	DATOS SALIDA Archivo PDF.
TABLAS: preguntas	CLASES: Pregunta, ExportadorPDF(pdfgen canvas)
INTERFACES: detalle_test.html	

CASO DE USO: Generar Preguntas de Apunte	
DESCRIPCIÓN: Genera preguntas tipo test a partir del apunte usando IA.	
PRECONDICIONES: Apunte cargado en el sistema.	POSTCONDICIONES: Preguntas disponibles (15).

DATOS ENTRADA	DATOS SALIDA
Texto del apunte.	Lista de preguntas.
TABLAS: preguntas	CLASES: Pregunta, OpenAi,json.load
INTERFACES: preguntas.html	

CASO DE USO: Visualizar Resumen	
DESCRIPCIÓN: Permite al usuario ver el resumen generado de un apunte.	
PRECONDICIONES: Resumen generado y asociado al apunte.	POSTCONDICIONES: Resumen mostrado al usuario.
DATOS ENTRADA ID resumen o apunte.	DATOS SALIDA Contenido del resumen.
TABLAS: Apunte	CLASES: Apunte,OpenAI
INTERFACES: detalle_resumen.html	

Actor : Usuario Invitado

Estas tablas describen los casos de uso disponibles para el **Usuario Invitado**, un actor que accede a EdukAI sin necesidad de registrarse. Sus interacciones con la plataforma son limitadas y temporales: puede subir apuntes, generar resúmenes o preguntas, y visualizar el contenido generado, pero sin almacenar resultados ni vincular acciones a un perfil. Este enfoque permite a nuevos usuarios explorar las funcionalidades principales de la plataforma de forma rápida antes de registrarse. A continuación detallamos los diferentes casos de uso que se puede encontrar este actor.

CASO DE USO: Registrar Usuario	
DESCRIPCIÓN: Permite crear una cuenta con envío de correo de confirmación.	
PRECONDICIONES: Formulario de registro completado.	POSTCONDICIONES: Usuario registrado en la base de datos.
DATOS ENTRADA Nombre, email, nivel academicico contraseña.	DATOS SALIDA Token de activación.
TABLAS: User	CLASES: Usuario, Registro, EmailToken
INTERFACES: Register.html	

CASO DE USO: Subir Apunte (PDF)	
DESCRIPCIÓN: El usuario invitado sube un archivo temporal para generar un resumen o preguntas.	
PRECONDICIONES: Acceso como invitado habilitado.	POSTCONDICIONES: El apunte se usa por OpenAI de forma temporal sin vinculación a un perfil.
DATOS ENTRADA Archivo PDF o texto.	DATOS SALIDA Generacion resumen, preguntas, confirmación.
TABLAS: ninguna (uso en sesión)	CLASES: OpenAI
INTERFACES: dashboard.html	

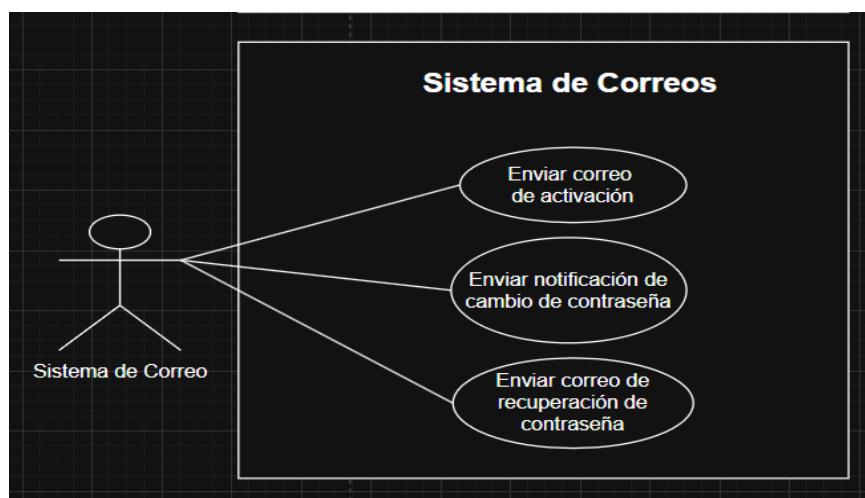
CASO DE USO: Generar resumen	
DESCRIPCIÓN: Genera un resumen por IA con apunte subido como invitado.	
PRECONDICIONES: Apunte cargado.	POSTCONDICIONES: Resumen generado sin guardar historia, solo visualizacion.
DATOS ENTRADA Archivo PDF o texto.	DATOS SALIDA Generacion resumen
TABLAS: ninguna (uso en sesión)	CLASES: OpenAI
INTERFACES: dashboard.html	

CASO DE USO: Generar Preguntas	
DESCRIPCIÓN: Crea preguntas tipo test a partir del contenido subido sin guardarlas permanentemente.	
PRECONDICIONES: Acceso como invitado habilitado.	POSTCONDICIONES: Preguntas generas no mostradas.
DATOS ENTRADA Archivo PDF o texto.	DATOS SALIDA Lista de preguntas en PDF para descargar.
TABLAS: ninguna (uso en sesión)	CLASES: OpenAI, ExportadorPDF(pdfgen canvas)
INTERFACES: dashboard.html (sección descargar preguntas)	

CASO DE USO: Visualizar resumen	
DESCRIPCIÓN: Permite al invitado visualizar el resumen generado.	
PRECONDICIONES: Resumen generado	POSTCONDICIONES: Resumen mostrado.
DATOS ENTRADA Resumen generado por la IA	DATOS SALIDA Contenido del resumen.
TABLAS: ninguna (uso en sesión)	CLASES: OpenAI
INTERFACES: dashboard.html (sección leer resumen)	

Sistema de Correos

En la siguiente sección se presenta el diagrama del caso de uso correspondiente al módulo “Sistema de Correos”. Este componente actúa como intermediario en las operaciones de comunicación automática entre la plataforma EdukAI y el usuario, mediante el envío de mensajes por correo electrónico. Incluye funcionalidades clave como el envío del correo de activación tras el registro, la recuperación de contraseña mediante token y la notificación de cambios críticos en la cuenta del usuario. A continuación se explica los casos de uso principales de este módulo.



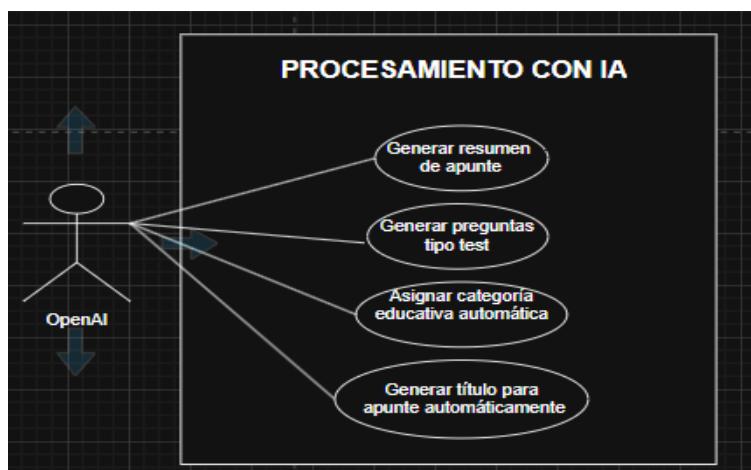
CASO DE USO: Enviar correo de activación	
DESCRIPCIÓN: El sistema envía un correo con un enlace de activación para validar una nueva cuenta registrada.	
PRECONDICIONES: Usuario ha completado el registro.	POSTCONDICIONES: El correo es enviado correctamente al email del usuario..
DATOS ENTRADA Email del usuario, token de activación.	DATOS SALIDA Confirmación de envío o error.
TABLAS: user, tokens	CLASES: EmailSender, TokenGenerator
INTERFACES: register.html, account_activation.html	

CASO DE USO: Enviar notificación de cambio de contraseña	
DESCRIPCIÓN: Se envía un correo al usuario notificando que su contraseña ha sido modificada.	
PRECONDICIONES: Cambio de contraseña ejecutado correctamente.	POSTCONDICIONES: Notificación enviada.
DATOS ENTRADA Email del usuario, fecha/hora del cambio.	DATOS SALIDA Confirmación de envío..
TABLAS: user	CLASES: emailSender,
INTERFACES: perfil.html	

CASO DE USO: Enviar correo de recuperación de contraseña	
DESCRIPCIÓN: Envía un correo con un enlace para restablecer la contraseña del usuario.	
PRECONDICIONES: Usuario ha solicitado recuperar el acceso.	POSTCONDICIONES: Correo de recuperación enviado.
DATOS ENTRADA Email del usuario	DATOS SALIDA Token de recuperación, confirmación.
TABLAS: User ,token	CLASES: EmailSender, TokenGenerator
INTERFACES: password_reset.html	

Procesamiento con IA

En la siguiente sección se presenta el diagrama del caso de uso correspondiente al módulo “Procesamiento con IA”. Este bloque representa uno de los aspectos más innovadores de EdukAI, ya que permite aplicar inteligencia artificial para transformar los apuntes del usuario en contenido útil para el estudio. Entre sus funcionalidades se encuentran la generación automática de resúmenes, preguntas tipo test, sugerencias de categorías educativas y títulos para los documentos subidos. Estas acciones son ejecutadas por modelos de lenguaje natural conectados a través de OpenAI,



CASO DE USO: Generar resumen de apunte	
DESCRIPCIÓN: La IA recibe el contenido del apunte y devuelve un resumen automático del mismo.	
PRECONDICIONES: PDF o texto cargado correctamente.	POSTCONDICIONES: Preguntas generadas y mostradas.
DATOS ENTRADA Texto de apuntes	DATOS SALIDA Texto resumido
TABLAS: apuntes	CLASES: openAI, apunte
INTERFACES: subir_apunte.html	

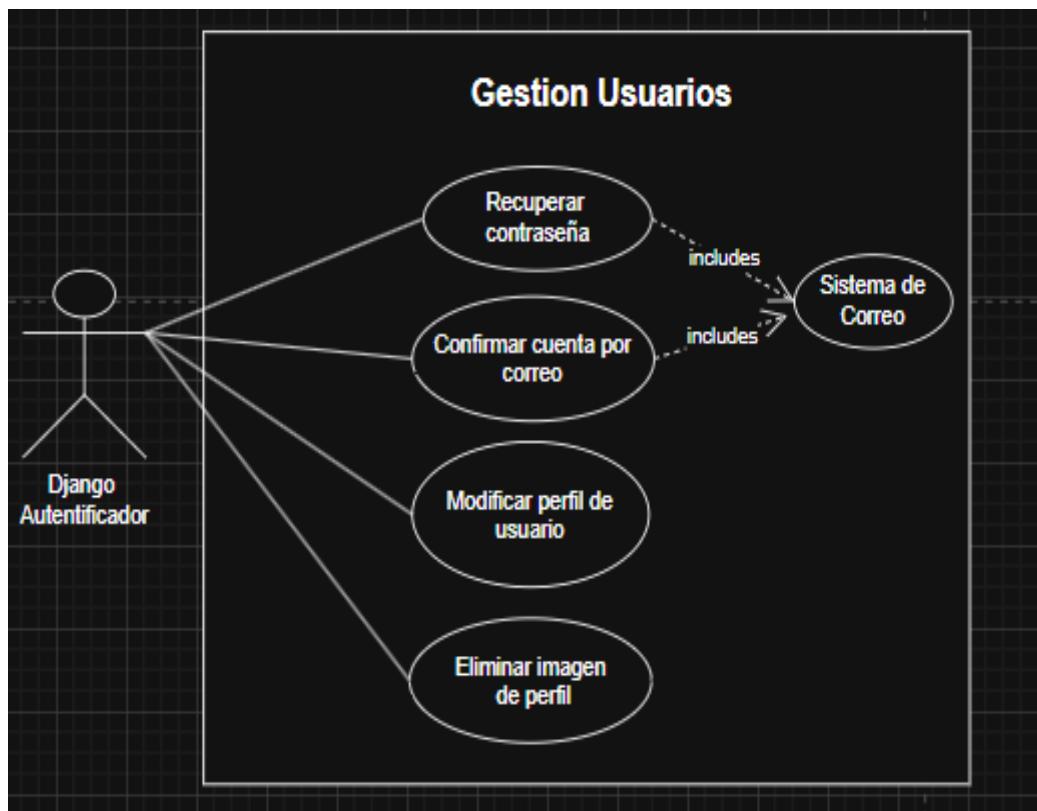
CASO DE USO: Generar preguntas tipo test	
DESCRIPCIÓN: La IA analiza el texto subido y genera automáticamente preguntas tipo test.	
PRECONDICIONES: Apunte cargado correctamente.	POSTCONDICIONES: Resumen generado y disponible para visualización o descarga.
DATOS ENTRADA Contenido del pdf o texto	DATOS SALIDA Lista de preguntas tipo test.
TABLAS: Apunte,pregunta	CLASES: openAI, apunte,pregunta,json.loads
INTERFACES: preguntas.html	

CASO DE USO: Asignar categoría educativa automática	
DESCRIPCIÓN: La IA analiza el contenido del apunte y sugiere una categoría educativa relevante.	
PRECONDICIONES: Apunte cargado correctamente sin categoría definida	POSTCONDICIONES: Categoría sugerida asignada al apunte.
DATOS ENTRADA Contenido del pdf o texto	DATOS SALIDA Categoría recomendada.
TABLAS: Apunte, categoría	CLASES: openAI, apunte, categoría
INTERFACES: subir_apunte.html, dashboard.html	

CASO DE USO: Generar título para apunte automáticamente	
DESCRIPCIÓN: A partir del contenido subido, la IA sugiere un título adecuado para el apunte.	
PRECONDICIONES: Apunte cargado sin título definido.	POSTCONDICIONES: Título generado y asociado al apunte.
DATOS ENTRADA Contenido del pdf o texto	DATOS SALIDA Título generado.
TABLAS: Apunte	CLASES: openAI, apunte
INTERFACES: subir_apunte.html	

Gestión de Usuarios

En esta sección se presenta el módulo “Gestión de Usuarios”, encargado de manejar las operaciones relacionadas con la identidad y configuración de los usuarios en EdukAI. Entre sus funcionalidades principales se encuentran la recuperación de contraseña, la activación de cuentas mediante confirmación por correo electrónico, y la edición del perfil de usuario. Además, está integrado con el sistema de correos para notificaciones automáticas.



En las siguientes tablas se explica los diferentes casos de uso que nos encontramos con esta sección .

CASO DE USO: Recuperar contraseña	
DESCRIPCIÓN: Permite al usuario recuperar su cuenta mediante el envío de un enlace de restablecimiento al correo electrónico.	
PRECONDICIONES: El usuario debe haber solicitado recuperar el acceso.	POSTCONDICIONES: El sistema envía un correo con un token para restablecer la contraseña.
DATOS ENTRADA Email usuario	DATOS SALIDA Token de recuperación, enlace de restablecimiento.
TABLAS: user, tokens	CLASES: User, Token, EmailSender
INTERFACES: reset_password.html	

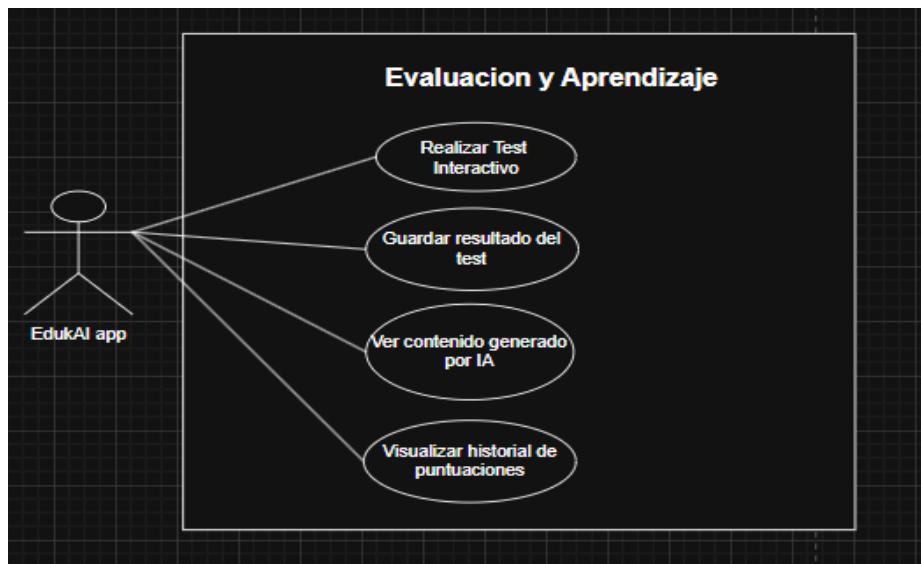
CASO DE USO: Confirmar cuenta por correo	
DESCRIPCIÓN: Valida una nueva cuenta mediante un token de activación enviado por email tras el registro.	
PRECONDICIONES: Usuario registrado con correo pendiente de activación.	POSTCONDICIONES: Cuenta activada y lista para iniciar sesión.
DATOS ENTRADA Email, token de activación.	DATOS SALIDA Confirmación de cuenta activada.
TABLAS: user, tokens	CLASES: User
INTERFACES: register.html,	

CASO DE USO: Modificar perfil de usuario	
DESCRIPCIÓN: Permite al usuario registrado actualizar sus datos personales como nombre, estudios, o imagen o elegir un avatar de la app y password.	
PRECONDICIONES: Sesión iniciada correctamente.	POSTCONDICIONES: Datos actualizados en el perfil del usuario.
DATOS ENTRADA ID usuario, nuevos datos (nombre, avatar, estudios,password) .	DATOS SALIDA Confirmación de cambio.
TABLAS: user, perfil	CLASES: User, Perfil
INTERFACES: perfil.html.	

CASO DE USO: Eliminar imagen de perfil	
DESCRIPCIÓN: Elimina la imagen de perfil actual del usuario dejando el campo vacío.	
PRECONDICIONES: Usuario autenticado con imagen configurada.	POSTCONDICIONES: Campo imagen vacío y perfil actualizado.
DATOS ENTRADA ID usuario,	DATOS SALIDA Confirmación de eliminación.
TABLAS: User,perfil	CLASES: User, Perfil
INTERFACES: perfil.html.	

Evaluación y Aprendizaje

En esta sección se presenta el módulo “Evaluación y Aprendizaje”, enfocado en proporcionar al usuario herramientas para medir y reforzar sus conocimientos dentro de la plataforma EdukAI. A través de este bloque funcional, los usuarios pueden realizar tests interactivos basados en sus apuntes, guardar sus resultados



CASO DE USO: Realizar Test Interactivo

DESCRIPCIÓN: Permite al usuario responder un conjunto de preguntas generadas previamente de forma dinámica.

PRECONDICIONES: Preguntas generadas y visibles.	POSTCONDICIONES: Respuestas procesadas y test completado.
DATOS ENTRADA Respuestas del usuario, ID del test o apunte.	DATOS SALIDA Resultado del test (puntaje).
TABLAS: preguntas, resultado_test	CLASES: Test, ResultadoTest
INTERFACES: test_interactivo.html.	

CASO DE USO: Guardar resultado test	
DESCRIPCIÓN: Al finalizar el test, el sistema guarda el puntaje y lo asocia al usuario registrado.	
PRECONDICIONES: Test completado	POSTCONDICIONES: Resultado guardado en base de datos.
DATOS ENTRADA ID usuario, puntaje, fecha,test	DATOS SALIDA Confirmación de almacenamiento.
TABLAS: preguntas, resultado_test	CLASES: Test, ResultadoTest
INTERFACES: historial_test.html.	

CASO DE USO: Ver contenido generado por IA	
DESCRIPCIÓN: Muestra al usuario tips, sugerencias o contenido generado por IA como parte del bloque educativo del dashboard.	
PRECONDICIONES: Acceso autenticado al dashboard.	POSTCONDICIONES: Contenido visualizado.
DATOS ENTRADA N/A	DATOS SALIDA Texto generado dinámicamente.
TABLAS: openAI	CLASES: OpenAI
INTERFACES: dashboard.html	

CASO DE USO: Visualizar historial de puntuaciones	
DESCRIPCIÓN: El usuario puede consultar los puntajes obtenidos en tests anteriores para evaluar su progreso.	
PRECONDICIONES: Usuario registrado con historial disponible.	POSTCONDICIONES: Historial mostrado.
DATOS ENTRADA ID user	DATOS SALIDA Lista de tests con fechas y puntajes.
TABLAS: resultados_test	CLASES: Resultados_test
INTERFACES: historial_tests.html	

Diagrama de clases.

En este proyecto se ha utilizado el framework **Django**, el cual sigue el patrón de diseño **MTV (Model–Template–View)**, similar al clásico **MVC (Modelo–Vista–Controlador)**. Por este motivo, la lógica del sistema y su estructura se han organizado en **componentes separados**, cada uno con responsabilidades bien definidas , en el siguiente apartado mostramos el diagrama de clases (Modelos) y la tabla con las vistas y funciones principales del sistema, aunque están separados conceptualmente, **las vistas interactúan directamente con los modelos** para consultar, crear, modificar o eliminar datos. Esta separación permite:

- Mayor claridad en la arquitectura
- Mejor mantenimiento del código
- Reutilización de componentes

Diagrama de Clases – Modelos

El **diagrama de clases** representa la estructura de los datos del sistema, es decir, los **modelos definidos en models.py**. Estas clases corresponden directamente a las tablas de la base de datos y se encargan de almacenar, validar y manipular la información relacionada con:

- Usuarios y perfiles
 - Apuntes, resúmenes, preguntas
 - Resultados de test y categorías
- ◇ Esta parte refleja la **capa de modelo (M)** de Django.

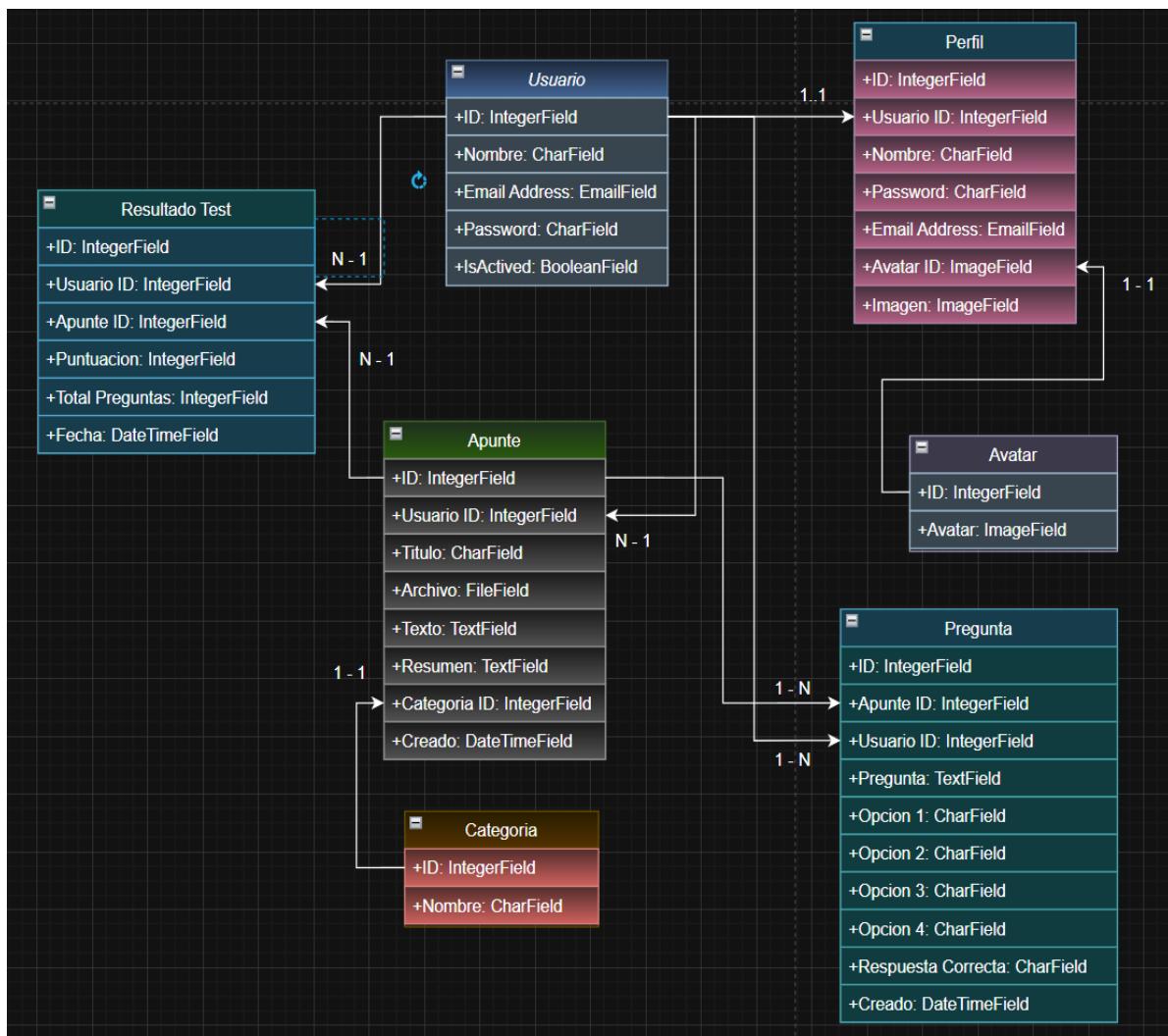


Tabla de Vistas y Funciones

Por otro lado, la **tabla de vistas y funciones** documenta cómo se gestiona la lógica del sistema desde las funciones y clases definidas en views.py. Estas vistas:

- Procesan las peticiones del usuario
 - Llaman a los modelos y servicios (como la IA o los correos)
 - Renderizan las respuestas mediante plantillas HTML
- ◊ Esto corresponde a la **capa de vista (V)** en Django, que en este caso **controla el flujo** del sistema

Vista/Funcion	Modelos Usados	Descripción Breve
register_view	Usuario, PerfilUsuario	Registra un nuevo usuario, crea perfil asociado y envía correo de activación.
reenviar_activacion_view / enviar_activacion	Usuario, Token	Reenvía un nuevo enlace de activación a usuarios inactivos.
activar_cuenta	Usuario, Token	Activa la cuenta del usuario al verificar el token recibido por correo.
perfil_view	PerfilUsuario, Usuario	Permite al usuario editar su perfil, cambiar imagen o contraseña.
subir_apunte_view	Apunte, Categoria	Permite al usuario subir un apunte y genera resumen, categoría y título automáticamente con IA.
generar_preguntas_view	Apunte, Pregunta	Genera preguntas tipo test mediante IA y las guarda en base de datos.
guardar_resultado_test	ResultadoTest, Apunte	Guarda el resultado del test, actualiza si la nueva puntuación es mayor.
historial_resultados_view	ResultadoTest, Apunte	Muestra al usuario el historial de tests realizados y sus puntajes.
exportar_test_pdf	Apunte, Pregunta	Exporta el test generado en formato PDF con las respuestas correctas.
dashboard_view / mis_apuntes_view / resumenes_view	Apunte	Muestran los apuntes subidos por el usuario en distintas vistas filtradas.

Diagrama E/R (Entidad - Relación)

El diagrama muestra la estructura lógica de la base de datos de EdukAI, representando las entidades principales del sistema (usuarios, apuntes, preguntas, resultados, etc.) y sus relaciones.

Destaca la relación funcional 1:1 entre Apunte y Resultado Test por usuario, controlada mediante restricciones internas, así como la organización de perfiles, categorías y resúmenes vinculados a los apuntes.

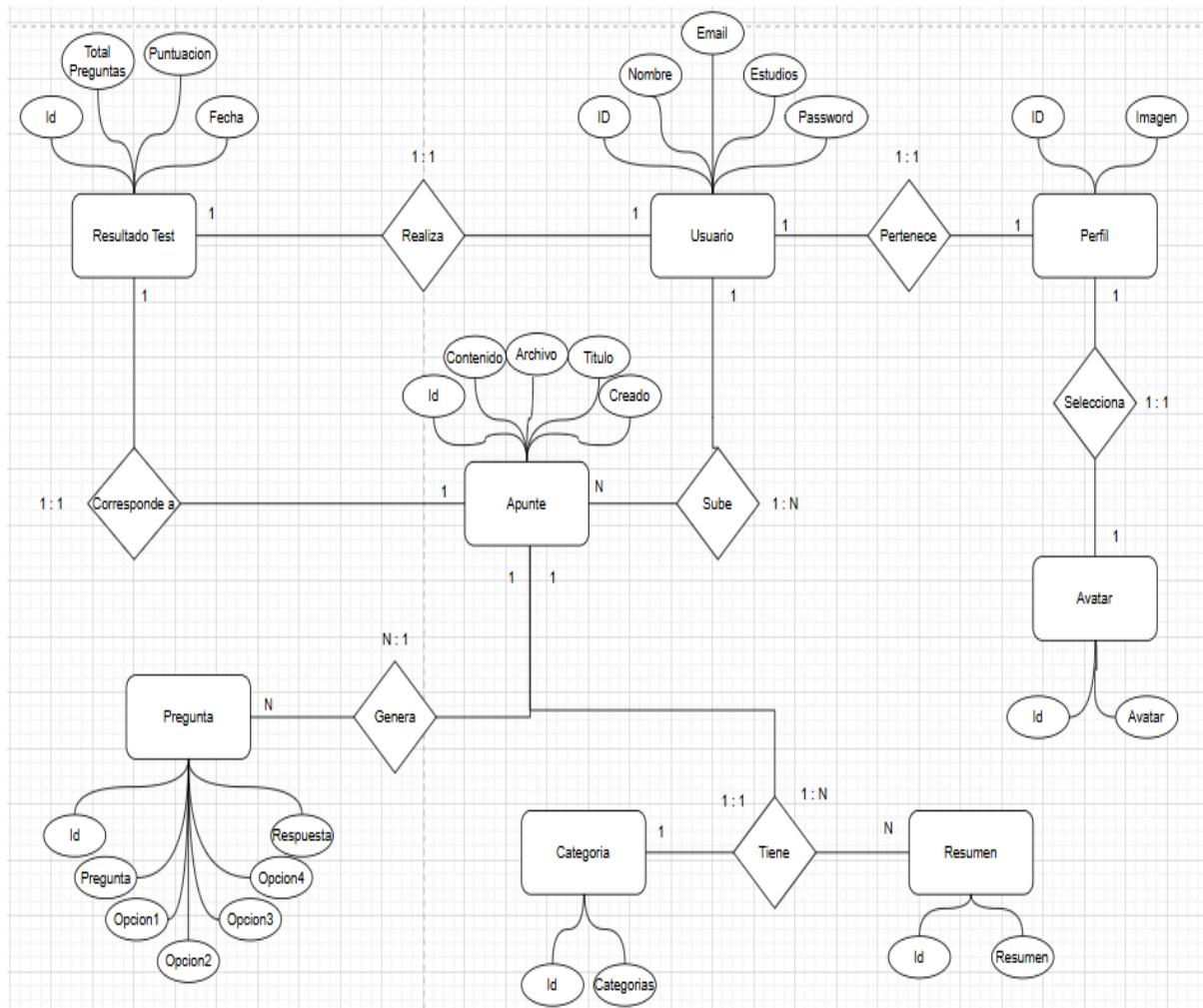


Diagrama de Base de Datos – EdukAI

El siguiente diagrama representa la estructura relacional de la base de datos de la plataforma EdukAI. En él se muestran las tablas principales del sistema, sus atributos clave y las relaciones entre entidades.

Destaca especialmente la relación entre Apunte y Resultado_Test, que aunque técnicamente es de uno a muchos (1 : N), en la app se comporta como una relación 1:1 gracias a una restricción lógica que impide que un mismo usuario registre más de un resultado por apunte este resultado solo se actualiza si el usuario alcanza una puntuación mayor.

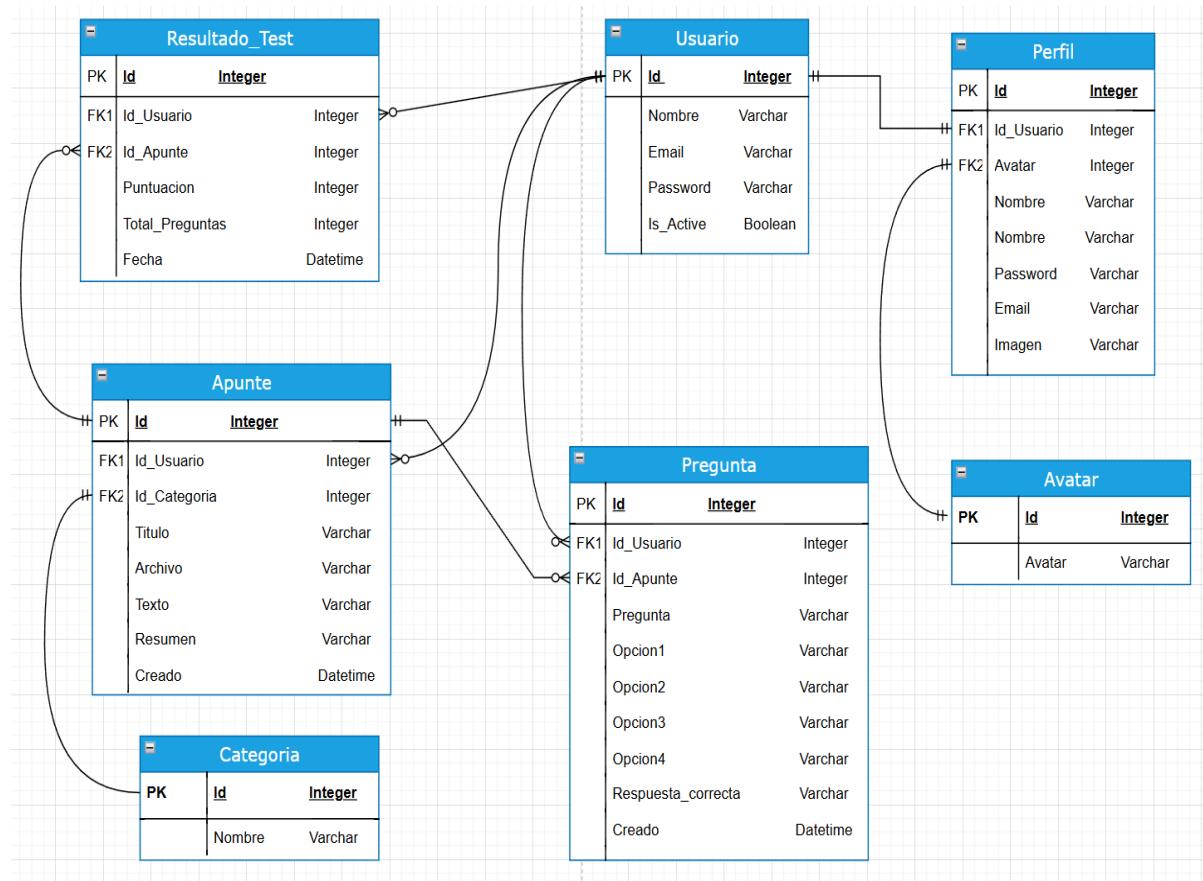
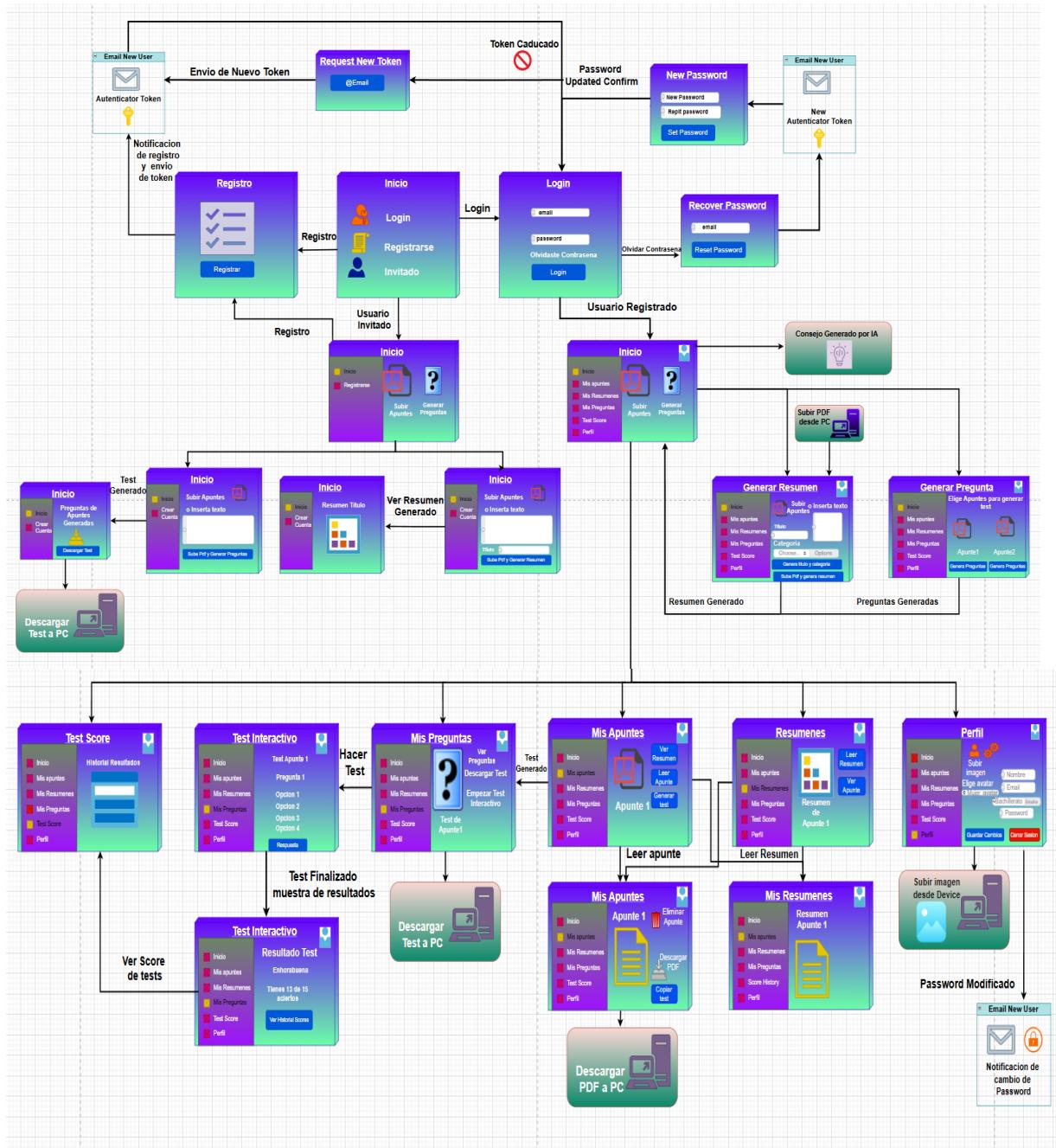
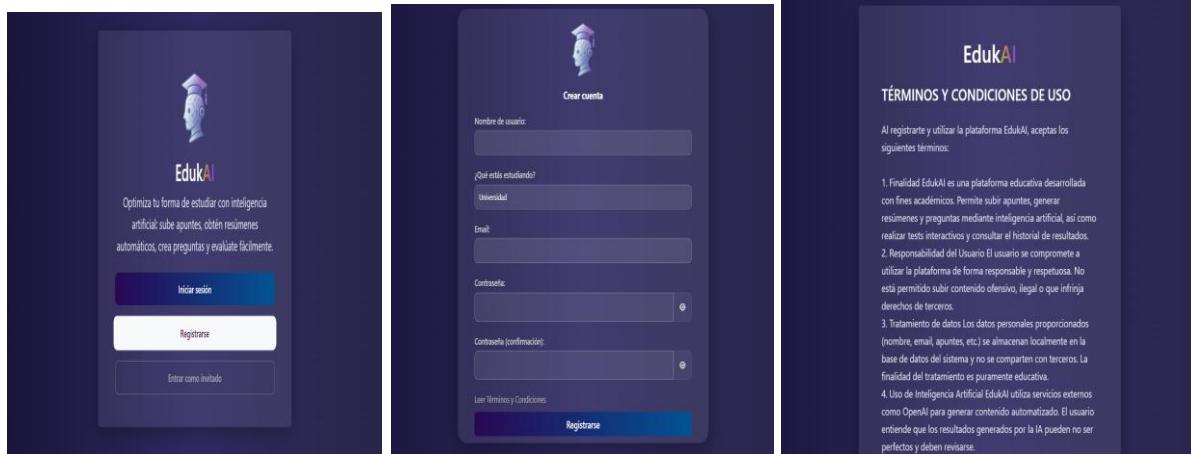


Diagrama de Flujo de Navegación : El siguiente diagrama muestra el flujo de navegación de la aplicación EdukAI representando cómo se accede a las distintas funcionalidades de la plataforma, desde el registro e inicio de sesión hasta la subida de apuntes, generación de resúmenes y preguntas, realización de tests y gestión del perfil. Además, se incluye el flujo especial para la recuperación de contraseña y el envío del token de activación.

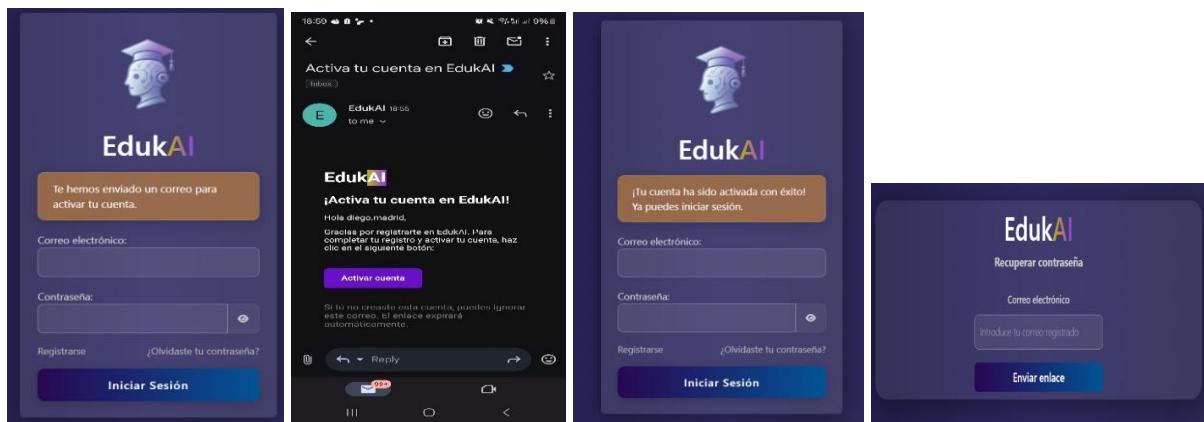


Interfaces de la Aplicacion.

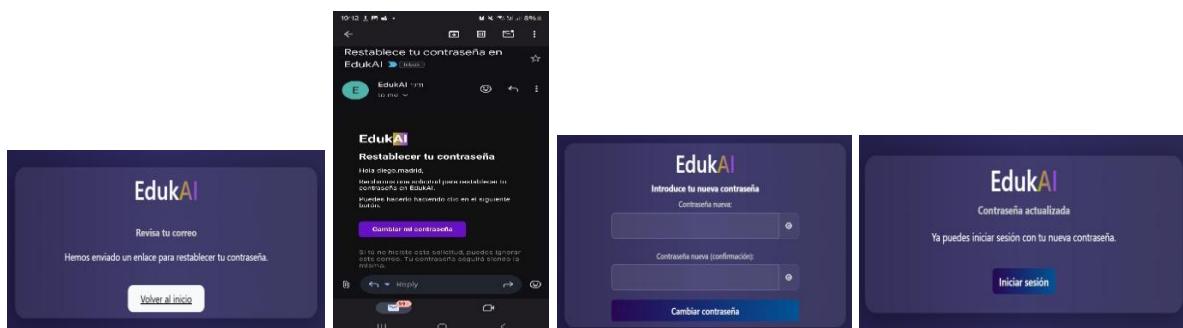
Interfaz de bienvenida: sección de registro, términos y condiciones



Una vez registrado el usuario recibe un email de activación. Esto lo redirige a login. El login tiene un enlace a recuperar password



Al introducir el email el usuario recibe un email de Edukai con enlace para cambiar el password de la cuenta si el password es válida se notifica y envia a login.



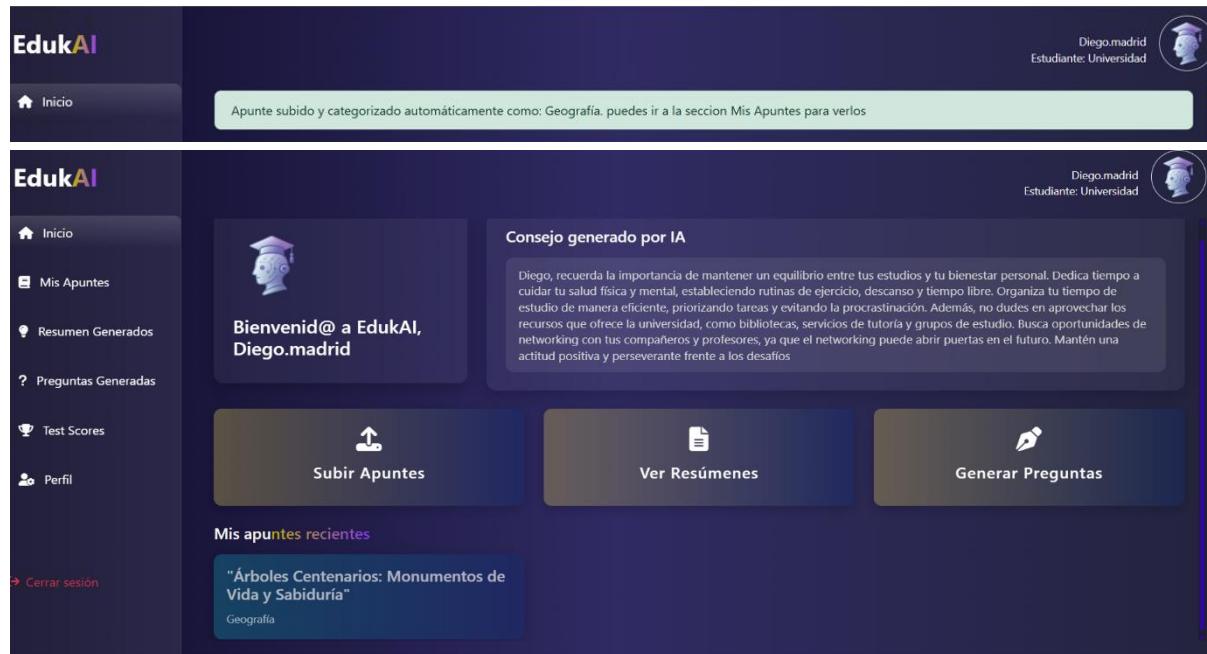
Dashboard de la aplicación muestra consejos al usuario con las secciones de la aplicación al ser un nuevo usuario no muestra los apuntes subidos recientemente

The screenshot shows the EdukAI dashboard. On the left is a sidebar with links: Inicio, Mis Apuntes, Resumen Generados, Preguntas Generadas, Test Scores, Perfil, and Cerrar sesión. The main area has a dark background. It features a welcome message from 'Bienvenid@ a EdukAI, Diego.madrid' with an AI icon. Below it are three buttons: 'Subir Apuntes' (Upload Notes) with an up arrow icon, 'Ver Resúmenes' (View Summaries) with a document icon, and 'Generar Preguntas' (Generate Questions) with a pen icon. A section titled 'Consejo generado por IA' (AI-generated advice) provides general study tips. At the bottom, a box says 'Mis apuntes recientes' (Recent notes) and 'No has subido ningún apunte todavía.'

En la sección **Subir Apuntes**, se permite subir un archivo PDF desde el dispositivo o, si se prefiere, introducir un texto manualmente. Además, existe la opción de introducir el título y la categoría de forma manual o dejar que la IA los genere automáticamente.

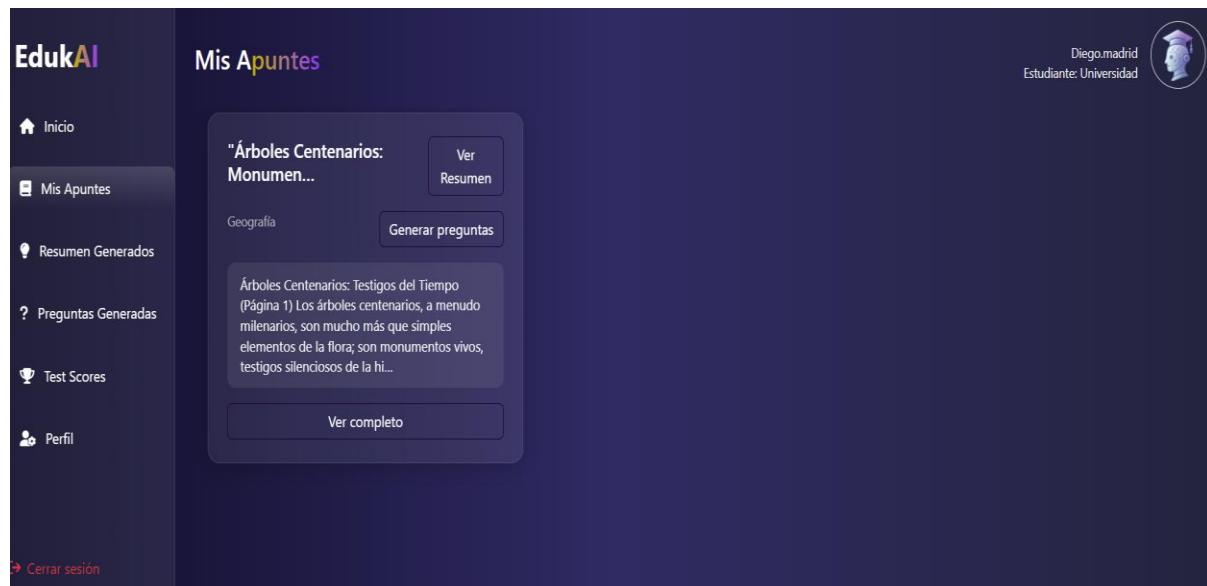
The screenshots show the 'Subir Apuntes' (Upload Notes) page. The top screenshot shows the form with options to upload a file or enter text manually, and a 'Subir Archivo' (Upload File) button. The bottom screenshot shows the same form after a note has been generated. It displays the generated title 'Un Mundo de Perros: Compañeros de Vida y Más Allá' and category 'Sistemas Operativos'. It also shows a text input field with 'apunte prueba' and a category selection field with 'Sistemas Operativos'.

Si la subida del apunte es correcta, se redirige al inicio con una notificación durante 30 segundos. Además, se actualiza la sección de "Apuntes Recientes" con el apunte subido.



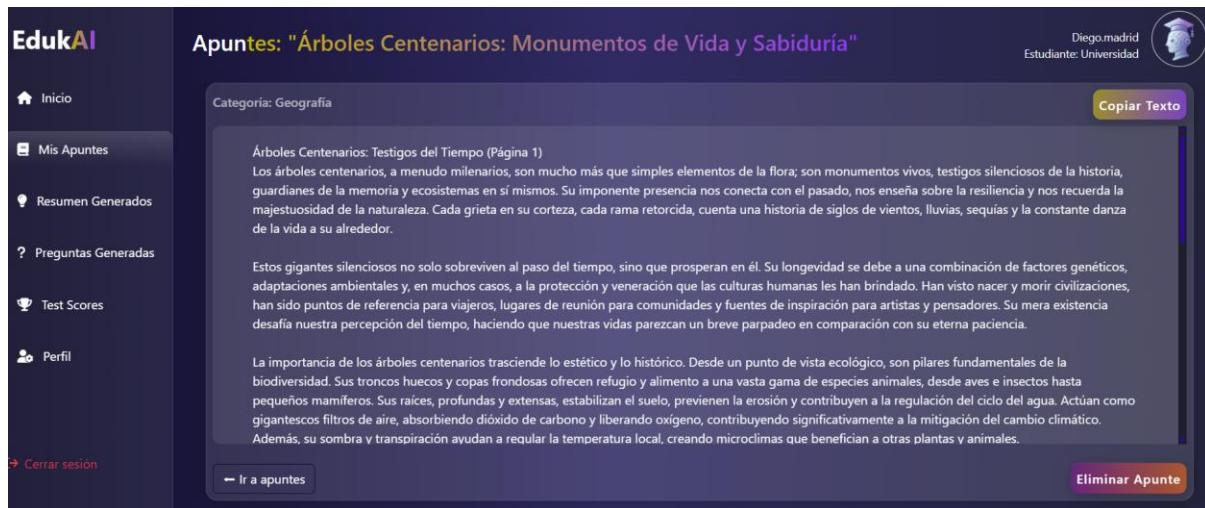
The screenshot shows the EdukAI platform interface. At the top, there's a navigation bar with 'EdukAI' and a user profile icon for 'Diego.madrid'. Below the header, a success message says 'Apunte subido y categorizado automáticamente como: Geografía. puedes ir a la sección Mis Apuntes para verlos'. The main content area has a sidebar with links like 'Inicio', 'Mis Apuntes', 'Resumen Generados', etc. The central area features a 'Consejo generado por IA' box with a message to Diego about maintaining balance between studies and personal well-being. Below it are three buttons: 'Subir Apuntes' (Upload Note), 'Ver Resúmenes' (View Summaries), and 'Generar Preguntas' (Generate Questions). A 'Mis apuntes recientes' (Recent Notes) section displays a note titled 'Árboles Centenarios: Monumentos de Vida y Sabiduría' under the category 'Geografía'.

En la sección **Mis Apuntes**, se pueden ver los apuntes del usuario. Al subir un apunte, se genera también su resumen, que se muestra en la sección **Resumen Generado**. Tenemos botones como “Ver Resumen”, que nos dirige a la sección de detalle del resumen, un botón para generar preguntas con ese apunte y otro botón “Ver Completo” que muestra el texto completo del apunte.



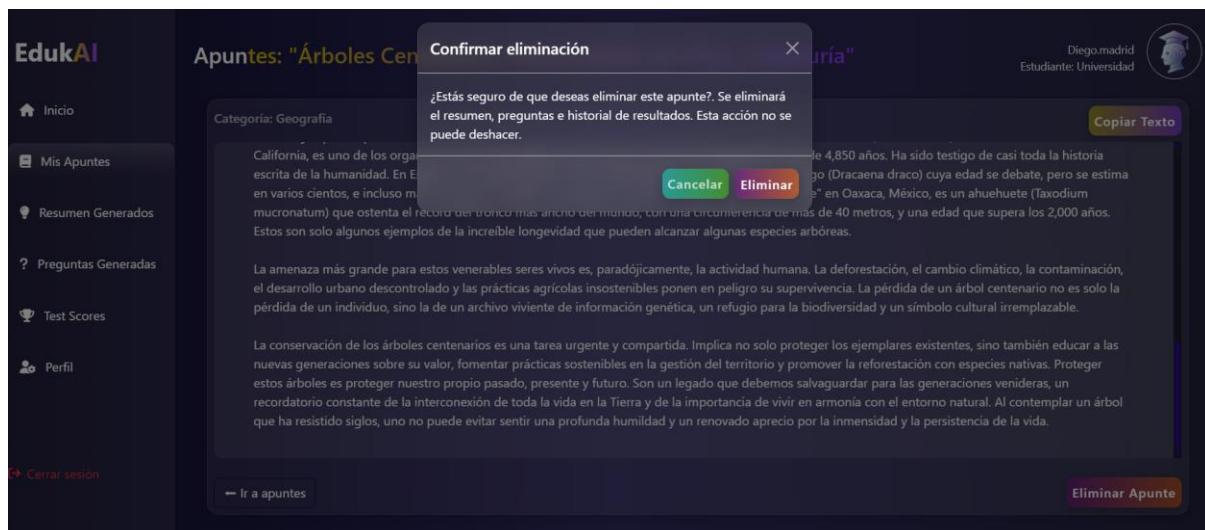
This screenshot shows the 'Mis Apuntes' section of the EdukAI platform. It features a sidebar with links like 'Inicio', 'Mis Apuntes', 'Resumen Generados', etc. The main content area displays a note titled 'Árboles Centenarios: Monumen...' with a 'Ver Resumen' button. Below the note, there are buttons for 'Generar preguntas' (Generate questions) and 'Ver completo' (View full text). The note content itself is partially visible, mentioning 'Árboles Centenarios: Testigos del Tiempo' and 'Los árboles centenarios, a menudo milenarios, son mucho más que simples elementos de la flora; son monumentos vivos, testigos silenciosos de la hi...'. A 'Geografía' tag is also present.

Al presionar el enlace **Ver Completo**, se visualiza el texto con el título, la categoría y el contenido del apunte. Se incluyen botones como **Copiar Texto**, que copia el contenido completo, y **Eliminar Apunte**, que elimina el apunte, su resumen, sus preguntas y su historial de test. Se muestra un mensaje para confirmar la eliminación. También hay un botón **Descargar PDF** para obtener el apunte en su versión original (solo si se subió como PDF). Si se introdujo como texto, esta opción no está disponible.



The screenshot shows the EdukAI interface. On the left is a sidebar with navigation links: Inicio, Mis Apuntes, Resumen Generados, Preguntas Generadas, Test Scores, Perfil, and Cerrar sesión. The main content area has a dark header with the title "Apuntes: 'Árboles Centenarios: Monumentos de Vida y Sabiduría'" and a user profile icon for "Diego.madrid Estudiante: Universidad". Below the header, the note's category is listed as "Geografía". The note content discusses ancient trees as witnesses to history, mentioning their role as guardians of memory and ecosystems. It highlights their adaptability and resilience over thousands of years. A second paragraph continues the discussion on their ecological importance, mentioning their role as refuges for various species. At the bottom of the note are two buttons: "Copiar Texto" (Copy Text) and "Eliminar Apunte" (Delete Note). A small "Ir a apuntes" (Go to notes) button is also visible.

El mensaje de confirmación de eliminación del apunte advierte sobre los riesgos de dicha acción.



This screenshot shows the same EdukAI interface as above, but with a modal dialog box centered over the note content. The dialog is titled "Confirmar eliminación" (Confirm deletion) and contains the message: "¿Estás seguro de que deseas eliminar este apunte?. Se eliminará el resumen, preguntas e historial de resultados. Esta acción no se puede deshacer." (Are you sure you want to delete this note? It will delete the summary, questions, and results history. This action cannot be undone.) At the bottom of the dialog are two buttons: "Cancelar" (Cancel) and "Eliminar" (Delete). The rest of the page, including the sidebar and the note content, is visible in the background.

En la sección **Resúmenes**, se pueden ver los resúmenes generados con la subida de apuntes, con un enlace **Ver Completo** para leer el texto del resumen.

EdukAI

Mis Resúmenes Generados

Diego.madrid
Estudiante: Universidad 

- Inicio
- Mis Apuntes
- Resumen Generados
- Preguntas Generadas
- Test Scores
- Perfil
- Cerrar sesión

Resumen: "Árboles Centenarios: Monumen..."
Geografía

Los árboles centenarios son monumentos vivos que actúan como testigos silenciosos de la historia y como ecosistemas en sí mismos. Su longevidad se debe a factores genéticos y adaptaciones ambientales...

[Ver completo](#)

Interfaz para leer el resumen: incluye un enlace para leer el apunte original.

EdukAI

Resumen: "Árboles Centenarios: Monumentos de Vida y Sabiduría"

Diego.madrid
Estudiante: Universidad 

- Inicio
- Mis Apuntes
- Resumen Generados
- Preguntas Generadas
- Test Scores
- Perfil
- Cerrar sesión

Categoría: Geografía

Los árboles centenarios son monumentos vivos que actúan como testigos silenciosos de la historia y como ecosistemas en sí mismos. Su longevidad se debe a factores genéticos y adaptaciones ambientales, así como a la protección y veneración de las culturas humanas. Estos árboles son fundamentales para la biodiversidad, proporcionando refugio y alimento a diversas especies animales, estabilizando el suelo y contribuyendo a la regulación del ciclo del agua. Culturalmente han sido venerados en muchas civilizaciones y son considerados patrimonio natural y cultural. A pesar de su importancia, la actividad humana representa una amenaza para su supervivencia. La conservación de los árboles centenarios es crucial, involucrando la protección de ejemplares existentes, la educación sobre su valor y la promoción de prácticas sostenibles. Son un legado que debemos preservar para las futuras generaciones, recordándonos la interconexión de toda la vida en la Tierra y la importancia de vivir en armonía con el entorno natural.

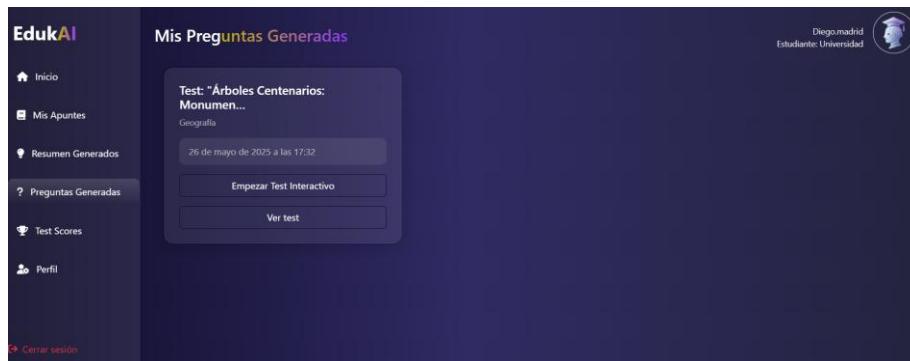
[Leer PDF original](#)

[← Volver a resúmenes](#)

En la sección **Generar Preguntas**, al presionar el botón “Generar Preguntas”, se muestran los apuntes subidos anteriormente. Si se desea generar preguntas sobre apuntes que aún no están en la aplicación, se puede usar un botón que redirige a la sección de subida de apuntes.



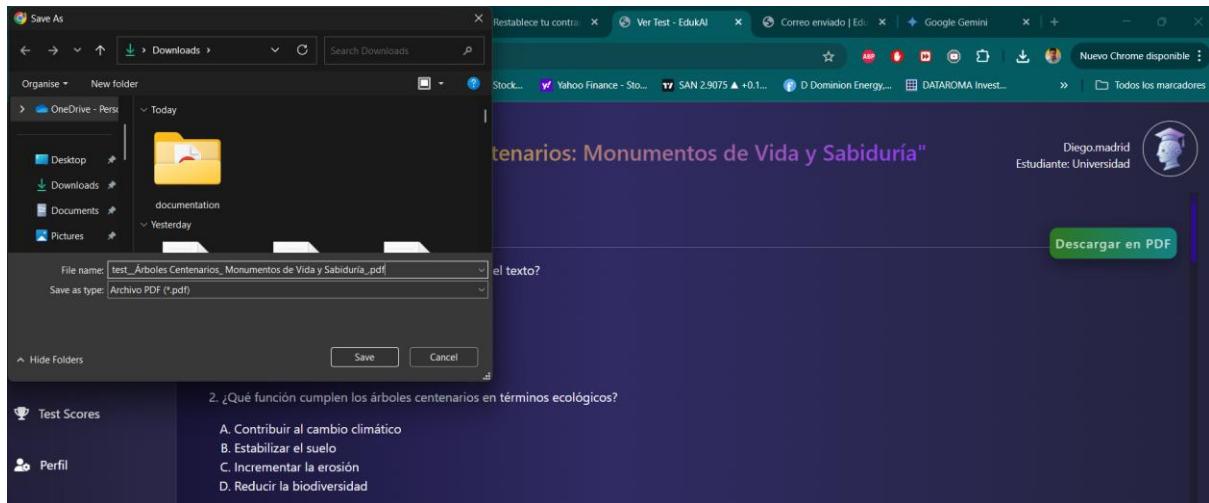
Una vez generadas las preguntas, se pueden ver en la sección **Preguntas Generadas**, que incluye dos enlaces: uno para ver las preguntas y otro para realizar un test interactivo.



La **interfaz para ver preguntas** muestra las preguntas seguidas de sus respuestas, además de un botón para **descargar en PDF**.



En el **proceso de descarga del test en PDF**, se selecciona la ubicación donde se guardará el archivo.



La **visualización del PDF** muestra las preguntas generadas con sus respuestas al final.

Test generado por EdukAI
"Árboles Centenarios: Monumentos de Vida y Sabiduría"

1. ¿Qué representan los árboles centenarios según el texto?
A) Elementos de la flora
B) Monumentos vivos
C) Guardianes de la memoria
D) Ecosistemas artificiales

2. ¿Qué función cumplen los árboles centenarios en términos ecológicos?
A) Contribuir al cambio climático
B) Estabilizar el suelo
C) Incrementar la erosión
D) Reducir la biodiversidad

3. ¿Cómo son considerados los árboles centenarios en muchas culturas antiguas?
A) Fuentes de contaminación
B) Morada de dioses
C) Símbolos de destrucción
D) Inútiles para la biodiversidad

Respuestas Correctas

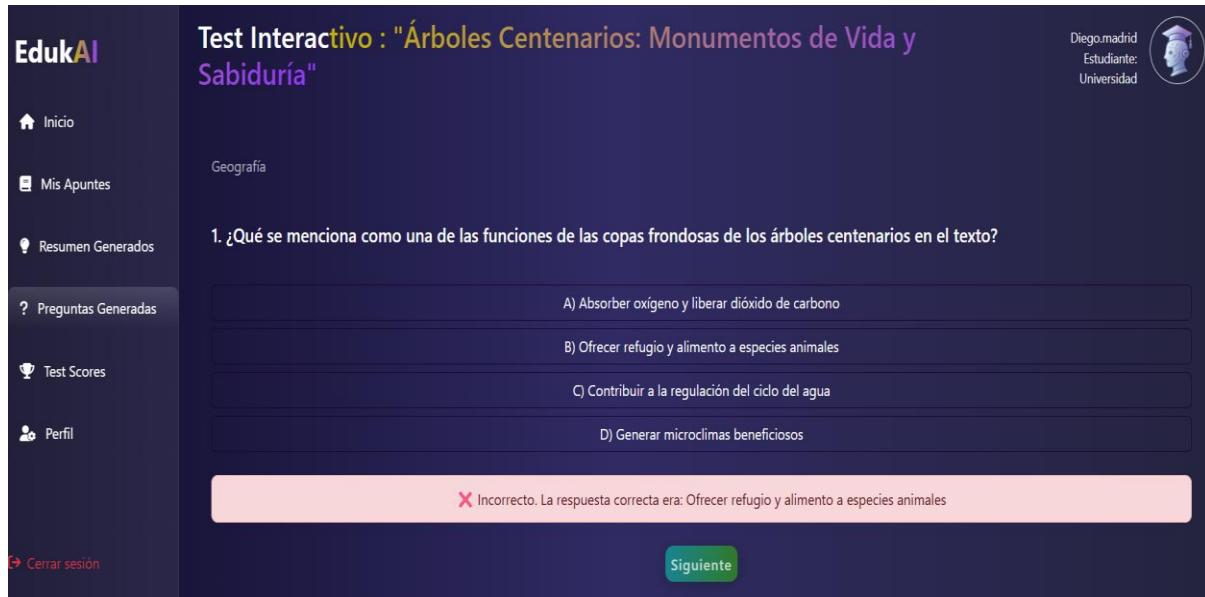
1. ¿En qué país nació Nikola Tesla?:
Croacia

2. ¿En qué año nació Nikola Tesla?:
1856

3. ¿Dónde comenzó Nikola Tesla su educación formal en ingeniería eléctrica?:
Graz

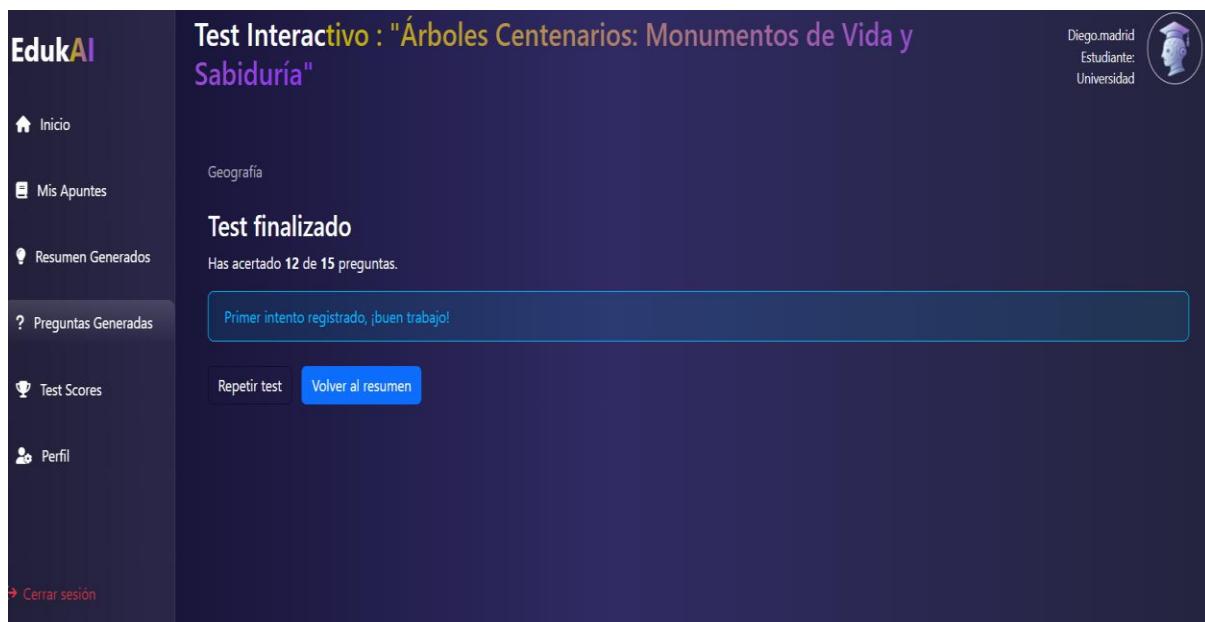
4. ¿Qué sistema de corriente eléctrica era defendido por Thomas Edison?:
Corriente Continua (CC)

Al ingresar en la sección “**Empezar test interactivo**”, se presenta una interfaz con las preguntas y opciones generadas, ordenadas aleatoriamente. Si el usuario se equivoca, se muestra la opción correcta.



The screenshot shows a dark-themed user interface for a test. On the left is a sidebar with navigation links: Inicio, Mis Apuntes, Resumen Generados, Preguntas Generadas (highlighted in purple), Test Scores, Perfil, and Cerrar sesión. The main area has a header "Test Interactivo : 'Árboles Centenarios: Monumentos de Vida y Sabiduría'" and a sub-header "Geografía". A question is displayed: "1. ¿Qué se menciona como una de las funciones de las copas frondosas de los árboles centenarios en el texto?". Four options are listed: A) Absorber oxígeno y liberar dióxido de carbono, B) Ofrecer refugio y alimento a especies animales, C) Contribuir a la regulación del ciclo del agua, and D) Generar microclimas beneficiosos. Below the options, a message says "Incorrecto. La respuesta correcta era: Ofrecer refugio y alimento a especies animales". At the bottom right is a green "Siguiente" button.

Una vez finalizado el test, se muestra el resultado y el avance. Si el resultado es mayor al registrado anteriormente, se actualiza. Se puede repetir el test o ver el resumen.



The screenshot shows the same interface after a test has been completed. The header remains the same. The main area displays the result: "Test finalizado" and "Has acertado 12 de 15 preguntas.". A message box says "Primer intento registrado, ¡buen trabajo!". At the bottom are two buttons: "Repetir test" and "Volver al resumen".

La sección **Historial de Resultados** muestra la puntuación del test interactivo, la fecha y la opción de repetir el test para mejorar el puntaje.

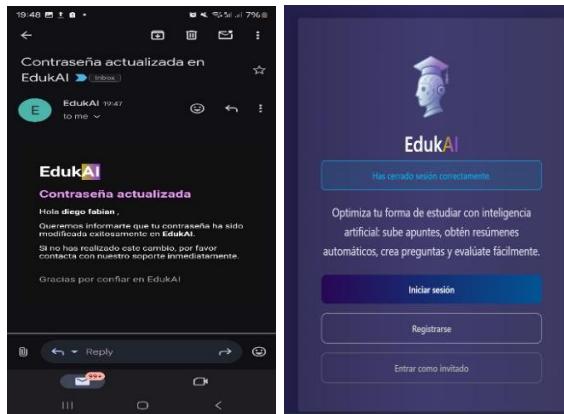
Título	Puntuación	Total Preguntas	Fecha	Test
"Árboles Centenarios: Monumentos de Vida y Sabiduría"	12 Excellent	15	26 de mayo de 2025 a las 17:43	Repetir Test

En **Perfil de Usuario** se pueden modificar los datos del usuario. Además, se puede agregar una imagen de perfil o elegir un avatar. (El email no se puede modificar; solo se muestra).

Es posible ver una **previsualización de los avatares o imágenes** antes de guardar los cambios.

Al guardar los cambios, estos se actualizan en el dashboard con un mensaje emergente. Si también se cambia la contraseña, se envía un correo de aviso al usuario.

Se envía un **email al usuario** cuando se cambia la contraseña. Al cerrar sesión, se muestra un mensaje de confirmación y se redirige a la pantalla de bienvenida.



En el modo invitado, no se muestran las secciones del menú lateral. Aparece un consejo al estudiante y un texto con un enlace para registrarse si así lo desea.



En **Subir Apuntes**, se usa la misma interfaz que para los usuarios registrados, pero los datos no se guardan en la base de datos, solo en caché.



Se puede leer el resumen generado, pero este desaparece al salir de la página. Existe un botón para generar otro resumen si se desea.

The screenshot shows a dark-themed web interface for EdukAI. At the top left is the logo 'EdukAI'. On the right, there's a circular profile icon with the text 'Invitad@' next to it. The main title 'Resumen: Nikola Tesla: Genio y Visionario de la Electricidad' is displayed. Below the title, a message says 'Estás viendo un resumen generado como invitado. Recuerda que este contenido no se guarda en tu perfil.' A text box contains a summary of Nikola Tesla's life and contributions. At the bottom left is a button labeled 'Generar otro'.

En **Generar Preguntas**, se usa la misma interfaz, aunque con una función diferente en el backend para crear el test.

This screenshot shows the 'Subir Apuntes' (Upload Notes) page. The interface is similar to the previous one, with the EdukAI logo and a user profile icon. The main heading is 'Subir Apuntes'. A text input field contains a summary of Nikola Tesla. Below the input field are two radio buttons: one selected for 'Switch si deseas Subir archivo PDF' and another for 'Switch para personalizar título y categoría de tus apuntes'. At the bottom is a blue button labeled 'Subir Archivo'.

Se permite visualizar los tests generados y descargar el archivo en PDF.

This screenshot shows a generated test titled 'Test del apunte: El Legado de Nikola Tesla: Genio e Innovación'. It includes two questions: 14. '¿Qué actividad excéntrica realizaba Tesla en Central Park durante sus últimos años?' with options A, B, C, D; and 15. '¿Qué reconocimiento recibió Nikola Tesla póstumamente por sus contribuciones a la radio?' with options A, B, C, D. Below the questions is a section titled 'Soluciones del test' containing two questions: 1. '¿En qué país nació Nikola Tesla?' (correct answer: Croacia) and 2. '¿En qué año nació Nikola Tesla?' (correct answer: 1856).

El test generado por la IA incluye las preguntas, opciones y respuestas al final del archivo PDF.

The screenshot shows a PDF document with a dark theme. At the top left, there are two small thumbnail images labeled '1' and '2'. The main content area has a white background and contains the following text:

Test generado por EdukAI
El Legado de Nikola Tesla: Genio e Innovación

1. ¿En qué país nació Nikola Tesla?
A) Croacia
B) Austria
C) Estados Unidos
D) Italia

2. ¿En qué año nació Nikola Tesla?
A) 1850
B) 1870
C) 1856
D) 1860

Respuestas Correctas

1. ¿En qué país nació Nikola Tesla?:
Croacia
2. ¿En qué año nació Nikola Tesla?:
1856
3. ¿Dónde comenzó Nikola Tesla su educación formal en ingeniería eléctrica?:
Graz
4. ¿Qué sistema de corriente eléctrica era defendido por Thomas Edison?:
Corriente Continua (CC)
5. ¿Qué inventor rivalizó con Tesla en la 'Guerra de las Corrientes'?:
Thomas Edison

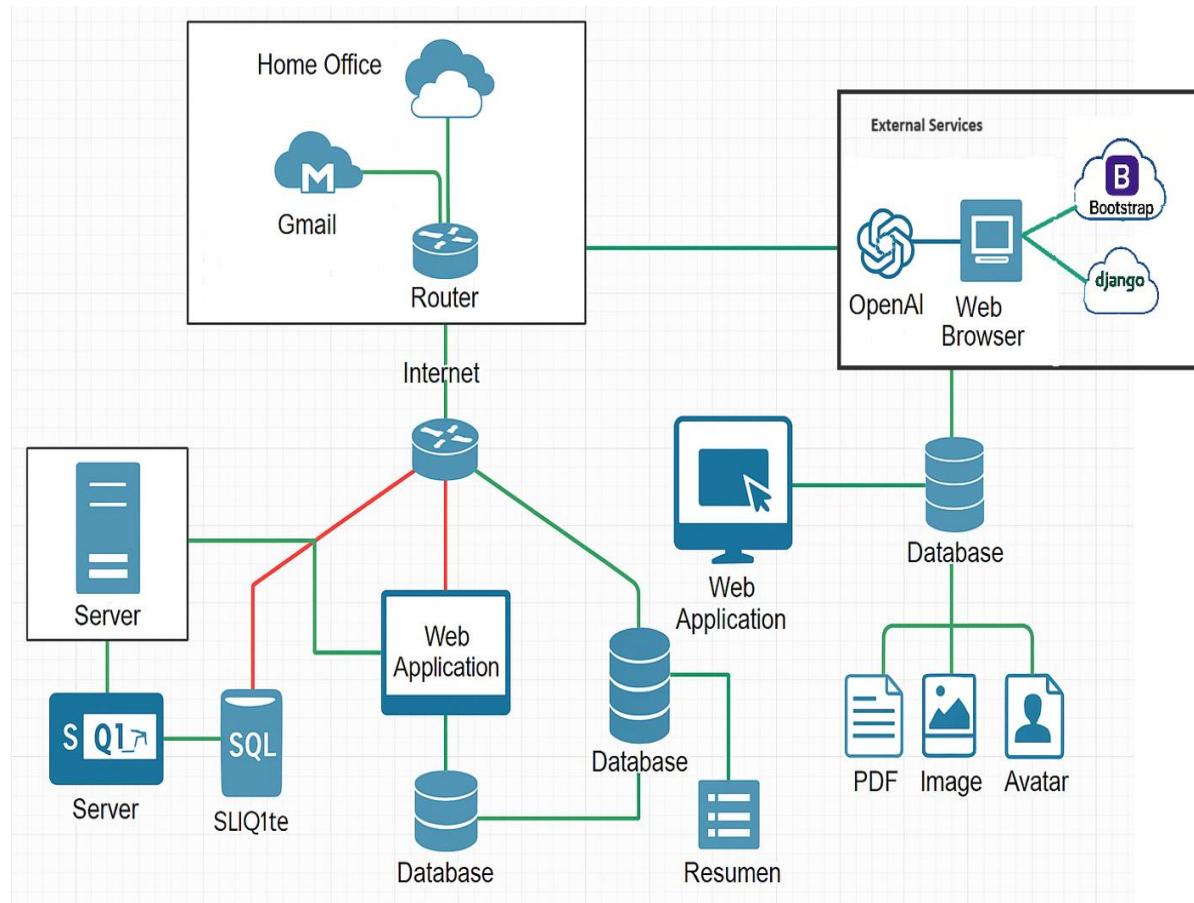
Nota final:

Las interfaces de la plataforma EdukAI han sido diseñadas pensando en la usabilidad, accesibilidad y experiencia del usuario. Cada sección ofrece funcionalidades intuitivas que permiten al estudiante interactuar de forma sencilla con sus apuntes, resúmenes y evaluaciones, tanto si accede como usuario registrado como en modo invitado.

Diagrama de Red – EdukAI

El siguiente diagrama de red representa la infraestructura técnica del proyecto EdukAI. En él se visualiza la conexión entre los distintos componentes del sistema, como el servidor web, la base de datos local en SQLite, el servicio de envío de correos vía Gmail, y los servicios externos utilizados, como OpenAI para la generación de contenidos con IA, Bootstrap para el diseño visual y Django como framework backend.

La red también muestra cómo los dispositivos cliente acceden a la aplicación a través de Internet y cómo se gestionan internamente archivos PDF, imágenes de perfil y avatares almacenados en la base de datos.



TECNOLOGÍA :

Tecnologías, librerías y herramientas empleadas en el desarrollo de la plataforma EdukAI. Se incluye una breve descripción de cada tecnología, así como su aplicación específica dentro del sistema.

Python

Descripción: Lenguaje principal del backend del proyecto.

Uso: Lógica de negocio, integración con IA, validaciones, envío de correos y manejo del servidor.

Django (v5.2)

Descripción: Framework web de alto nivel para Python.

Uso: Estructura principal de la app (rutas, modelos, autenticación, vistas, formularios, templates).

OpenAI (v1.76.0)

Descripción: Plataforma de inteligencia artificial.

Uso: Generación automática de resúmenes y preguntas a partir del contenido de los apuntes.

PyPDF2 (v3.0.1)

Descripción: Librería de Python para manipular archivos PDF.

Uso: Lectura y extracción de texto desde archivos PDF subidos por los usuarios.

HTML, CSS y JavaScript

Descripción: Tecnologías estándar del frontend para estructurar contenido (HTML), dar estilo (CSS) y añadir interactividad (JavaScript).

Uso en el proyecto: Componen la interfaz web visible para el usuario junto a Bootstrap. Se utilizan en conjunto con las plantillas de Django.

Python-decouple (v3.8)

Descripción: Librería para gestionar variables de entorno de forma segura.

Uso: Protección de credenciales como la clave API de OpenAI o la configuración SMTP de Gmail.

Pydantic (v2.11.3)

Descripción: Librería de validación de datos basada en tipos de Python.

Uso: Validación y estructuración de datos en comunicación con APIs o formularios.

Bootstrap (v5)

Descripción: Framework CSS para diseño web responsive.

Uso: Construcción de una interfaz moderna, adaptable y estilizada.

Font Awesome

Descripción: Biblioteca de íconos vectoriales para la web.

Uso: Mejora de la interfaz visual mediante íconos en botones, menús y formularios.

Gmail (SMTP)

Descripción: Servicio de correo electrónico de Google.

Uso: Envío de correos para activación de cuentas, recuperación de contraseña y notificaciones.

SQLite

Descripción: Base de datos relacional ligera.

Uso: Almacenamiento local de usuarios, apuntes, preguntas, resultados, archivos y perfiles.

Django-mathfilters

Descripción: Paquete de filtros personalizados para operaciones matemáticas dentro de los templates de Django.

Uso: Permite realizar cálculos como sumas, restas, porcentajes o divisiones directamente en plantillas HTML (por ejemplo, mostrar porcentajes de aciertos en los tests).

Django-widget-tweaks

Descripción: Herramienta que facilita la personalización de los formularios de Django directamente desde el template.

Uso: Utilizado para aplicar clases CSS, atributos o estilos directamente sobre los widgets de formularios (forms.py) en los archivos HTML, sin necesidad de modificar el backend.

ReportLab

Descripción: Librería de Python para generar documentos PDF de forma dinámica.

Uso: Utilizado para exportar tests generados por IA en formato PDF, incluyendo preguntas y respuestas, con estilos personalizados y estructura visual.

Django ORM / Aggregates

Descripción: Sistema de consultas de Django sobre modelos, que permite operaciones como filtros, agregados y ordenamientos.

Uso: Max() se usa para obtener la última fecha de pregunta por apunte; get_object_or_404, annotate, filter y otros son parte del ORM para acceder a la base de datos.

EmailMessage / EmailMultiAlternatives

Descripción: Clases de Django para el envío de correos electrónicos HTML y alternativos.

Uso: Envío de correos de activación, recuperación de contraseña y confirmaciones mediante plantillas HTML.

Token y seguridad

- urlsafe_base64_encode, force_bytes, default_token_generator:

Uso: Envío y validación de tokens de activación y recuperación de contraseña.

Manejo de formularios personalizados

- CustomUserCreationForm, PerfilForm, CustomEmailLoginForm:

Uso: Formularios extendidos para registro, login con correo, edición de perfil y selección de avatar.

Decoradores y seguridad

- @never_cache, @require_POST, @login_required:

Uso: Control del acceso a vistas protegidas, seguridad de formularios y control de caché para evitar errores de validación de tokens.

METODOLOGÍA

Metodología aplicada: *Kanban con planificación incremental*

Durante el desarrollo del proyecto se ha aplicado una metodología ágil basada en **Kanban**, acompañada de una planificación incremental por fases. Esto ha facilitado la gestión visual del flujo de trabajo mediante tableros, permitiendo priorizar tareas según su urgencia e importancia, y adaptarse a los cambios de manera flexible.

Cada funcionalidad del sistema —desde la subida de apuntes y su procesamiento mediante IA, hasta la generación de tests y la gestión del perfil de usuario— fue abordada como una **unidad de trabajo individual**. Esta forma de trabajo también ayudó a detectar cuellos de botella, como tareas bloqueadas por dependencias técnicas o rediseños visuales.

La elección de esta metodología se justifica por las características del proyecto:

- **Multidisciplinariedad:** combina desarrollo backend (API, base de datos), frontend (interfaz y usabilidad), integración de modelos de IA, seguridad, y pruebas.
- **Plazo limitado:** el calendario del RFTP exige flexibilidad y capacidad de adaptación sin perder el control del avance.
- **Gestión individual del tiempo:** permite adaptar las tareas al ritmo real de desarrollo sin depender de una planificación rígida.

Planificación Inicial vs Final

A continuación se presenta la comparación entre la planificación inicial del proyecto y las horas reales dedicadas a cada tarea, con base en la experiencia durante el desarrollo.

Tarea	Horas planificadas	Horas reales
Ideación del proyecto	8	12
Mockups de login/dashboard	8	9
Definición de estilo visual	6	8
Creación de logo y nombre	6	10
Generación de avatares	8	8
Estructura base del proyecto Django	8	10
Sistema de login y registro	6	8
Sistema de recuperación y activación de	8	12

cuenta		
Dashboard funcional con menú lateral	6	5
Subida de apuntes y extracción de texto	6	6
Generación de resumen con OpenAI	7	11
Generación automática de título y categoría	6	6
Visualización de resúmenes	7	7
Generación de preguntas tipo test	7	9
Vista para realizar el test y guardar puntuación	5	6
Historial de tests y mejoras	6	6
Descarga de test en PDF	6	9
Exportación de apuntes en texto	6	8
Preparacion de Documentacion	12	11
Configuracion Funcionalidades Usuarios no registrados	3	4
Creacion de animacion de espera en JavaScript	-	-
Comprobacion flujo de mensajes emergentes de generacion de resumenes	1	4
Agregar input de comprobar password en perfil de usuario	-	-
Agregar boton para vizualizar contrasena en password reset complete html	2	3
Crear link para terminos y condiciones de privacidad	1	3
Anadir funcionalidad en el view para la busqueda de apuntes por categorias	-	-
Agregar boton para ver test generados en preguntas.html	2	3
Reparar problema del layout de los pdf generados	2	1
Nuevo template para ver los test generados sin tener que descargarlo en PDF	2	5
Nuevos titulos dinamicos por seccion y menu lateral fijo con scroll personalizado	3	5
Creacion diagra de flujos y casos de uso del proyecto	12	13
Modificar eliminacion de apuntes tambien preguntas generadas e historial	1	1
Agregar texto generado por la IA con consejos al estudiante	1	1
Total	162	212

Total de horas planificadas: 162 h

Total de horas reales estimadas: 212 h

Análisis de desviaciones

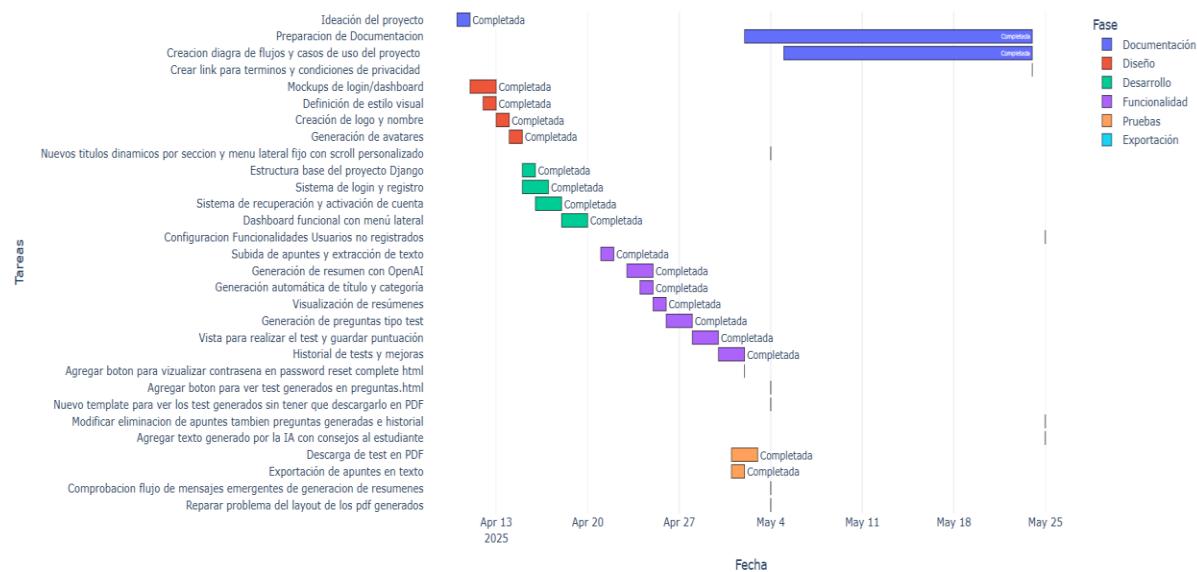
Durante el desarrollo del proyecto EdukAI se produjeron diversas desviaciones entre las horas inicialmente planificadas y las horas realmente dedicadas a cada tarea. A continuación, se detallan las principales causas observadas:

- **Incremento en tareas visuales y de frontend:** La fase de diseño de interfaz (mockups, estilos visuales y creación de componentes visuales) requirió más tiempo del previsto debido a ajustes de diseño, pruebas de usabilidad y mejoras iterativas sugeridas durante el desarrollo.
- **Integración con IA y procesamiento de apuntes:** La integración de modelos para el procesamiento de texto, generación de resúmenes y preguntas automáticas exigió más dedicación de la esperada. Se destinaron horas adicionales a pruebas, afinación de los resultados y manejo de archivos PDF.
- **Desviaciones en tareas pequeñas acumuladas:** Algunas tareas de menor duración (como ajustes visuales, validaciones de formularios, o integración de funciones secundarias) acumularon más tiempo por el número de revisiones necesarias y su dependencia con otras funcionalidades.
- **Reducción en algunas tareas técnicas:** En contrapartida, algunas tareas backend (como la estructura base de Django y configuraciones iniciales) fueron completadas en menos tiempo, gracias a plantillas reutilizables y experiencia previa con el framework.

En conjunto, el proyecto tuvo una desviación aproximada de **+52 horas** sobre la planificación inicial, representando un incremento de alrededor del **16.5%**. Esta diferencia es razonable en el contexto de una metodología ágil.

Diagrama de Gantt

Diagrama de Gantt - Proyecto EdukAI



EdukAI Proyecto DAM		Tareas	Inicio	Duración (horas)	Días estimados	Fin	Fase	Estado
Ideación del proyecto			10/04/2025	8	1	11/04/2025	Documentación	Completada
Mockups de login/dashboard			11/04/2025	8	2	13/04/2025	Diseño	Completada
Definición de estilo visual			12/04/2025	6	1	13/04/2025	Diseño	Completada
Creación de logo y nombre			13/04/2025	6	1	14/04/2025	Diseño	Completada
Generación de avatares			14/04/2025	8	1	15/04/2025	Diseño	Completada
Estructura base del proyecto Django			15/04/2025	8	1	16/04/2025	Desarrollo	Completada
Sistema de login y registro			15/04/2025	6	2	17/04/2025	Desarrollo	Completada
Sistema de recuperación y activación de cuenta			16/04/2025	8	2	18/04/2025	Desarrollo	Completada
Dashboard funcional con menu lateral			18/04/2025	6	2	20/04/2025	Desarrollo	Completada
Subida de apuntes y extracción de texto			21/04/2025	6	1	22/04/2025	Funcionalidad	Completada
Generación de resumen con OpenAI			23/04/2025	7	2	25/04/2025	Funcionalidad	Completada
Generación automática de título y categoría			24/04/2025	6	1	25/04/2025	Funcionalidad	Completada
Visualización de resúmenes			25/04/2025	7	1	26/04/2025	Funcionalidad	Completada
Generación de preguntas tipo test			26/04/2025	7	2	28/04/2025	Funcionalidad	Completada
Vista para realizar el test y guardar puntuación			28/04/2025	5	2	30/04/2025	Funcionalidad	Completada
Historial de tests y mejoras			30/04/2025	6	2	2/05/2025	Funcionalidad	Completada
Descarga de test en PDF			1/05/2025	6	2	3/05/2025	Pruebas	Completada
Exportación de apuntes en texto			1/05/2025	6	1	2/05/2025	Pruebas	Completada
Preparacion de Documentacion			2/05/2025	12	10	24/05/2025	Documentación	Completada
Configuracion Funcionalidades Usuarios no registrados			25/05/2025	3	1	25/05/2025	Desarrollo	Completada
Creacion de animacion de espera en JavaScript			3/05/2025				Funcionalidad	Bloqueada
Comprobacion flujo de mensajes emergentes de generacion de resúmenes			4/05/2025	1	1	4/05/2025	Pruebas	Completada
Agregar input de comprobar password en perfil de usuario			3/05/2025				Funcionalidad	Bloqueada
Agregar boton para vizualizar contrasena en password reset complete html			2/05/2025	2	1	2/05/2025	Funcionalidad	Completada
Crear link para terminos y condiciones de privacidad			24/05/2025	1	1	24/05/2025	Documentación	Completada
Anadir funcionalidad en el view para la busqueda de apuntes por categorias							Funcionalidad	Bloqueada
Agregar boton para ver test generados en preguntas.html			4/05/2025	2	1	4/05/2025	Funcionalidad	Completada
Reparar problema del layout de los pdf generados			4/05/2025	2	1	4/05/2025	Exportación	Completada
Nuevo template para ver los test generados sin tener que descargarlo en PDF			4/05/2025	2	1	4/05/2025	Funcionalidad	Completada
Nuevos titulos dinamicos por seccion y menu lateral fijo con scroll personalizado			4/05/2025	3	1	4/05/2025	Diseño	Completada
Creacion diagra de flujos y casos de uso del proyecto			5/05/2025	12	4	24/05/2025	Documentación	Completada
Modificar eliminacion de apuntes tambien preguntas generadas e historial			25/05/2025	1	1	25/05/2025	Funcionalidad	Completada
Agregar texto generado por la IA con consejos al estudiante			25/05/2025	1	1	25/05/2025	Funcionalidad	Completada
Total					162	51	Documentación	Completada

Totalización Final de Horas por Categoría

En esta sección se presenta un resumen total del tiempo dedicado a cada tipo de tarea durante el desarrollo del proyecto EdukAI. Esta distribución agrupa las actividades por categoría funcional, permitiendo evaluar la carga de trabajo de cada área y la proporción de esfuerzo invertido en cada una.

Tipo de tarea	Total de horas
Documentación	37 h
Backend (Django/API)	47 h
Frontend (HTML/CSS/JS)	35 h
IA y Procesamiento de PDF	18 h
Pruebas y mejora	25 h

Total general: 162 h

Presupuesto del Proyecto

A continuación, se detalla el presupuesto estimado para el desarrollo del proyecto EdukAI. Se incluyen los costes derivados del tiempo invertido, herramientas utilizadas, y recursos externos empleados durante el desarrollo.

1. Coste por horas de trabajo

El desarrollo ha sido realizado de forma individual por el alumno, quien ha invertido un total de **162 horas** en las diferentes fases del proyecto. Considerando un coste medio de **25 €/hora** para un desarrollador junior/freelance en el mercado español:

Concepto	Horas	Coste por hora	Subtotal
Desarrollo Backend y Frontend	77	25 €	1.925 €
Procesamiento de IA y funcionalidad PDF	38	25 €	950 €
Documentación y pruebas	22	25 €	550 €
Diseño UI/UX e integración	25	25 €	625 €

Concepto	Horas	Coste por hora	Subtotal
Total mano de obra	162		4.050 €

2. Coste en herramientas y APIs

- **Tokens OpenAI (IA y NLP):** ~5 € (uso de la API para generación de resúmenes y preguntas).
- **Herramientas de desarrollo:** GitHub Desktop, PyCharm (versión gratuita), Django (open source), Plotly (uso gratuito), etc. — **0 €**
- **Otras bibliotecas utilizadas (spaCy, transformers, etc.):** Gratuitas (open source)

3. Coste estimado de infraestructura

Durante la fase de desarrollo no se ha desplegado el proyecto en producción, pero se estima un coste aproximado mensual si se publicara en servicios cloud como AWS:

Recurso	Coste mensual estimado
Hosting básico (EC2 o similar)	15 – 20 €
Base de datos (RDS pequeña)	10 – 15 €
Almacenamiento S3 (PDFs)	5 €
Total mensual estimado	30 – 40 €

4. Coste estimado de una app web similar en el mercado

Si el desarrollo de esta plataforma fuera encargado a una empresa de software, se estima un coste total de entre **6.000 € y 12.000 €**, en función de:

- Complejidad técnica (procesamiento de texto, test interactivos)
- Funcionalidades IA integradas
- Panel de usuario personalizado, exportación a PDF y organización de contenido

Resumen de presupuesto

Categoría	Coste estimado
Mano de obra (162 h @ 25 €/h)	4.050 €
Tokens OpenAI	5 €
Coste mensual de infraestructura	30 – 40 € (estimado)
Total estimado	4.085 € + infraestructura

README y Control de Versiones

A continuación se presentan capturas del repositorio del proyecto alojado en GitHub, que reflejan tanto la documentación general (README.md) como el historial de control de versiones mediante Git, también se muestran fragmentos del historial de commits, que evidencian el desarrollo progresivo del sistema, la gestión de ramas, y el trabajo estructurado en fases.



• Bootstrap

☞ **Otros:**

- PyPDF2 – extracción de texto
- ReportLab – generación de PDF
- AWS S3 – almacenamiento de archivos (opcional)
- GitHub – control de versiones

☞ **Instalación local**

1. Clona el repositorio:

```
gh repo clone diegofabian89/plataforma-estudios-ia
cd edukai
```
2. Crea un entorno virtual e instala dependencias:

```
python -m venv venv
source venv/bin/activate # o venv\Scripts\activate en Windows
pip install -r requirements.txt
```
3. Aplica migraciones y ejecuta el servidor:

```
python manage.py migrate
python manage.py runserver
```
4. Accede desde: <http://127.0.0.1:8000/>

☞ **Estructura del proyecto**

```
edukai/
├── backend/          # Lógica Django y APIs FastAPI
├── frontend/         # Plantillas HTML, CSS, JS
├── media/            # Archivos subidos por el usuario
├── static/           # Archivos estáticos
├── templates/        # Interfaces visuales
├── tests/            # Pruebas unitarias
└── requirements.txt
└── README.md
```

☞ **Planificación y Gantt**

Consulta el progreso del proyecto y su planificación en el [Diagrama de Gantt](#).

☞ **Casos de uso y testing**

- Test de carga de documentos
- Test de precisión en resúmenes generados
- Validación de preguntas creadas
- Pruebas de sesión, permisos y funcionalidades para invitados

☞ **Presupuesto**

- Horas estimadas: 162 h
- Coste estimado de desarrollo: 4.085 €
- Inversión en OpenAI: 5 €
- Infraestructura: uso local y/o AWS (~30 €/mes estimado)

☞ **Licencia**

Proyecto académico realizado como parte del Ciclo Formativo de Grado Superior en **Desarrollo de Aplicaciones Multiplataforma (DAM)**.
Uso educativo – Todos los derechos reservados © 2025.

☞ **Contacto**

DIEGO FABIAN
Email: diegofabian1989@hotmail.es
GitHub: [@diegofabian89](https://github.com/diegofabian89)

plataforma-estudios-ia · Private

main · 2 Branches · 0 Tags

Go to file · Add file · Code

About

Repository para el proyecto final de DAM

- Readme
- Activity
- 0 stars
- 1 watching
- 0 forks

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Languages

HTML 45.8% · Python 37.4% · CSS 15.6% · JavaScript 1.2%

Commit	Author	Date
feat: Implemented main functionalities for registered users...	diegofabian89	3 weeks ago
merge: fusionar rama dev en main y eliminar archivos compi...	diegofabian89	11 hours ago
feat: enable guest mode features and optimize UI for small s...	diegofabian89	14 hours ago
Add url,template and view of Terms and conditions, into regis...	diegofabian89	yesterday
feat: enable guest mode features and optimize UI for small s...	diegofabian89	14 hours ago
requiremmts file updated	diegofabian89	11 hours ago
chore: ignore file pyc y _pycache_	diegofabian89	11 hours ago
Propuesta de Proyecto.pdf	diegofabian89	Last month
First commit:Initial structure of Django project	diegofabian89	Last month
readme text modified	diegofabian89	9 minutes ago
First commit:Initial structure of Django project	diegofabian89	Last month
requiremmts file updated	diegofabian89	11 hours ago
README	diegofabian89	9 minutes ago

Git · Log: dev · History: README.md · + · -

HEAD (Current Branch)

Local

- main
- dev

Remote

- origin
- main
- dev

Q · Text or hash · * Cc · Branch: dev · User · Date · Paths · J · Select commit to view changes

Commit	Author	Date
origin & dev requirements file updated	diegofabian89	Today 02:14
feat: Readme added to the project	diegofabian89	Today 02:11
refactor: remove premature toast notification on download click	diegofabian89	Yesterday 23:47
feat: enable guest mode features and optimize UI for small screen	diegofabian89	Yesterday 23:11
feat: display daily AI-generated personalized advice per day	diegofabian89	Yesterday 17:29
Add url,template and view of Terms and conditions, into register	diegofabian89	Yesterday 15:58
Improve layout UX: fixed sidebar + scrollable content + dynamic	diegofabian89	04/05/2025 19:29
Improve UX and PDF formatting: redirect success messages to	diegofabian89	04/05/2025 16:25
UI improvements and guest access: added show-password toggle	diegofabian89	03/05/2025 00:06
feat: Implemented main functionalities for registered users: note	diegofabian89	02/05/2025 19:06

Contacto

Diego.Fabian

Email: diegofabian1989@hotmail.es

GitHub: <https://github.com/diegofabian89/plataforma-estudios-ia>

TRABAJOS FUTUROS

A lo largo del desarrollo de EdukAI han surgido varias ideas que podrían implementarse en el futuro para seguir mejorando la plataforma. Estas propuestas no solo buscan hacer la experiencia del usuario más completa y personalizada, sino también fomentar la colaboración entre estudiantes y preparar el sistema para crecer y adaptarse a nuevas necesidades.

Personalización del aprendizaje

Una de las funcionalidades clave que se plantea incorporar es la **posibilidad de personalizar la generación de preguntas**. El usuario podrá elegir la **cantidad de preguntas** que desea generar y definir su **nivel de dificultad** (fácil, medio o difícil), adaptando así el contenido a sus necesidades o a su nivel de conocimiento.

Comunidad y colaboración

Se prevé la creación de una **comunidad de usuarios** donde sea posible **compartir resúmenes personalizados y bancos de preguntas generadas**. Además, se podrán incorporar funcionalidades de **ranking y competiciones**, permitiendo a los usuarios comparar sus puntuaciones sobre temas específicos y fomentar un aprendizaje gamificado. Otras opciones incluyen un **sistema de comentarios, valoraciones o revisiones** para enriquecer colectivamente el contenido compartido.

Expansión multiplataforma

A futuro, se plantea el **despliegue completo de la aplicación web** en un servidor en la nube (como Heroku, Render o AWS), para que pueda accederse desde cualquier dispositivo con conexión a Internet.

También se considera desarrollar una **versión móvil de EdukAI**, ya sea como app nativa (Android/iOS) o híbrida, permitiendo un acceso más cómodo y portátil.

Mejora de la inteligencia artificial

Se pueden incorporar mejoras en la lógica de la IA, como un sistema de **adaptación automática al perfil del usuario**, en función de su nivel educativo, historial de interacción o

rendimiento previo.

Otra mejora posible es la **detección automática del idioma** del contenido subido, para soportar múltiples lenguas y ampliar así el alcance de la plataforma.

Seguridad y escalabilidad

Para mejorar la robustez y escalabilidad del sistema, sería recomendable **migrar la base de datos de SQLite a PostgreSQL o MySQL**, lo cual permitiría un manejo más eficiente de múltiples usuarios concurrentes y mayor volumen de datos.

También se plantea la implementación de un **sistema de roles**, como profesor, estudiante o administrador, para gestionar permisos y funcionalidades específicas por tipo de usuario.

CONCLUSIONES

El desarrollo de EdukAI ha supuesto un reto técnico y creativo que ha permitido aplicar de forma práctica los conocimientos adquiridos durante el ciclo formativo de Desarrollo de Aplicaciones Multiplataforma. A lo largo del proyecto, se ha diseñado una plataforma funcional, intuitiva y centrada en el usuario, que combina el potencial de la inteligencia artificial con herramientas modernas de desarrollo web.

El uso de tecnologías como Django, OpenAI y Bootstrap ha facilitado la creación de un sistema escalable, seguro y visualmente atractivo. Además, se ha conseguido integrar funcionalidades clave como la generación automática de contenido, el almacenamiento personalizado, la autenticación de usuarios y la interacción dinámica mediante formularios y tests.

EdukAI no solo resuelve un problema concreto —facilitar el estudio y la autoevaluación—, sino que abre las puertas a nuevas formas de aprendizaje personalizadas y colaborativas. El proyecto queda en una base sólida para seguir creciendo, con múltiples posibilidades de mejora futura tanto a nivel técnico como funcional.

En definitiva, EdukAI representa una experiencia completa de desarrollo de software, desde la idea inicial hasta su implementación final, demostrando la capacidad de transformar una necesidad educativa en una solución tecnológica real y aplicable.

REFERENCIAS

En este apartado se recogen las fuentes de información, herramientas y recursos que han sido consultados o utilizados durante el desarrollo del proyecto EdukAI.

Aplicado en el aprendizaje del framework Django, su estructura y desarrollo de vistas, modelos y autenticación.

Cancino, A. (2022). *CURSO de DJANGO desde CERO | (COMPLETO)*. Recuperado de https://www.youtube.com/watch?v=CFc_zCKjSNU

Aplicado en la integración de tokens de activación, recuperación de contraseña y envío de correos electrónicos desde Django.

Wellat (2024). *Implementación de seguridad y recuperación de cuentas en aplicaciones web internas*. Centro de prácticas personales.

Aplicado para el diseño visual inicial de la aplicación y la creación de mockups de navegación y estructura.

Confianza23 (2024). *Diseño de interfaces y experiencia de usuario para apps educativas*. Laboratorio de seguridad – centro de prácticas personales.

Aplicado en la generación de resúmenes automáticos, redacción de código y resolución de dudas técnicas durante el desarrollo del proyecto.

OpenAI. (2024). *ChatGPT - Asistente de desarrollo y documentación*. Recuperado de <https://chat.openai.com/>

Aplicado como soporte en la documentación oficial del framework y su configuración en el proyecto.

Django Software Foundation. (s. f.). *Django Documentation*. Recuperado de
<https://docs.djangoproject.com/en/5.2/>

Aplicado en el uso de Bootstrap para el diseño responsive de la interfaz web.

Bootstrap. (s. f.). *Bootstrap v5.3 Documentation*. Recuperado de
<https://getbootstrap.com/docs/5.3/getting-started/introduction/>

Aplicado para la creación de archivos PDF exportables dentro de la aplicación.

ReportLab. (s. f.). *The ReportLab Toolkit Documentation*. Recuperado de
<https://www.reportlab.com/docs/reportlab-userguide.pdf>