Desarrollo de una herramienta educativa basada en GPT para fomentar la exploración del universo entre los niños del páramo de Santurbán

Jean Pablo Ruiz Torres

Geovani Andrés Arenas Cuevas

Trabajo de Grado para Optar al Título de Ingeniero Electrónico

Director

Homero Ortega Boada

Doctor en Ciencias de la Ingeniería, Radiocomunicaciones

Codirector

Julián Gustavo Rodríguez Ferreira

Doctor en Astrofísica

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones

Ingeniería Electrónica

Bucaramanga

2025

Dedicatoria

Dedicado a mis padres, por darme la oportunidad de formarme profesionalmente y por acompañarme con su apoyo y esfuerzo en cada etapa de este camino.

Jean Pablo Ruiz Torres

Agradecimientos

A mis padres, por su apoyo incondicional, esfuerzo y confianza en cada etapa de mi vida.

A mi novia, por motivarme e impulsarme a dar lo mejor de mí en los momentos más exigentes de este camino.

A mis amigos, por ser parte de este proceso y hacerlo más ameno.

Al director y codirector de este proyecto, por su orientación y compromiso para el desarrollo de esta investigación.

A la Universidad Industrial de Santander y a mi Escuela E3T, por brindarme el espacio, el conocimiento y las herramientas necesarias para mi formación académica y profesional.

Jean Pablo Ruiz Torres

Tabla de Contenido

Pág.

Lista de Figuras

Pág.

Lista de Apéndices

Ver apéndices adjuntos

Apéndice A. Diagramas de flujo

Apéndice B. Pruebas de validación BOTE3T

Apéndice C. Video demostrativo

Apéndice D. Documento para el uso de BOTE3T

Apéndice E. Costos de Funcionamiento

Apéndice F. Consideraciones de la solución

Resumen

Título: Desarrollo de una herramienta educativa basada en GPT para fomentar la exploración del universo entre los niños del páramo de Santurbán[*]

Autor: Jean Pablo Ruiz Torres, Geovani Andrés Arenas Cuevas [**]

Palabras Clave: Inteligencia Artificial, Procesamiento de Lenguaje Natural, Radioastronomía, ChatBot, Herramienta Educativa, GPT API.

Descripción: Este proyecto busca promover el aprendizaje de la radioastronomía entre los niños del Páramo de Santurbán, al tiempo que fortalece los procesos de socialización del conocimiento vinculados al proyecto de instalación de una estación de radioastronomía en el Páramo de Berlín, para lograrlo, se diseñó una herramienta educativa basada en inteligencia artificial que permite responder preguntas de forma sencilla y personalizada, esta herramienta tomó como base un chatbot ya existente en la escuela E3T, adaptándolo para brindar información sobre el universo de manera amigable, con una interfaz pensada para niños y con interacción por voz. El sistema fue diseñado para aprender a partir de documentos que se le suministran, actualizando automáticamente sus contenidos, además de una interfaz amigable para los niños; durante el desarrollo, se adaptó la herramienta para que pueda entregar respuestas comprensibles y amigables, ajustadas a la edad de cada niño, adecuadas al contexto y con dos modos de uso, modo preguntas y modo interactivo. Esta iniciativa no solo apoya un proyecto científico, sino que también promueve el interés de los niños por la ciencia y tecnología; el proyecto demuestra cómo la inteligencia artificial puede ser usada como una herramienta educativa

---

[*] Trabajo de Grado

[**] Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones. Ingeniería Electrónica. Director: Homero Ortega Boada. Doctor en ciencias de la ingeniería, radiocomunicaciones. Codirector: Julián Gustavo Rodríguez Ferreira. Doctor en Astrofísica

Abstract

Title: Development of an educational tool based on GPT to encourage the exploration of the universe among the children of the Santurbán páramo[*]

Author(s): Jean Pablo Ruiz Torres, Geovani Andrés Arenas Cuevas [**]

Key Words: Artificial Intelligence, Natural Language Processing, Radio Astronomy, ChatBot, Educational Tool, GPT API

Description: This project aims to promote the learning of radio astronomy among children from the Páramo de Santurbán, while strengthening the processes of knowledge socialization linked to the project of installing a radio astronomy station in the Páramo de Berlín. To achieve this, an educational tool based on artificial intelligence was designed to answer questions in a simple and personalized manner. This tool was built on an existing chatbot from the E3T School, adapting it to provide information about the universe in a friendly way, with an interface designed for children and voice interaction. The system was designed to learn from documents provided to it, automatically updating its content. During the development, the tool was adapted to deliver understandable and friendly answers, adjusted to each child's age, appropriate to the context, and with two usage modes: question mode and interactive mode. This initiative not only supports a scientific project, but also promotes children's interest in science and technology; the project demonstrates how artificial intelligence can be used as an educational tool

[*] Degree Work

[**] Faculty of Physical and Mechanical Engineering. School of Electrical, Electronic, and Telecommunications Engineering. Director: Homero Ortega Boada. PhD in Engineering Sciences, Radio Communications. Co-director: Julián Gustavo Rodríguez Ferreira. PhD in Astrophysics.

Introducción

La inteligencia artificial (IA) y la radioastronomía constituyen dos campos de conocimiento que han adquirido un papel protagónico en el avance científico y educativo. Mientras la primera permite el desarrollo de sistemas capaces de comprender y generar lenguaje humano, la segunda posibilita el estudio del universo a través de ondas de radio, despertando el interés por la ciencia en nuevos públicos. Este proyecto se enmarca en el propósito de acercar el estudio de radioastronomía a los niños del Páramo de Santurbán, en el contexto del proyecto de instalación de una estación de radioastronomía promovido por la Universidad Industrial de Santander en el Páramo de Berlín.

La iniciativa se fundamentó en una experiencia previa con GPTE3T, un chatbot desarrollado por estudiantes de la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T) como parte del proyecto de grado “Diseño e implementación de un bot de charla basado en GPT para la acreditación internacional de los programas de pregrado de la E3T”. (Rueda, M., & Hernández, C., 2024). Este chatbot, orientado inicialmente a brindar información sobre el proceso de acreditación ABET, demostró ser eficaz como canal de consulta institucional.

A partir de esta base tecnológica, se planteó la creación de una versión mejorada de GPTE3T, capaz de actualizar automáticamente su conocimiento a partir de documentos que se le suministren sobre radioastronomía, incorporar una interfaz intuitiva adaptada a niños, la posibilidad de implementar una versión hablada y facilitar el uso por parte de educadores mediante una guía de uso de la herramienta. Asimismo, se realizaron pruebas piloto con usuarios reales, lo

que permitió realizar ajustes en su diseño, funcionalidad e interacción, con el fin de garantizar una experiencia educativa sencilla, accesible y efectiva.

Como resultado, se desarrolló una herramienta adaptada para enseñar contenidos relacionados con la radioastronomía a niños, con una interfaz interactiva, amigable e intuitiva, con dos modos principales de interacción; El primero, de tipo pregunta/respuesta, que permitió al usuario formular dudas y recibir respuestas claras, breves y adaptadas a su edad, el segundo, denominado modo de aprendizaje interactivo, ofreció explicaciones interactivas acompañadas de preguntas formativas, minijuegos y retroalimentación positiva, adicionalmente, se creó un sistema de gestión de sesiones capaz de registrar la interacción por usuario, resumir el proceso de aprendizaje al finalizar y retomar el contexto en futuras visitas, este mecanismo representa un valor agregado ya que el sistema genera resúmenes automáticos al final de cada sesión, lo que no solo permite mantener una trazabilidad del proceso de aprendizaje, sino que también aporta insumos útiles para su análisis pedagógico y la integraciones en entornos escolares llevando un monitoreo docente.

Este proyecto potencia las capacidades de la E3T y la UIS para desarrollar soluciones basadas en inteligencia artificial al incluir nuevos avances a la solución GPTE3T, pero además representa una oportunidad para conquistar la curiosidad de los niños por la ciencia y las competencias tecnológicas y el estímulo de vocaciones científicas desde edades tempranas.

.

1. Objetivos

1.1 Objetivo General

Diseñar e implementar una versión mejorada de la herramienta GPTE3T para automatizar el proceso de aprendizaje de la misma a partir de documentos que se van escribiendo sobre la radioastronomía para brindar apoyo a los procesos de socialización en el marco del proyecto "Desarrollo de un arreglo interferométrico de Radio Telescopios para establecer una estación de Radio Astronomía de la UIS en el Páramo de Berlín Santander" que desarrollaron los grupos de investigación CEMOS, RadioGIS y CPS con apoyo de Minciencias.

1.2 Objetivos Específicos

Diseñar un proceso mejorado de cargue de información que permita a la herramienta GPTE3T actualizar automáticamente su contexto a medida que se añaden nuevos documentos y se amplía la base de datos sobre radioastronomía.

Diseñar una interfaz amigable, que sea intuitiva para los niños y simplifique la interacción con la herramienta, fomentando así un ambiente de aprendizaje sencillo y accesible.

Explorar la posibilidad de implementar una versión hablada que responda en tiempo real a las necesidades de los niños.

Realizar pruebas piloto y ajustes basados en la retroalimentación obtenida de estas pruebas para perfeccionar el diseño, la funcionalidad y la interacción de la herramienta con los niños y educadores.

Realizar una documentación para el uso de la herramienta, asegurando que los educadores puedan integrar eficazmente a la herramienta con los niños del páramo y actualizar sus contenidos.

2. Marco Teórico

2.1 Procesamiento del Lenguaje Natural (NLP)

El procesamiento del lenguaje natural (NLP, por sus siglas en inglés) es una subdisciplina de la inteligencia artificial que estudia las interacciones entre las computadoras y el lenguaje humano. Su propósito es permitir que las máquinas comprendan, interpreten, generen y respondan al lenguaje de forma natural, similar a como lo haría una persona. El NLP combina técnicas de lingüística computacional con algoritmos de aprendizaje automático para realizar tareas como clasificación de texto, análisis de sentimientos, traducción automática, generación de lenguaje, entre otras. En el contexto de los modelos de lenguaje como GPT (Generative Pre-trained Transformer), el NLP se convierte en la base que permite generar respuestas coherentes, contextualizadas y adaptadas al usuario.

En este proyecto, el NLP se empleó como núcleo del motor conversacional de la herramienta educativa. Gracias a esta tecnología, el chatbot fue capaz de interpretar las preguntas formuladas por los niños, ya sea por texto o por voz y generar respuestas ajustadas a su nivel de comprensión. La comprensión semántica del lenguaje permitió, además, incorporar estrategias de interacción adaptativa, favoreciendo una experiencia educativa personalizada

2.2 API GPT de OpenAI

La API de OpenAI proporciona acceso a modelos de lenguaje de última generación, basados en arquitecturas de aprendizaje profundo como GPT (Generative Pre-trained Transformer). Esta tecnología permite generar texto de manera coherente y contextual, a partir de entradas en lenguaje natural. La API puede ajustarse a tareas específicas mediante técnicas de

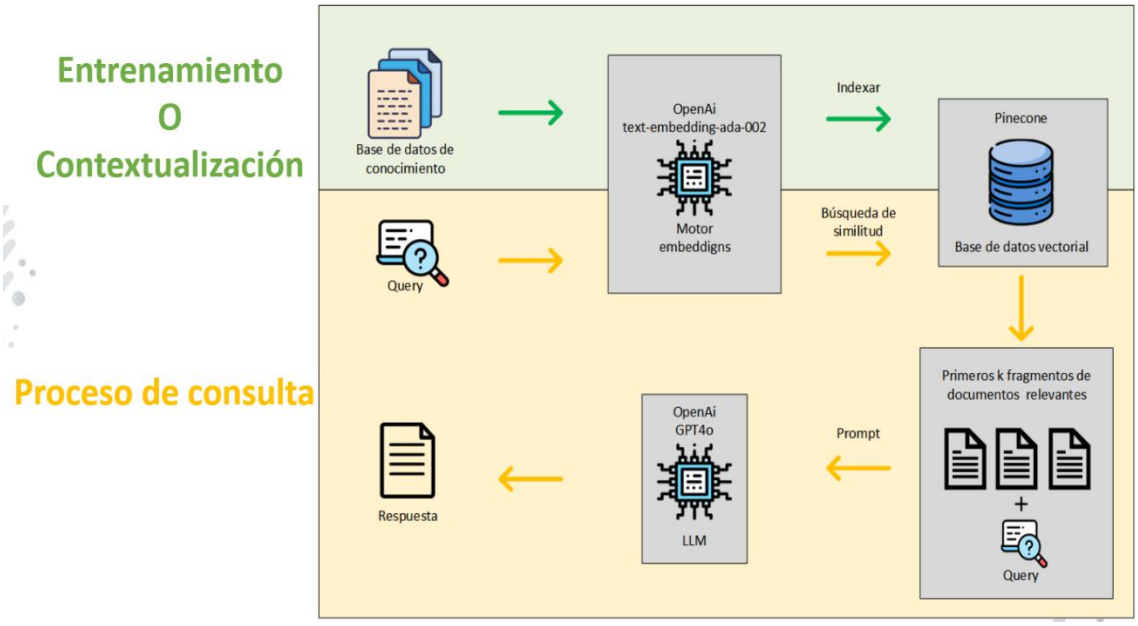
adaptación por contexto, lo que la convierte en una herramienta versátil para el desarrollo de asistentes conversacionales. Una de sus características más relevantes es la capacidad de incorporar información personalizada a través de instrucciones o fragmentos externos, sin necesidad de reentrenar el modelo. Esto permite que los desarrolladores integren sus propios contenidos y definan el estilo de interacción esperado.

En el presente proyecto, la API de OpenAI fue empleada como motor principal del chatbot educativo. A partir de los documentos suministrados sobre radioastronomía, se construyó un sistema que proporciona respuestas claras, breves y adaptadas a la edad del usuario. Además, se diseñaron instrucciones específicas (prompts) que orientan al modelo sobre el tono, el nivel de detalle y el tipo de interacción deseada en cada uno de los dos modos implementados: pregunta/respuesta y aprendizaje interactivo

2.3 Sistema RAG (Retrieval-Angmented Generation)

Figura 1

Diagrama de funcionamiento sistema RAG



El sistema RAG (Retrieval-Augmented Generation) es una técnica utilizada en procesamiento del lenguaje natural que combina dos componentes: la recuperación de información desde una base documental externa y la generación de texto basada en dicha información. A diferencia de los modelos que generan respuestas únicamente desde su entrenamiento previo, RAG permite enriquecer el contexto con documentos específicos y actualizados, lo cual reduce las respuestas imprecisas o "alucinaciones" típicas de los modelos generativos. El funcionamiento de un sistema RAG involucra dos etapas principales. En primer lugar, se realiza una búsqueda semántica a partir de la consulta del usuario, con el fin de identificar los fragmentos de texto más relevantes dentro de una base de datos vectorial. Posteriormente, esos fragmentos se combinan con la consulta original y se envían al modelo generativo (por ejemplo, GPT), que construye la respuesta utilizando exclusivamente ese contexto recuperado.

En este proyecto, se utilizó el enfoque RAG para asegurar que el chatbot educativo respondiera únicamente con base en los documentos proporcionados sobre radioastronomía. Estos documentos se alojaron en una solución de almacenamiento en la nube, lo que facilitó todo el proceso de acceso y actualización. Esta arquitectura permitió actualizar el conocimiento del sistema de manera dinámica y mantener la precisión de las respuestas sin necesidad de modificar el modelo base. La implementación de RAG resultó clave para garantizar que las respuestas ofrecidas estuvieran lineadas con los contenidos específicos de los documentos suministrados.

2.4 Tokenización

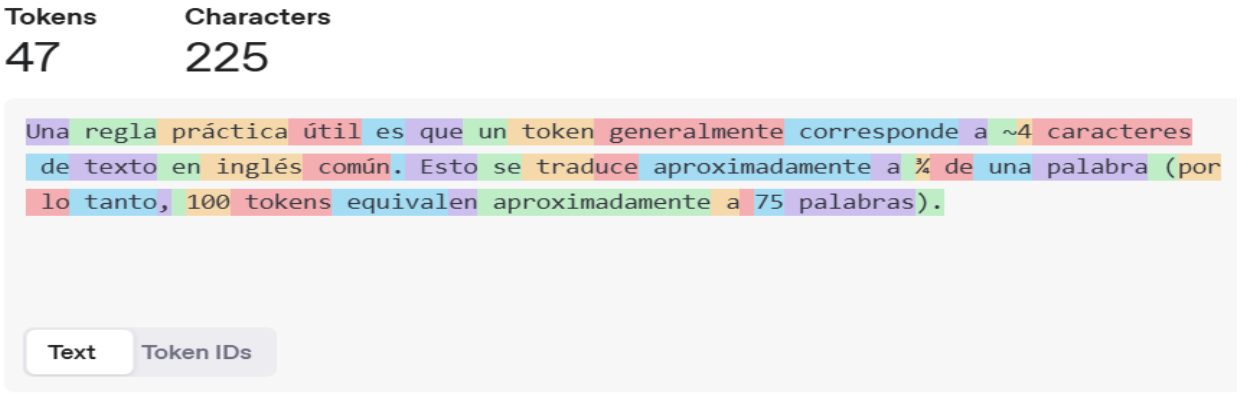
La tokenización es el proceso mediante el cual un modelo de procesamiento del lenguaje natural divide un texto en unidades más pequeñas llamadas tokens estas unidades pueden corresponder a palabras, partes de palabras o incluso caracteres, dependiendo del sistema utilizado.

La tokenización permite que los modelos transformen el texto en representaciones numéricas que pueden ser procesadas por redes neuronales. En modelos como GPT, la tokenización es esencial para convertir preguntas, instrucciones y documentos en una secuencia que el modelo pueda interpretar y relacionar con su entrenamiento, cada token representa una parte del texto, y el modelo predice los siguientes tokens en función del contexto anterior.

A continuación, se muestra un ejemplo:

Figura 2

Ejemplo de un texto tokenizado



Nota. Tomado de https://platform.openai.com/tokenizer

En el desarrollo de la herramienta educativa, la tokenización fue una etapa fundamental para convertir los contenidos de radioastronomía y las preguntas de los niños en secuencias comprensibles para el modelo. Gracias a este proceso, fue posible asegurar que las respuestas generadas mantuvieran coherencia con los documentos suministrados y se adaptaran al estilo de lenguaje definido en los prompts diseñados para cada modo de uso.

2.5 Langchain

LangChain es un framework de desarrollo diseñado para facilitar la construcción de aplicaciones basadas en modelos de lenguaje, su función principal es orquestar la interacción entre

diferentes componentes como interfaces, bases de datos, herramientas de recuperación de información y modelos generativos, mediante el uso de cadenas (chains) y agentes. LangChain permite definir flujos de trabajo personalizados que determinan cómo debe responder un sistema ante una solicitud específica del usuario, una cadena es una secuencia de pasos definidos que se ejecutan de forma estructurada, mientras que un agente tiene la capacidad de decidir qué acción tomar a partir de la entrada recibida. LangChain ofrece integraciones con modelos como GPT, bases de datos vectoriales y herramientas externas, lo que lo convierte en una opción flexible para sistemas conversacionales complejos. En el proyecto, LangChain fue conecta el modelo GPT con la base de datos vectorial, gestionar la recuperación de información relevante y controlar el flujo de interacción entre el usuario y el chatbot, por lo que fue posible definir la lógica que rige cada uno de los modos de uso (pregunta/respuesta y aprendizaje interactivo), así como estructurar la forma en que se consultaban los documentos y se generaban las respuestas adaptadas al perfil del usuario.

2.6 Base de datos vectorial

Una base de datos vectorial es un tipo de sistema diseñado para almacenar y gestionar representaciones numéricas multidimensionales conocidas como embeddings, estas representaciones permiten medir la similitud semántica entre fragmentos de texto, lo cual es fundamental en tareas de recuperación de información basada en significado, y no solo en coincidencias exactas de palabras. Cuando un usuario formula una consulta, el sistema convierte esa entrada en un vector que luego se compara con los vectores previamente almacenados en la base de datos, para encontrar los más similares, se emplean métricas como la similitud coseno y algoritmos de búsqueda eficiente como el k-nearest neighbors (k-NN) o aproximaciones como

Approximate Nearest Neighbors (ANN). En este proyecto, se utilizó una base de datos vectorial almacenada en Pinecone para almacenar los documentos relacionados con la radioastronomía que alimentaban el sistema, estos documentos fueron previamente convertidos en embeddings, al momento de responder una pregunta, el sistema consultaba esta base para recuperar los fragmentos más relevantes, que luego eran usados como contexto para generar respuestas a través del modelo GPT, esta arquitectura permitió mantener el sistema actualizado y centrado exclusivamente en los contenidos definidos por el proyecto, garantizando precisión y control en las respuestas generadas.

2.7 Interacción por voz (STT y TTS con Google Cloud)

La interacción por voz en sistemas conversacionales permite ampliar la accesibilidad y mejorar la experiencia del usuario, especialmente cuando se trata de públicos como niños o personas con dificultades para escribir, esta funcionalidad se compone habitualmente de dos procesos fundamentales: el reconocimiento automático de voz (Speech-to-Text, STT) y la síntesis de voz (Text-to-Speech, TTS). El módulo de STT se encarga de convertir la entrada de voz del usuario en texto que pueda ser interpretado por el sistema, para ello, se utilizan modelos acústicos entrenados con grandes volúmenes de datos lingüísticos. Por su parte, el módulo TTS transforma las respuestas generadas por el sistema en voz sintética, permitiendo una comunicación fluida y bidireccional. En el presente proyecto, ambos procesos fueron implementados mediante los servicios de Google Cloud, los cuales ofrecen modelos de alta precisión, soportan múltiples acentos y permiten ajustes personalizados en el tono, la velocidad y la entonación de la voz generada, el reconocimiento de voz fue integrado como método de entrada principal en la interfaz infantil, mientras que la síntesis de voz se utilizó para entregar las respuestas de forma auditiva, favoreciendo la comprensión y el dinamismo de la experiencia educativa.

3. Desarrollo de la solución

3.1 Diagnóstico de la herramienta original

El desarrollo de esta solución tuvo como punto de partida las limitaciones funcionales detectadas en la versión original del chatbot GPTE3T, una herramienta inicialmente diseñada para proporcionar información sobre el proceso de acreditación ABET en la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones (E3T), entre dichas limitaciones se encontraban la necesidad de realizar el entrenamiento de forma manual, una interfaz básica poco adecuada para usuarios infantiles, y la falta de interacción por voz, lo que restringía significativamente su aplicabilidad en contextos educativos diversos.

Uno de los principales retos técnicos consistió en diseñar un mecanismo simple y robusto para cargar y actualizar automáticamente la información con la que respondería el chatbot, sin requerir intervención constante del equipo de desarrollo, también algunas librerías ya estaban desactualizadas y creaban problemas en el funcionamiento del chatbot, se identificó la necesidad de mejorar la experiencia de interacción, superando la rigidez de una interfaz exclusivamente textual, que dificultaba el uso por parte de niños en etapas tempranas de aprendizaje.

3.2 Nuevo diseño orientado a los nuevos objetivos pedagógicos

A partir de estas observaciones, se planteó el rediseño e implementación de una nueva versión de GPTE3T, enfocada en facilitar el aprendizaje de la radioastronomía entre los niños del Páramo de Santurbán, esta solución se integró como apoyo al proyecto “Desarrollo de un arreglo interferométrico de radio telescopios para establecer una estación de radioastronomía en el Páramo de Berlín”, liderado por grupos de investigación de la E3T con respaldo de Minciencias.

3.3 Diseño de la arquitectura

La arquitectura adoptó el enfoque de recuperación aumentada por generación (RAG), permitiendo combinar la búsqueda semántica de información con la generación de respuestas a partir del modelo GPT, para ello, se utilizó una base de datos vectorial para almacenar los documentos suministrados, un sistema de recuperación que selecciona los fragmentos más relevantes y el modelo de lenguaje de OpenAI (accedido mediante su API), responsable de construir respuestas contextualizadas, la interacción entre estos módulos fue orquestada mediante el framework LangChain, mientras que el flujo general de la aplicación fue implementado usando flask, esto inluyó el manejo de sesiones y usuarios, manejo de rutas y la entrega de vistas web .

3.4 Implementación de funcionalidades conversacionales

El núcleo conversacional de la herramienta se basó en técnicas de procesamiento del lenguaje natural (NLP), lo cual permitió interpretar preguntas formuladas por los niños ya sea por voz o por texto y generar respuestas comprensibles, breves y adaptadas a su nivel de comprensión, esta capacidad se implementó mediante la API de GPT de OpenAI, utilizando plantillas de instrucciones específicas (prompts) diseñadas para cada modo de uso.

La solución contempló dos modos principales, uno de tipo pregunta/respuesta, que permite al niño resolver dudas puntuales mediante explicaciones simples contextualizadas y uno de aprendizaje interactivo, que ofrece explicaciones paso a paso acompañadas de preguntas formativas, juegos interactivos y retroalimentación positiva.

Ambos modos fueron diseñados para adaptarse al nivel de desarrollo del usuario, además, se incorporaron funcionalidades de entrada y salida por voz utilizando los servicios de Google Cloud (Speech-to-Text y Text-to-Speech), junto con un sistema de gestión de sesiones capaz de

registrar la interacción por usuario, resumir el proceso de aprendizaje al finalizar y retomar el contexto en futuras visitas, este mecanismo representa un valor agregado ya que el sistema genera resúmenes automáticos al final de cada sesión, lo que no solo permite mantener una trazabilidad del proceso de aprendizaje, sino que también aporta insumos útiles para su análisis pedagógico y la integraciones en entornos escolares llevando un monitoreo docente.

3.5 Método del desarrollo

El desarrollo se llevó a cabo siguiendo un enfoque iterativo e incremental, centrado en el prototipado funcional. A lo largo del proceso se realizaron ajustes progresivos en la arquitectura, el diseño de la interfaz y los modos de interacción, basado en pruebas piloto y en observaciones obtenidas durante las sesiones de validación con usuarios. Este enfoque permitió optimizar el comportamiento del sistema, mejorar su accesibilidad y asegurar la alineación entre la solución técnica y los objetivos pedagógicos del proyecto.

Cabe destacar que el modo interactivo surgió como resultado de este proceso de desarrollo progresivo. Las observaciones realizadas en los primeros prototipos mostraron la necesidad de ofrecer una interacción más estructurada y estimulantes, lo cual finalmente motivó el diseño de una manera diferente de manejar la metodología RAG estándar.

Como resultado de este proceso de iteración y validación, se tomaron una serie de decisiones técnicas clave que guiaron la implementación final de la herramienta y garantizaron su alineación con los objetivos pedagógicos del proyecto, las cuales serán expuestas a continuación.

3.5.1 Arquitectura RAG (Retrieval-Augmented Generation):

Se adoptó el enfoque RAG porque permite integrar un modelo de lenguaje (GPT) con un sistema de recuperación semántica basado en embeddings vectoriales, garantizando que las respuestas estén fundamentadas en los documentos cargados, sin reentrenar el modelo base.

3.5.2 Modelo Generativo GPT:

Se seleccionó la API de OpenAI (GPT) por su capacidad para generar respuestas naturales, coherentes y adaptadas al contexto, a través de prompts específicos, el modelo ajusta su tono y nivel de detalle según la edad del usuario.

3.5.3 Framework LangChain:

Utilizado para gestionar el flujo conversacional, organizando cadenas de operaciones que incluyen:

a. Búsqueda semántica en la base de datos vectorial.
b. Inserción de contexto relevante en el prompt.
c. Generación de respuesta por GPT.
d. Manejo de variables dinámicas (nombre, edad, historial).

3.5.4 Servicios de voz (Google Cloud STT/TTS):

Implementados para permitir interacción multimodal. STT (Speech-to-Text) convierte las preguntas orales en texto, y TTS (Text-to-Speech) sintetiza respuestas con voces amigables.

3.5.5 Base de datos vectorial en la nube (Pinecone):

Seleccionada por su eficiencia en búsquedas semánticas rápidas y escalables, sin la complejidad ni costo computacional asociadas a una opción local.

3.5.6 Interfaz HTML y CSS:

Se optó por un diseño con colores suaves (verde claro, blanco), tipografía redondeada y elementos gráficos relacionados con el espacio, para un entorno visual atractivo para niños.

3.6 Aportes significativos de la solución

En conjunto, esta solución representó una evolución significativa respecto al sistema original, al transformar un chatbot institucional en una herramienta educativa, accesible, personalizable y alineada con los principios de apropiación social del conocimiento científico en contextos escolares, adicionalmente, el enfoque modificado del método RAG, donde el contexto no se actualiza en cada intervención, sino que se conversa durante toda la sesión y solo se reemplaza si el usuario decide cambiar de tema, representa un aporte metodológico que permite mejorar la coherencia, reducir el costo computacional (el cual resulta en una reducción de costos) y mantener la continuidad pedagógica en sesiones extendidas.

3.7 Arquitectura general del sistema

Una vez se definieron las decisiones técnicas, fue necesario estructurar los distintos componentes de la solución dentro de una arquitectura modular y escalable.

Esta organización se basa en el enfoque de generación aumentada por recuperación (Retrieval Augmented Generation).

El sistema se organizó en seis módulos funcionales, cuya interacción sigue un flujo estructurado que garantiza coherencia conversacional, personalización y continuidad pedagógica. A continuación, se describe el rol de cada módulo en el diseño global:

- **Interfaz de usuario:** Diseñada para facilitar la interacción del usuario con el sistema, permite seleccionar el modo de uso, ingresar consultas mediante texto o voz, y visualizar o escuchar las respuestas del chatbot, incluye pantallas para el registro, inicio de sesión, selección de modo y ventana de conversación, su diseño fue orientado a la usabilidad infantil.
- **Gestión de sesiones y memoria:** Este módulo controla la autenticación de usuarios, el historial de aprendizaje por sesión y la continuidad educativa, permite recuperar información de interacciones anteriores y genera un resumen de cada sesión, está vinculado con la base de datos y con la lógica conversacional.
- **Recuperación semántica de información:** Responsable de identificar, dentro de una base de documentos, los fragmentos más relevantes para cada consulta, estos fragmentos son utilizados para alimentar el contexto que se le proporciona al modelo de lenguaje, garantizando que las respuestas estén fundamentadas en conocimiento validado por el proyecto.
- **Generación de respuestas:** Este módulo representa al motor de inteligencia artificial que, a partir del contexto recuperado y la pregunta del usuario, produce una respuesta escrita en lenguaje natural, su función es interpretar la entrada, junto a las intrucciones y generar un mensaje claro y útil, adaptado al perfil del usuario.
- **Flujo conversacional:** Coordina la interacción entre los módulos, gestiona el orden de ejecución de cada componente (recuperación, generación, almacenamiento), la incorporación de variables como edad o nombre del usuario, y la estructura del diálogo, según el modo seleccionado (pregunta/respuesta o aprendizaje interactivo).

- **Entrada y salida por voz:** Este módulo permite que las consultas se realicen mediante reconocimiento de voz, y que las respuestas generadas sean entregadas como audio, se integra directamente con la interfaz de usuario, ampliando la accesibilidad de la herramienta.
- **Carga y procesamiento de documentos externos:** Permite actualizar automáticamente la base de conocimiento desde una carpeta de Google Drive, procesando y vectorizando los documentos sin requerir intervención técnica.
- **Panel de administración:** Diseñado para que los docentes/administradores puedan supervisar el aprendizaje de los usuarios.

Esta arquitectura permitió construir una herramienta educativa funcional, personalizable y centrada en el usuario, su diseño modular garantiza la escalabilidad hacia otros contenidos o públicos, mientras que su implementación técnica asegura precisión, control de contexto y continuidad pedagógica en cada sesión.

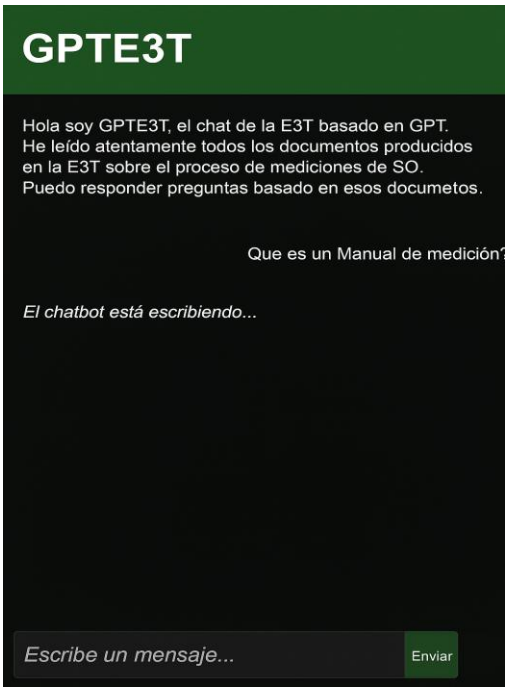
3.8 Descripción de los módulos implementados

Durante el proceso de desarrollo, se implementaron y adaptaron diversos componentes que permitieron transformar al chatbot GPTE3T en una herramienta educativa funcional, adaptativa y centrada en el público infantil, a continuación, se describen los módulos clave implementados y las decisiones técnicas adoptadas en cada uno.

3.8.1 Interfaz gráfica

Figura 3

Interfaz gráfica GPTE3T



Nota. Tomado de https://n9.cl/xj4cj

Figura 4

Interfaz gráfica BOTE3T

En la **Figura 3** muestra la interfaz principal del chatbot GPTE3T, utilizada como base para el rediseño de la herramienta educativa, esta interfaz presenta una estructura de chat sencilla, con un área para la entrada de texto y un espacio para las respuestas generadas por el modelo, su diseño inicial estaba orientado a consultas institucionales relacionadas con la acreditación ABET, por lo que carecía de elementos visuales y funciones específicas para usuarios infantiles, lo que motivó su adaptación en este proyecto.

La interfaz de usuario de BOTE3T fue desarrollada empleando HTML y CSS, sin estructuras adicionales, con el objetivo de mantener una estructura ligera, flexible y fácil de desplegar en entornos web básicos, el diseño se centró en ofrecer una experiencia visual amigable para niños, con elementos gráficos alusivos al espacio, botones amplios, emojis, colores suaves

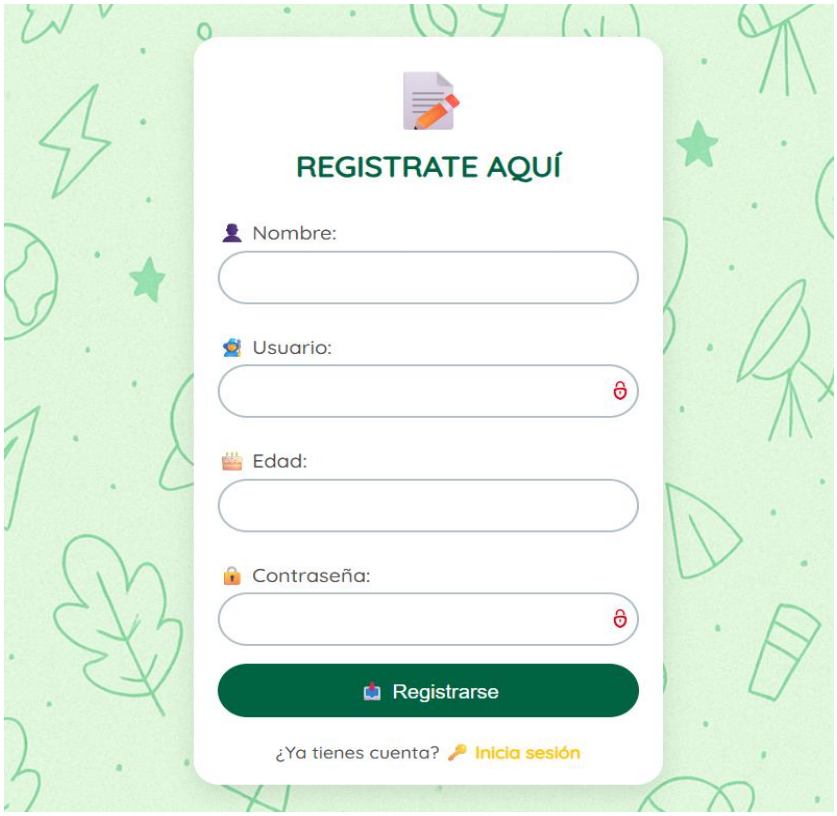
(principalmente tonos verdes representativos de la UIS) y tipografía redondeada, lo que refuerza el carácter lúdico y educativo de la herramienta.

La **Figura 4** muestra la interfaz principal de BOTE3T, en la que se evidencia la transformación realizada a partir de la versión original de GPTE3T, esta nueva interfaz se diseñó no solo para consultas textuales, sino también para interacción mediante voz, mejorando la accesibilidad y ofreciendo una experiencia más intuitiva. A continuación, se observan las pantallas del interfaz:

a. Pantalla de registro: permite crear un perfil ingresando nombre, edad, usuario y contraseña.

Figura 5

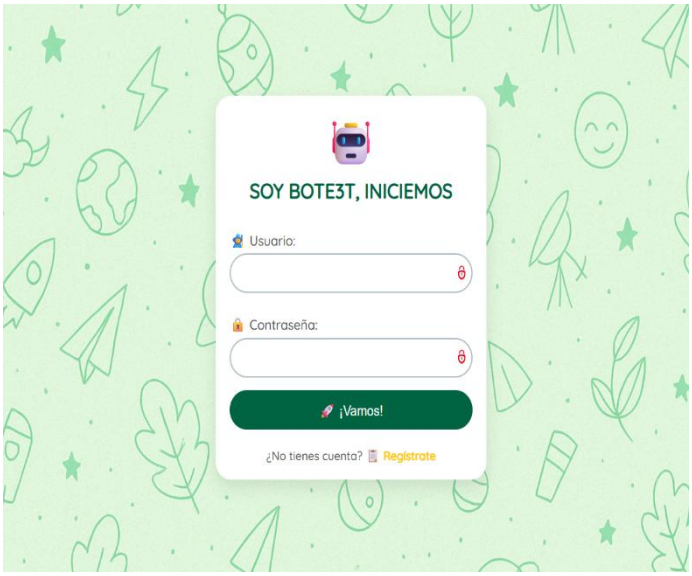
Pantalla de registro



b. **Inicio de sesión**: acceso al sistema por medio de credenciales registradas.

Figura 6

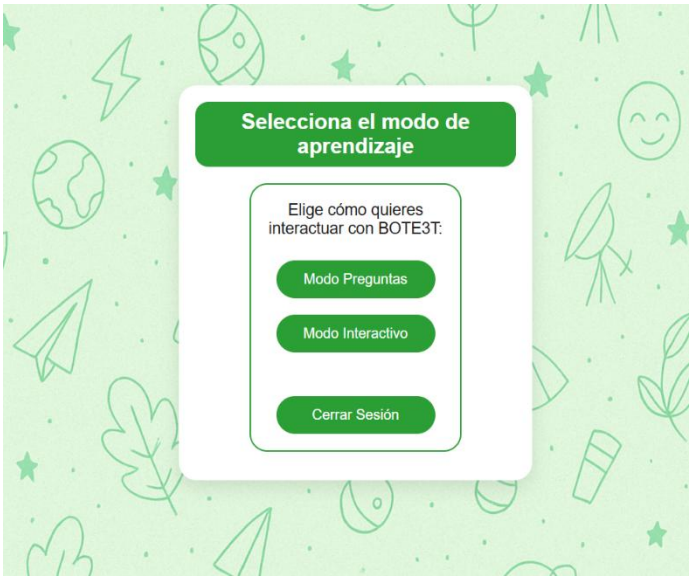
Pantalla de inicio de sesión



c. **Selección de modo**: permite escoger entre “modo preguntas” o “modo interactivo”.

Figura 7

Pantalla de selección del modo



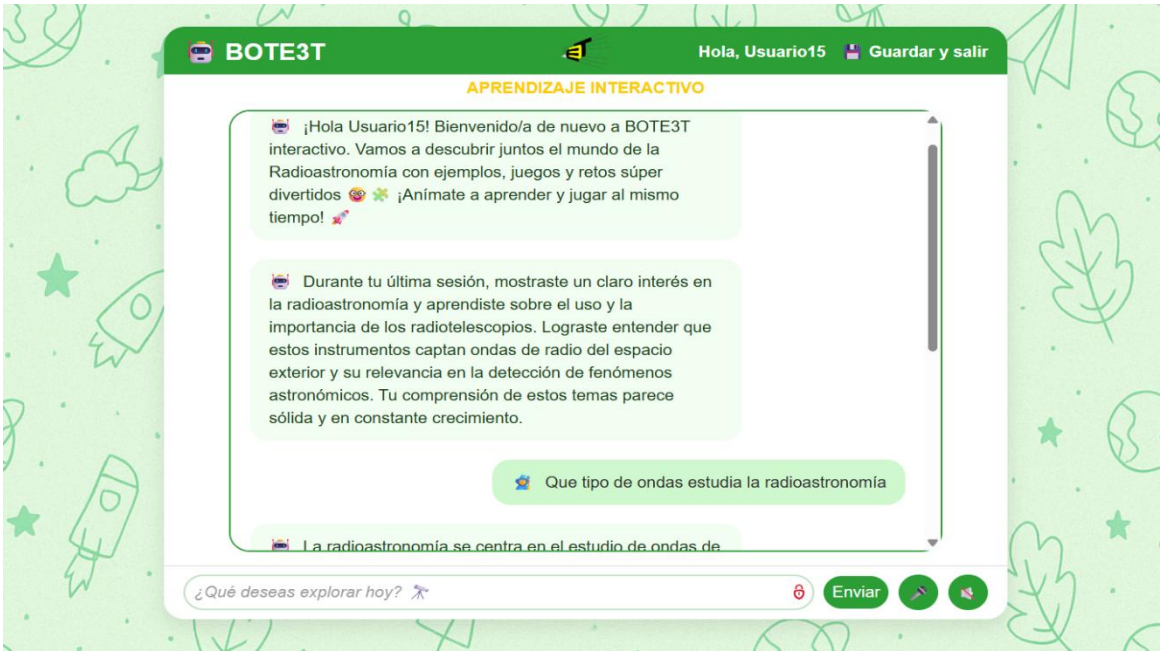
d. **Zona conversacional**: espacios donde se produce la interacción entre el niño y BOTE3T.

Figura 8

Pantalla modo preguntas

Figura 9

Pantalla modo interactivo



Cada pantalla se conectó mediante rutas internas gestionadas por JavaScript, que activaban o desactivaban las secciones visibles en la interfaz según el flujo del usuario, permitiendo una navegación dinámica sin necesidad de recargar la página.

En el **Apéndice C** se encuentra disponible el video demostrativo del funcionamiento de la herramienta educativa BOTE3T, este material audiovisual tiene como objetivo evidenciar las principales características de la solución, su interfaz de usuario y las funcionalidades implementadas, mostrando de forma práctica cómo el sistema responde a las interacciones propuestas, el video complementa la información descrita en este informe, proporcionando una visión más dinámica y clara de la experiencia de uso.

3.8.2 Gestión de sesiones y memoria

Cada usuario registrado al iniciar sesión, el sistema verificaba su existencia en la base de datos local y, luego de seleccionar un modo, recuperaba el historial reciente y el resumen de la última sesión, esta información se mostraba como parte del saludo inicial.

Durante la sesión:

- Se almacenaban todas las interacciones entre el usuario y el sistema.
- Se creaban resúmenes automáticos de toda la sesión mediante una llamada a GPT al finalizar.
- Se gestionaba la continuidad de contenido según el modo y los temas tratados, usando los resúmenes y el historial de interacciones.

Para llevar a cabo este proceso se definió una base de datos relacional implementada con SQLAlchemy, una librería de python que permite el manejo de bases de datos SQL de manera sencilla. Las tablas que se usaron fueron:

a. **User:** almacena los datos personales de cada usuario (nombre, edad, correo, contraseña cifrada y rol).
b. **Conversation:** registra cada pregunta y respuesta, asociadas al usuario, sesión y modo de interacción.
c. **SessionSummary:** guarda un resumen general de cada sesión al finalizar, permitiendo ofrecer continuidad y trazabilidad del aprendizaje.
d. **GuidedContext:** almacena el contexto recuperado por RAG en el modo interactivo, lo que permite retomar un mismo tema en futuras sesiones.

El flujo paso a paso está representado en las Figuras 18 y 19, y su implementación se integró con el módulo de LangChain para mantener el contexto activo.

3.8.3 Gestión de sesiones y memoria

Se utilizó una arquitectura RAG para garantizar que las respuestas del chatbot estuvieran basadas exclusivamente en los documentos suministrados por el equipo del proyecto.

Componentes clave:

a. **Pinecone como base vectorial:** se indexaron los documentos procesados en fragmentos de 1000 caracteres. Cada fragmento fue convertido en un embedding utilizando el modelo text-embedding-ada-002 de OpenAI.
b. **Búsqueda semántica:** al recibir una consulta, el sistema generaba un vector de la pregunta y recuperaba los fragmentos más cercanos en el espacio vectorial, retornando los 3 vectores más relevantes.
c. **Inyección de contexto:** los fragmentos recuperados eran formateados e insertados en la plantilla del prompt antes de enviarlo a GPT.

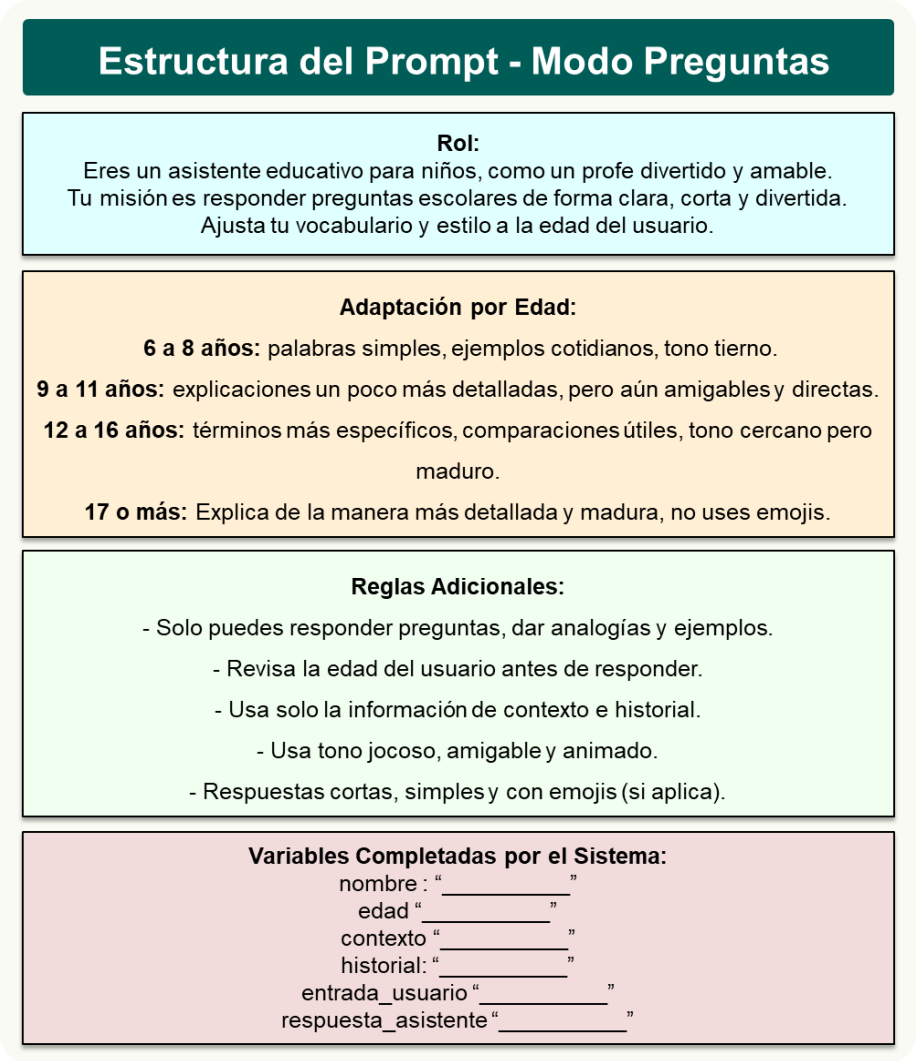
3.8.4 Generación de respuestas

La generación de respuestas fue gestionada mediante la API de OpenAI, configurada con un modelo de lenguaje del tipo GPT-4o. Se diseñaron dos prompts diferenciados:

a. **Modo pregunta/respuesta**: Instrucciones orientadas a entregar respuestas breves, claras, con lenguaje adaptado a la edad y tono divertido, el prompt incluye reglas específicas de adaptación lingüística para el entendimiento de cada usuario, ejemplos, emojis, y comportamientos ante preguntas no cubiertas.

Figura 10

Diagrama prompt modo preguntas

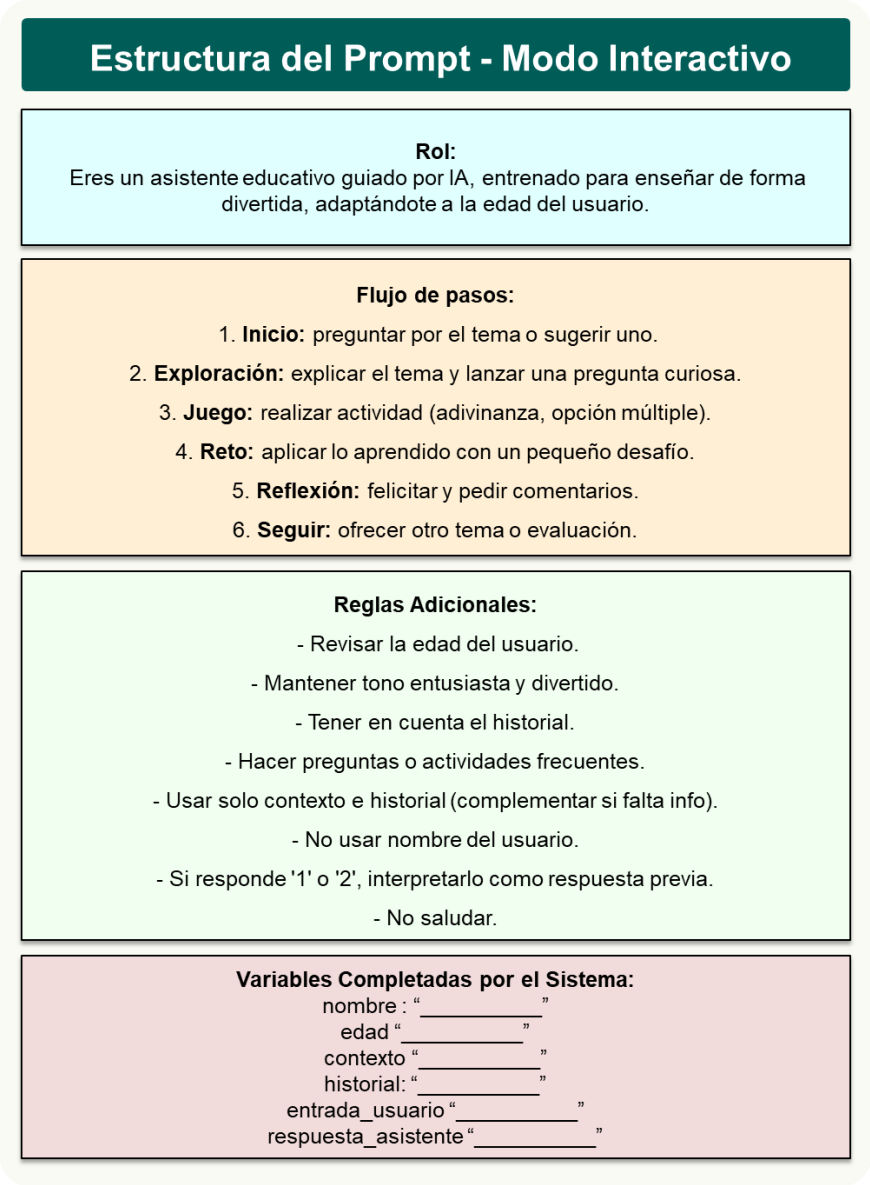


El **prompt - modo preguntas** establece una estructura técnica que guía al chatbot para comportarse como un asistente educativo especializado, define un **rol**, asegurando que las respuestas tengan un tono amigable y pedagógico, y mediante la **adaptación por edad** ajusta el nivel de lenguaje, complejidad y ejemplos en función de la variable {edad}, las **reglas adicionales** actúan como un sistema de control, limitando al chatbot a responder únicamente preguntas, usar analogías, mantener un estilo breve y claro basado en el contexto, además, las **variables del sistema** ({nombre}, {historial}, {contexto}, {entrada_usuario}, {respuesta_asistente}) permiten personalizar cada respuesta, conservar coherencia con la conversación previa y mejorar la experiencia del usuario mediante una interacción dinámica y contextualizada.

b. **Modo de aprendizaje interactivo**: Diseñado para guiar la interacción de forma progresiva, manteniendo el interés del usuario y ofreciendo retroalimentación constante, cada parte del proceso incorpora instrucciones claras para garantizar un lenguaje accesible, lúdico y cercano, integrando minijuegos, emojis y adaptaciones dinámicas basadas en la edad del usuario.

Figura 11

Diagrama prompt modo interactivo



El **prompt - modo interactivo** define un flujo dinámico para que el chatbot actúe como un asistente educativo que guía el aprendizaje de forma entretenida y adaptada a la edad del usuario, a través del **rol**, se establece su función como un profesor virtual con un tono cercano y divertido, el **flujo de pasos** organiza la interacción en seis fases: inicio del tema, exploración con preguntas curiosas, actividades de juego, retos para aplicar lo aprendido, reflexión con

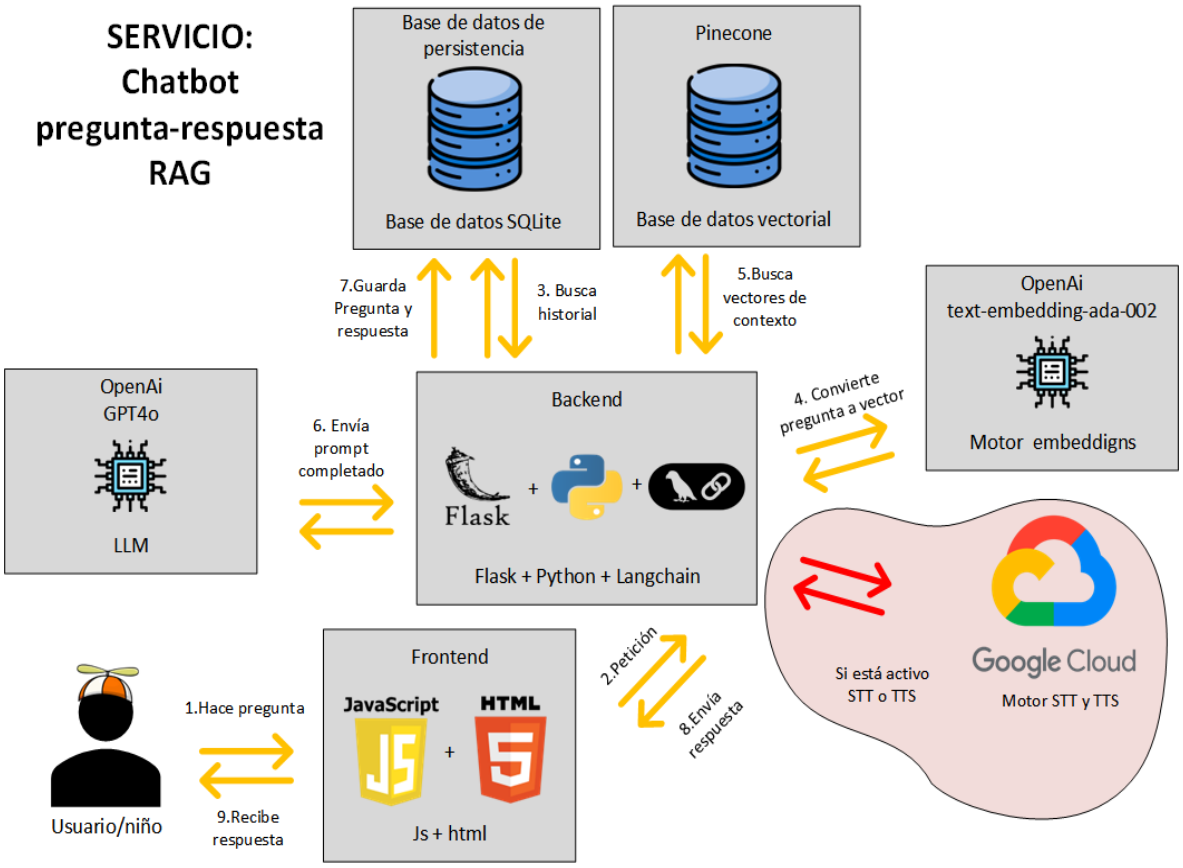
retroalimentación y seguimiento para continuar el aprendizaje, las **reglas adicionales** imponen límites técnicos, como usar únicamente el contexto e historial, interpretar respuestas numéricas como opciones previas y mantener un tono entusiasta, evitando saludos innecesarios, por último, las **variables del sistema** (nombre, edad, historial, contexto, entrada_usuario, respuesta_asistente) permiten personalizar cada respuesta y mantener coherencia en todo el diálogo.

Ambos prompts se generaban dinámicamente en cada sesión, insertando los fragmentos recuperados, el nombre, edad del usuario, así como las demás variables de diálogo.

3.8.5 Flujo conversacional

Figura 12

D*iagrama RAG modo preguntas*



Se utilizó LangChain como herramienta para orquestar los distintos pasos del flujo conversacional, este framework permitió:

- Encadenar procesos como recuperación semántica, construcción de prompt, llamada a GPT, posprocesamiento de respuesta.
- Inyectar variables dinámicas (edad, nombre, historial, fragmentos).

LangChain también facilitó la adaptación de los modos de uso como flujos independientes reutilizables. Además, permite la flexibilidad de cambiar tanto la base vectorial usada como el modelo de LLM.

3.8.6 Entrada y salida por voz

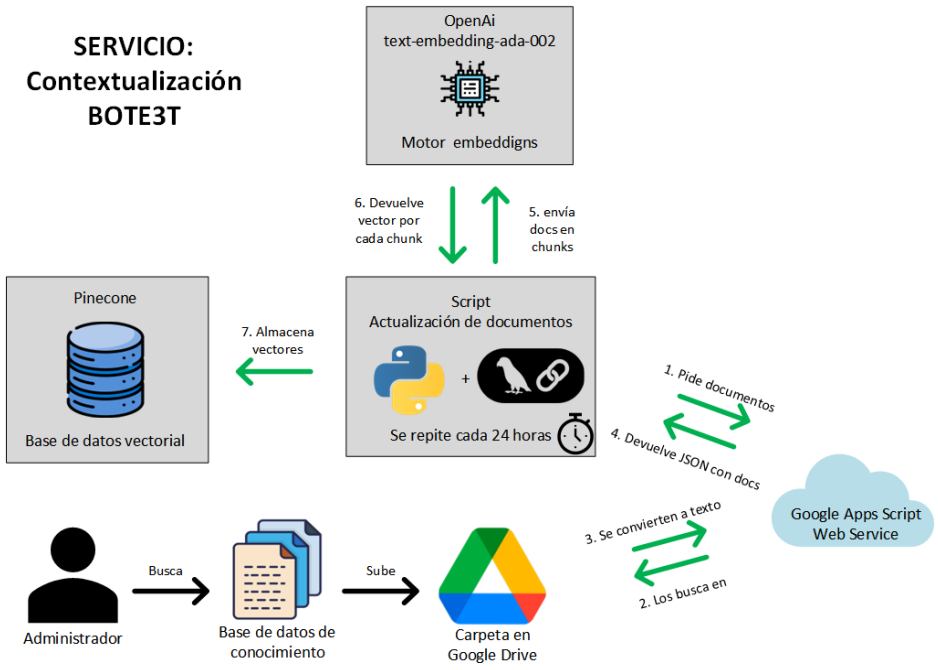
Para permitir una experiencia multimodal, se integraron dos servicios de Google Cloud:

a. **Speech-to-Text (STT):** transcribía las preguntas orales del usuario y las convertía en texto. Se utilizó el modelo para español.
b. **Text-to-Speech (TTS):** generaba la respuesta hablada utilizando una voz femenina (modelo es-US-Neural2-A), con ajustes de tono suave y velocidad reducida, para mejorar la comprensión en edades tempranas.

3.8.7 Carga y procesamiento de documentos externos

Figura 13

Diagrama contextualización



En la versión original de la herramienta, la actualización del conocimiento requería ejecutar manualmente un script desde la plataforma de Google Colab, lo que implicaba conocimientos técnicos por parte de los usuarios que actuaban como administradores, esta limitación dificultaba la participación directa de docentes o personas que no tuviera conocimientos técnicos.

El nuevo mecanismo se diseñó con el objetivo de hacerlo accesible a cualquier usuario no técnico, este permite que todo el conocimiento del chabot sean gestionados directamente desde una carpeta compartida de Google Drive, facilitando que uno o varios miembros de un equipo puedan incorporar, modificar o eliminar información sin necesidad del equipo de desarrollo.

El acceso a estos documentos se realiza mediante un WebService desarrollado en Google Apps Script, que nos devuelve el texto dentro de los documentos que se encuentren en la carpeta cual le indiquemos, estos contenidos son consumidos por un script que se ejecuta automáticamente

desde el VPS (servidor virtual privado) el cual los procesará, dividiéndolos, y transformándolos en vectores usando la API de embeddings de OpenAI, luego, los vectores generados son almacenados en una base de datos vectorial en Pinecone, la cual actúa como repositorio de información para el sistema, además, se emplea una archivo de metadatos local para detectar cambios en la carpeta de google drive y asegurar que estos sean reflejados en la base de datos vectorial, el flujo que sigue el sistema, paso a paso, se encuentra en la figura 17, junto a una descripción corta de cada paso.

Esta arquitectura permite que el sistema se mantenga actualizado, alineado con los contenidos educativos definidos por el proyecto siendo gestionable por personas sin conocimiento técnico, democratizando el mantenimiento del sistema.

En el **Apéndice F** se describen las principales consideraciones técnicas y conceptuales que guiaron el diseño y desarrollo de la solución BOTE3T, se incluyen aspectos relacionados con las decisiones de arquitectura, selección de tecnologías, criterios de usabilidad y factores que garantizan la escalabilidad y adaptabilidad del sistema.

3.8.8 Panel de administración

Figura 14

Botón panel admin en pantalla de modos

Para facilitar la supervisión del sistema y brindar herramientas de gestión a los docentes y administradores del proyecto, se desarrolló un panel de administración accesible solo para usuarios con rol de administrador, ver Figura 14.

Figura 15

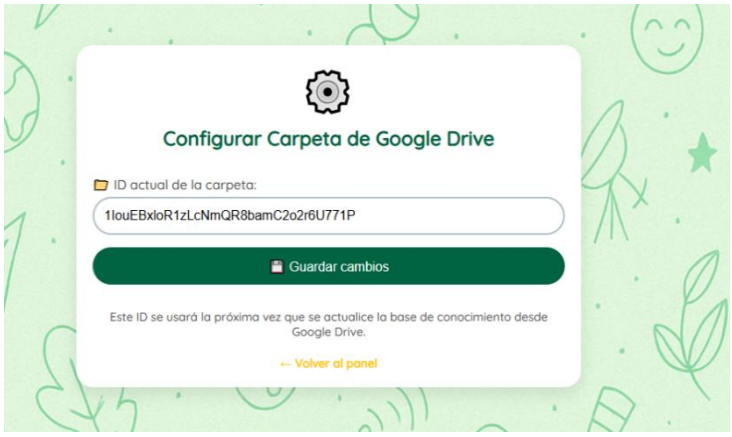
Panel de administración

En la Figura 15, este panel permite fácilmente:

- Visualizar conversaciones recientes filtrando por usuario.
- Filtrar registros por usuario o por tipo de modo (preguntas/interactivo).
- Consultar los resúmenes automáticos generados por el sistema al finalizar cada sesión.
- Redireccionar fácilmente a la configuración de la carpeta de carga de documentos desde Google Drive, ver Figura 16.

Figura 16

Panel de configuración de la carpeta de Google Drive



Esta interfaz fue diseñada pensando en principios de simplicidad y usabilidad, de modo que cualquier docente o miembro del equipo pueda acceder a los reportes sin necesidad de conocimientos técnicos.

En el **Apéndice D** se presenta la documentación elaborada para el uso de la herramienta educativa, uno de los objetivos de este proyecto con el fin de facilitar su integración, este material está diseñado para guiar a los educadores en el uso de BOTE3T, la documentación complementa este informe, garantizando una implementación efectiva y sostenible de la herramienta.

3.9 Flujo funcional

Esta sección funciona como una guía paso a paso de los procesos más importantes que realiza el sistema para su funcionamiento.

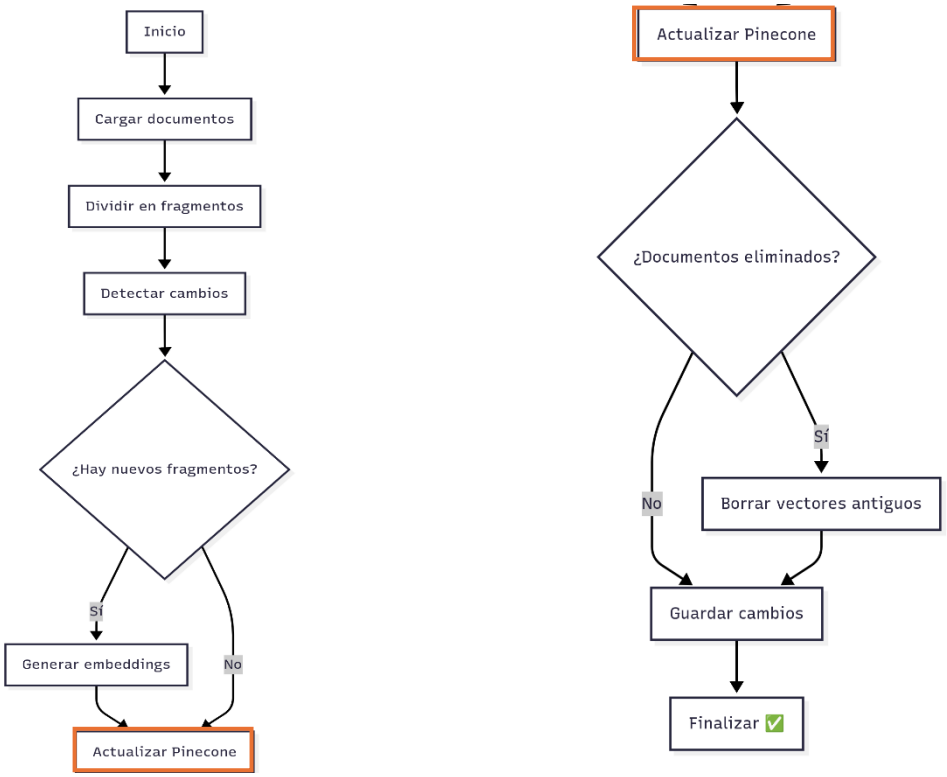
A lo largo del desarrollo del sistema, se definieron e implementaron distintos flujos funcionales que articulan cómo el usuario interactúa con la herramienta, cómo se actualiza su base de conocimiento y cómo se gestionan las sesiones personalizadas de aprendizaje.

Se definieron cuatro flujos principales: el flujo de contextualización, el flujo de gestión de sesión y dos flujos de interacción conversacional, uno para cada modo, todos representados en los diagramas funcionales elaborados durante el diseño del sistema.

3.9.1 Flujo de contextualización

Figura 17

Flujo contextualización



Este diagrama de flujo se encuentra disponible en el repositorio del proyecto (ver **Apéndice A**).

Una de las mejoras más importantes del sistema fue la implementación de un mecanismo para la actualización automatizada de la base de conocimiento del chatbot sin requerir del usuario administrador más que subir el documento a una carpeta de Google drive, este proceso, mostrado en las Figuras 13 y 17 asegura que los cambios en la documentación fuente (Google drive) se vean reflejados en la base de conocimiento del bot (base vectorial, en pinecone), esto incluye la adición, modificación o eliminación de documentos.

El flujo completo se ejecuta periódicamente o cuando se necesite, y sigue los siguientes pasos:

a. **Carga de documentos en Google Drive:** El usuario administrador de la documentación sube nuevos documentos (PDF, DOCX, o TXT) a una carpeta específica en Google drive.
b. **Ejecución del script de actualización:** El sistema ejecuta un script local, de manera programada cada 24 horas.
c. **Consumo del WebService:** Un script que desarrollado en GAS (Google Apps Script) consulta la carpeta de Google drive señalada en la llamada del WebService, y nos responde con el nombre de los documentos y el contenido textual de ellos.
d. **Segmentación y procesamientos:** Cada documento es dividido en fragmentos de 1.000 caracteres con un solapamiento de 100 caracteres, este solapamiento le permite al sistema entender el orden entre fragmentos y su relación entre ellos.
e. **Generación de identificadores únicos (hash):** A cada fragmento se le calcula una huella digital única mediante un algoritmo SHA-256, este identificador permite determinar si un

fragmento ya se había almacenado anteriormente, solo los fragmentos que no se encuentren en la lista de fragmentos ya registrados son procesados.

f. **Vectorización mediante API de embeddings:** Cada fragmento que se considere nuevo es convertido a una representación vectorial utilizando el modelo “text-embedding-ada-002” de OpenAI.

g. **Indexación en Pinecone:** Los vectores generados se almacenan en la base de datos vectorial en Pinecone.

h. **Eliminación de fragmentos obsoletos:** El sistema compara los nombres de archivos que entregó el WebService con los archivos previamente indexados, si un archivo que existía en la base vectorial ya no está presente en la carpeta de Drive, se considera eliminado, en ese caso, todos los fragmentos asociados a dicho archivo (identificados por sus hashes) son eliminados de Pinecone para mantener la coherencia entre el almacenamiento documental y la base vectorial.

i. **Base actualizada y lista para consulta:** Una vez completado el proceso, la base de conocimiento está actualizada y lista para ser utilizada en las respuestas del sistema.

En el **Apéndice E** se presenta el detalle de los costos asociados al funcionamiento de la herramienta BOTE3T, incluyendo los recursos técnicos, servicios en la nube y otros gastos operativos necesarios para garantizar su disponibilidad y correcto desempeño, este análisis permite comprender la inversión requerida para la sostenibilidad y actualización del sistema a lo largo del tiempo.

3.9.2 Flujo de gestión de sesión y resúmenes

Figura 18

Flujo inicio de sesión

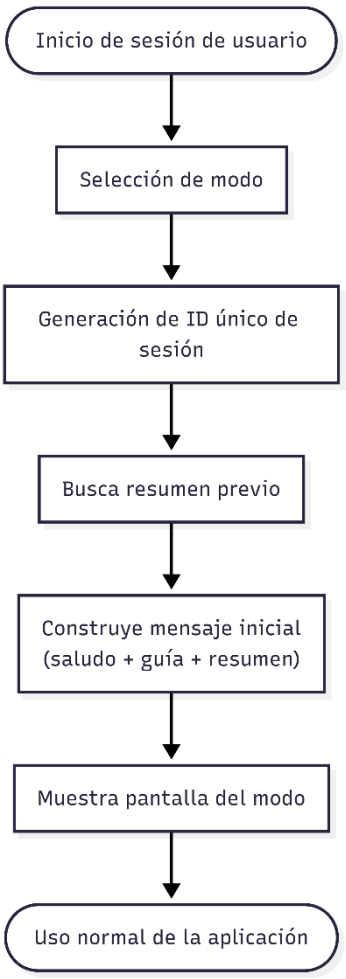
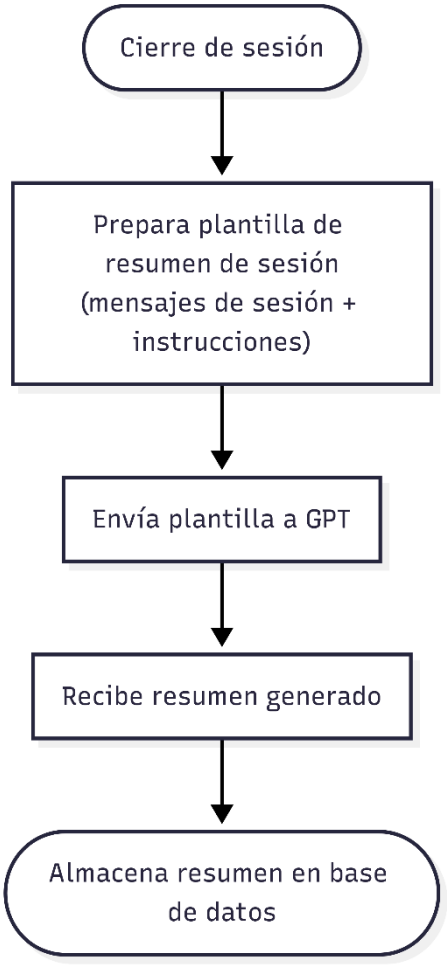


Figura 19

Flujo cierre de sesión



Este diagrama de flujo se encuentra disponible en el repositorio del proyecto (ver **Apéndice A**).

Este flujo corresponde al proceso descritos en las Figuras 18 y 19, y tienen como objetivo identificar al usuario, gestionar su historial de aprendizaje y preparar el entorno personalizado de interacción.

Flujo de inicio de sesión:

a. **Inicio del sistema:** El usuario accede a la interfaz de loggeo o ingreso.
b. **Inicio de sesión o registro:** Se solicita el ingreso del nombre de usuario y contraseña. En caso de no estar registrado, el sistema permite crear una cuenta nueva, solicitando también la edad.
c. **Validación de credenciales:** Se verifica la información ingresada.
d. **Selección del modo de interacción:** El usuario escoge entre:
    - Modo pregunta/respuesta
    - Modo de aprendizaje interactivo
e. **Asignación de identificador único:** El sistema genera un ID de sesión asociado al usuario y el modo que escogió.
f. **Consulta del historial:** Se recupera el resumen de la sesión anterior de este modo.
g. **Presentación del resumen inicial:** Se muestra el resumen como parte del saludo del chatbot al comienzo de cada sesión.

Flujo de cierre de sesión:

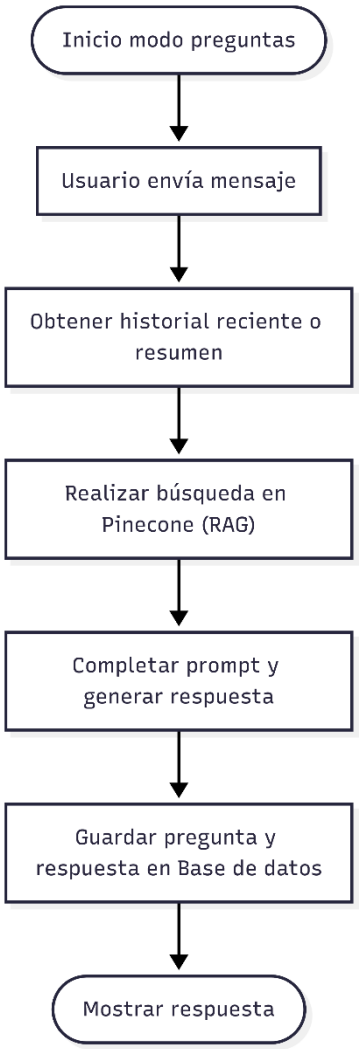
a. **Cierre de la sesión actual:** El usuario decide terminar la sesión actual en cualquiera de los dos modos.
b. **Recopilación de interacciones:** Usando la tabla de historial de mensajes se recopilan todos los mensajes que corresponda al ID único de sesión, además del último resumen de sesión.
c. **Completado de prompt:** Se combinan los mensajes recopilados y el resumen junto a un grupo de instrucciones para la generación correcta de un nuevo resumen de sesión.
d. **Envío y recepción del resumen:** Se envía el prompt o plantilla completa a la API de GPT4o, para luego recibir el resumen que se usará para una próxima sesión.

Estos flujos permiten dar continuidad a los procesos de aprendizaje por aparte, reconociendo el avance previo y personalizando la interacción desde el inicio.

3.9.3 Flujo modo Preguntas (RAG Convencional)

Figura 20

Flujo modo preguntas



Este diagrama de flujo se encuentra disponible en el repositorio del proyecto (ver **Apéndice A**).

Este flujo muestra toda la secuencia completa de pasos que se ejecutan cuando un usuario realiza una pregunta hasta que el sistema muestra una respuesta, esta implementación puede considerarse una versión de RAG simple, como el usado en la herramienta GPTE3T, pero mejorada por un sistema de memoria por usuario. Este sigue los siguientes pasos:

a. **Ingreso de pregunta:** El usuario escribe o dice en voz alta su pregunta. Si lo hizo mediante voz el sistema la transcribe usando el servicio de voz a texto (Speech-to-text) de Google Cloud.
b. **Obtención de historial o resumen:** Se recupera de la base de datos las últimas 3 interacciones entre el chatbot y el usuario. Si no existen, se usa el último resumen de sesión como historial.
c. **Conversión a vector y búsqueda:** La pregunta es convertida a una representación vectorial, lo que permite realizar la búsqueda de similitud en la base de datos en Pinecone, esto devuelve los 3 vectores más cercanos a la pregunta, los cuales contienen 3.000 caracteres que será usados como contexto.
d. **Construcción del prompt:** Se completa la plantilla (figura 10) que incluye:
    - Contexto: 3 fragmentos recuperados de pinecone.
    - Historial: 3 últimas interacciones entre el usuario y bot. O un resumen si no existen estas.
    - Nombre y edad.
    - La pregunta.

Luego el prompt o plantilla completa se envía al modelo GPT4o para que este de su respuesta.

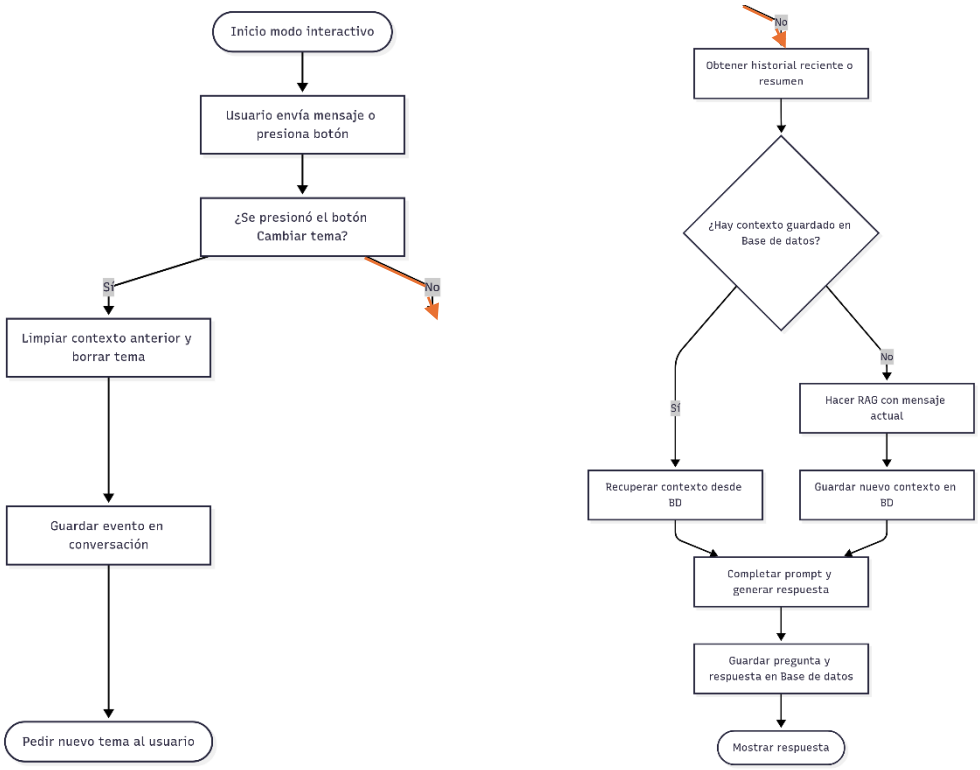
e. **Registro de interacción:** Pregunta y respuesta se almacenan junto a la información de la sesión y usuario, adicionalmente, si se tiene activa la opción de texto a voz (Voice-to-Speech) se envía la respuesta al servicio de Google Cloud para recibir un archivo de audio.

f. **Entrega y síntesis de la respuesta:** La respuesta se muestra en pantalla y, si está activa la opción, se reproduce el audio generado de esta.

3.9.4 Flujo modo interactivo (RAG modificado)

Figura 21

Flujo modo interactivo



Este diagrama de flujo se encuentra disponible en el repositorio del proyecto (ver **Apéndice A**).

El modo interactivo representa una versión modificada del enfoque RAG, pensada para mejorar toda la experiencia del usuario al momento de realizar el aprendizaje progresivo de un

tema específico, la diferencia principal del modo preguntas es que el contexto que se busca en la base de datos vectorial, el cual es 3 veces más grande (10 fragmentos de texto en vez de 3), se mantiene constante durante toda una conversación, y solo se actualiza si el usuario decide cambiar de tema. Los pasos que sigue son:

a. **Inicio de sesión y selección del modo:** El usuario accede al sistema y elige el modo interactivo desde la interfaz principal.
b. **Entrada del usuario o cambio de tema:** El usuario envía un mensaje o presiona el botón "Cambiar tema".
c. Si se presiona el botón:
    - Se elimina el contexto anterior guardado en la base de datos.
    - Se borra el tema interactivo actual de la sesión.
    - Se registra el evento y el sistema responde solicitando un nuevo tema.
d. Gestión del contexto:
    - Si ya hay un contexto guardado, este se reutiliza.
    - Si no lo hay, el sistema ejecuta RAG sobre la base de Pinecone con el vector del mensaje del usuario y retorna 10 vectores, lo cuales conforman el contexto que es almacenado para toda la sesión.
e. **Recuperación del historial:** Se obtiene el historial reciente, las últimas 6 interacciones, y si estas no existen se usa el último resumen de sesión.
f. **Construcción del prompt guiado:** Se completa la plantilla (figura 11) que incluye:
    - Contexto: 10 fragmentos recuperados de pinecone.
    - Historial: 6 últimas interacciones entre el usuario y chatbot, o un resumen si no existen estas.

- Nombre y edad.
- La pregunta.

Luego el prompt o plantilla completa se envía al modelo GPT4o para que este de su respuesta.

g. **Registro de la conversación:** Cada bloque de interacción se guarda junto con el contexto para mantener la coherencia temática en el resto de la sesión.
h. **Resumen final:** Al finalizar la sesión, se genera automáticamente un resumen del proceso de aprendizaje, útil tanto para el seguimiento del usuario como para su reactivación en futuras sesiones.

3.10 Pruebas realizadas y resultados obtenidos

Con el objetivo de cumplir con la etapa de validación, se llevaron a cabo pruebas piloto orientadas a evaluar la funcionalidad, el diseño y la experiencia de interacción del sistema con usuarios, estas pruebas permitieron obtener retroalimentación directa de los usuarios, la cual se utilizó para realizar ajustes en la interfaz, optimizar los modos de interacción y perfeccionar la respuesta del chatbot, garantizando así un desempeño adecuado en contextos educativos reales.

3.10.1 Pruebas funcionales

Las pruebas funcionales se centraron en verificar el comportamiento de cada uno de los módulos del sistema, de acuerdo con la arquitectura propuesta. Para ello, se definieron los siguientes criterios de validación:

- **Inicio y cierre de sesión:** Se verificó que el sistema permitiera el ingreso con credenciales válidas, el registro de nuevos usuarios, la asignación de identificadores únicos y la recuperación del historial previo.

- **Selección y funcionamiento de modos:** Se comprobó que el usuario pudiera seleccionar correctamente entre el modo pregunta/respuesta y el modo interactivo, y que ambos respondieran según sus respectivos flujos.
- **Interacción por voz:** Se verificó la transcripción de entradas orales mediante el servicio de Speech-to-Text (STT), así como la correcta generación de respuestas audibles mediante Text-to-Speech (TTS), empleando voces naturales y amigables para el público infantil.
- **Recuperación semántica:** Se validó que, frente a consultas diversas, el sistema ejecutaba búsquedas eficientes en la base de datos vectorial, seleccionando los fragmentos más relevantes y coherentes con el contexto temático, garantizando respuestas fundamentadas en el contenido cargado.
- **Generación de respuestas:** Se evaluó que las respuestas generadas por el modelo de lenguaje fueran coherentes, contextualizadas con el contenido recuperado, comprensibles y adaptadas al nivel de edad de cada usuario, manteniendo un tono amigable y pedagógico.
- **Gestión de sesiones:** Se verificó que cada sesión guardara correctamente el historial de conversación, y que el sistema generara resúmenes automáticos que podían ser consultados posteriormente.

Los resultados de estas pruebas mostraron un comportamiento estable y consistente del sistema en todos los componentes evaluados.

3.10.2 Pruebas de usabilidad

Para validar la experiencia de uso de BOTE3T, se realizaron sesiones piloto con personas que simularon tener diferentes edades de niños, bajo la supervisión de los desarrolladores del proyecto. Esta estrategia metodológica fue adoptaba debido a las dificultades logísticas para acceder a un número suficiente de participantes infantiles y todas las consideraciones logísticas y

de autorización que requieren estas actividades. Para esto hizo una encuesta en Google forms, estructurada donde a los usuarios se les entregaba un usuario con una edad específica, para que pudiera modificar su perspectiva al momento de responder la encuesta. La estructura de esta encuesta fue:

a. **Perfil del evaluador y condiciones:**
    - ¿Qué edad simulaste al interactuar con el chatbot?
    - Nivel educativo o profesión.
b. **Evaluación del funcionamiento general:**
    - ¿Qué modo utilizaste durante la prueba?
    - ¿Consideras que el chatbot adapta bien su lenguaje y estilo a la edad simulada?
    - ¿Qué tan fácil crees que sería para un niño de esa edad usar la interfaz?
c. **Interfaz y diseño:**
    - ¿Cómo calificarías la combinación de colores, tipografía e iconos en la interfaz gráfica de BOTE3T?
d. **Evaluación utilidad modo interactivo:**
    - ¿El modo interactivo logró mantener tu atención como niño simulado?
    - ¿Crees que las actividades del modo interactivo (juegos, retos) son adecuadas para niños?
e. **Accesibilidad:**
    - ¿Probaste la entrada por voz (hablar al chatbot)?
    - ¿Probaste la salida por voz (el chatbot hablándote)?
    - ¿Qué tan natural consideras la voz generada por el sistema?

f. Valor pedagógico:

- ¿Consideras que el chatbot logra despertar curiosidad científica?
- ¿Crees que un niño podría aprender contenido significativo con esta herramienta sin ayuda adulta?
- ¿Recomendarías esta herramienta como apoyo en clases de ciencia?

g. Retroalimentación y sugerencias:

- ¿Qué mejorarías o cambiarías?

A partir de la encuesta se obtuvieron los siguientes resultados agrupados por cada dimensión evaluada:

- **Perfil del evaluador y condiciones:** Los participantes contaban con distintos niveles educativos, desde bachillerato hasta formación profesional. Esta diversidad permitió obtener valoraciones tanto técnicas como pedagógicas.
- **Evaluación del funcionamiento general:** El 100% de encuestados consideró que el adapta de manera correcta su lenguaje y estilo a la edad del usuario. Asimismo, todos consideraron que un niño de su edad simulada podría usar la interfaz sin dificultad. Todos los usuarios interactuaron con ambos modos de funcionamiento, así validando el correcto funcionamiento del sistema.
- **Interfaz y diseño:** El 90% de los evaluadores calificó la interfaz (sus colores, tipografía e íconos) como adecuada o muy adecuada para un público infantil.
- **Evaluación utilidad modo interactivo:** El modo interactivo fue percibido como eficaz para captar y retener la atención, especialmente las etapas de juego y reto. Las actividades (juegos, adivinanzas, preguntas) fueron valoradas como apropiadas.

- **Accesibilidad:** El 60% por ciento de los participantes probó las funciones de entrada o salida por voz. Quienes usaron estas funciones calificaron la voz como “adecuada para niños” y “muy natural”.
- **Valor pedagógico:** El 90% de los evaluadores afirmó que el chatbot despierta curiosidad científica, lo cual representa el logro central del proyecto. Además, un 85% consideró que un niño podría aprender contenido usando la herramienta sin necesidad de asistencia de un adulto. Finalmente, el 100% de los encuestado recomendó el uso de BOTE3T como recurso educativo complementario en clases de ciencias.
- **Retroalimentación y sugerencias:** En la sección abierta la mayoría de los participantes expresó satisfacción general con el sistema. Sin embargo, se recogieron observaciones útiles. Una sugerencia se enfocó en el modo interactivo, proponiendo que las respuestas fueran más cortas para mantener la atención de niños. Otro participante propuso mejorar el sistema de transcripción ya que este no devuelve texto con signos de puntuación.

En el **Apéndice B** se presentan los resultados de las pruebas realizadas a la herramienta BOTE3T.

4. Conclusiones

El modelo de lenguaje GPT, consumido a través de la API de OpenAI, fue configurado con prompts especializados que incluían instrucciones pedagógicas, control de tono, restricciones lingüísticas por edad y dinámicas interactivas, los resultados de las pruebas demostraron que esta parametrización fue efectiva para producir respuestas coherentes, apropiadas al nivel cognitivo del usuario.

El sistema de gestión de sesiones implementado permitió mantener persistencia de información relevante entre sesiones, incluyendo la identificación de usuario, historial conversacional y resúmenes automáticos generados por el modelo, este diseño no solo mejoró la continuidad pedagógica, sino que facilitó la trazabilidad para posibles integraciones futuras con plataformas de seguimiento educativo.

Una lección clave aprendida fue la importancia de simplificar los procesos técnicos de mantenimiento y actualización del sistema, lo que finalmente abre la puerta al uso de herramientas basadas en inteligencia artificial, especialmente en contextos donde los docentes no poseen la información técnica especializada.

Desde una perspectiva de ingeniería, el proyecto demostró la viabilidad técnica de una herramienta conversacional basada en inteligencia artificial, que puede convertirse en una herramienta pedagógica poderosa, especialmente si se estructura desde una perspectiva educativa, se adapta al contexto del usuario y se combina con principios de accesibilidad y personalización, la propuesta desarrollada contribuye no solo a la apropiación social del conocimiento, sino también a la accesibilidad de la ciencia y la tecnología.

La incorporación del Modo Pregunta y Modo Interactivo en BOTE3T permitió diseñar una experiencia educativa flexible y adaptada a diferentes necesidades de los usuarios, el Modo Pregunta ofrece respuestas rápidas, claras, optimizando la resolución de dudas puntuales, por su parte, el Modo Interactivo combina explicaciones estructuradas con minijuegos, retos y evaluaciones, promoviendo un aprendizaje activo y lúdico, esta dualidad de modos fortaleció tanto la personalización de la enseñanza como la motivación de los usuarios, permitiendo que la herramienta funcione como un recurso dinámico y versátil dentro del proceso educativo.

El uso de tecnologías de voz en BOTE3T mejoró la experiencia del usuario al integrar entrada por voz y respuestas auditivas, para una interacción más inclusiva y haciendo el sistema más atractivo para los niños.

5. Recomendaciones

A partir de los resultados obtenidos durante el desarrollo e implementación del chatbot, se formulan las siguientes recomendaciones orientadas a mejorar su desempeño, ampliar su impacto y alcance:

- Ampliar la base de contenidos: Se sugiere incorporar gradualmente nuevos documentos temáticos, validados por expertos, que permitan cubrir más tópicos de radioastronomía u otras áreas científicas afines, esto fortalecerá la capacidad de respuesta del sistema y diversificará las rutas de aprendizaje.
- Explorar la integración de generación de imágenes: Se recomienda evaluar la incorporación de modelos de inteligencia artificial capaces de generar imágenes a partir del contenido textual tratado durante la conversación, esto permitiría complementar las explicaciones con recursos visuales adaptados al contexto educativo, facilitando la comprensión de conceptos abstractos en temas como radioastronomía, la generación de ilustraciones, esquemas o representaciones animadas podría enriquecer significativamente la experiencia de aprendizaje, especialmente en el público infantil.
- Fortalecer la estructura pedagógica del modo de aprendizaje interactivo: Se recomienda enriquecer las etapas del modo interactivo con una mayor diversidad de actividades interactivas y mecanismos adaptativos, ejercicios de refuerzo ante errores frecuentes. se sugiere vincular los contenidos a referentes curriculares oficiales, para facilitar su integración en procesos educativos formales.
- Ampliar la cobertura temática del sistema: Se sugiere adaptar la herramienta para abarcar otras áreas del conocimiento, como ciencias naturales, matemáticas básicas, entre otras,

aprovechando su arquitectura modular y capacidad de aprendizaje a partir de documentos, esta expansión permitiría que el sistema funcione como una plataforma versátil de apoyo escolar, manteniendo el enfoque lúdico y personalizado que lo caracteriza, la incorporación de nuevas materias podría realizarse mediante la carga de contenidos validados y el diseño de nuevos flujos pedagógicos interactivos.

- Desarrollar una aplicación móvil: Se recomienda crear una aplicación móvil que permita el acceso a la herramienta desde teléfonos o tabletas, esto facilitaría su uso en contextos rurales o educativos donde el acceso a computadores es limitado, y permitiría aprovechar funcionalidades propias de los dispositivos móviles, como notificaciones, cámara o almacenamiento local; una app móvil también contribuiría a mejorar la portabilidad, la escalabilidad del sistema y su adopción por parte de comunidades escolares más amplias.

Referencias Bibliográficas

Briggs, J., & Ingham, F. (s. f.). *LangChain AI Handbook*. Pinecone. https://www.pinecone.io/learn/series/langchain/

Chase, H. (2022). *LangChain* [Framework de software]. Lanzamiento en octubre de 2022. LangChain. https://langchain.com

Contreras Martínez, M. E., & Olguín Ruiz, L. (2024). Radioastronomía, el universo visto con otros ojos. *Epistemus*, 18 (36), 1-27

Fraknoi, A., Morrison, D., & Wolff, S. C. (2022). *Astronomy 2e* [PDF]. OpenStax. https://openstax.org/books/astronomy-2e/pages/1-introduction

Google Cloud. (s. f.). *Conceptos básicos de Text-to-Speech*. Google. https://cloud.google.com/text-to-speech/docs/basics?hl=es-419

Google Cloud. (s. f.). Solicitudes de Speech-to-Text. Google. https://cloud.google.com/speech-to-text/docs/speech-to-text-requests?hl=es-419

Google. (2024). *Apps Script Web Apps*. https://developers.google.com/apps-script/guides/web

Google. (2024). *How to get a RAG application to add citations.* LangChain. https://python.langchain.com/docs/how_to/qa_citations/

LangChain. (s. f.). *LangChain documentation*. https://www.langchain.com/

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. *ArXiv*. https://doi.org/10.48550/arXiv.2005.11401

Mendoza Torres, J. E. (2010). *Introducción a la astronomía y a la astrofísica [Material para la Olimpiada Nacional de Astronomía en México]*. Instituto Nacional de Astrofísica, Óptica

y Electrónica. https://astro.inaoep.mx/archivos2020s/Directorio/Investigadores/LibroAstronomia2010.pdf

MINTIC Colombia. (2022). Colombia Adopta de Forma Temprana recomendaciones de ética en inteligencia artificial de la UNESCO para la Región. Ministerio de las Tecnologías de la Información y las Comunicaciones. https://mintic.gov.co/portal/inicio/Sala-de-prensa/Noticias/208109:Colombia-adopta-de-f

Moez Ali. (2023). *Mastering Vector Databases with Pinecone Tutorial: A Comprehensive Guide*. DataCamp. https://www.datacamp.com/tutorial/mastering-vector-databases-with-pinecone-tutorial

OpenAI. (2024). *OpenAI API documentation*. https://platform.openai.com/docs/

Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura UNESCO. (1970, Enero 1). *Ética de la inteligencia artificial*. UNESCO. https://www.unesco.org/es/artificial-intelligence/recommendation-ethics

Pinecone Systems. (s. f.). *Pinecone: Vector database for AI applications*. https://www.pinecone.io/

PythonAnywhere LLP. (s. f.). *PythonAnywhere: Python in the cloud*. https://www.pythonanywhere.com/

Rueda Rodríguez, M., & Hernández Prince, C. A. (2024). *Diseño e implementación de un bot de charla basado en GPT para la acreditación internacional de los programas de pregrado de la E3T* (Trabajo de grado, Universidad Industrial de Santander). Repositorio Institucional UIS. https://noesis.uis.edu.co/handle/20.500.14071/15822

Soriano, P. (2022, Marzo 26). HTML, *CSS y JavaScript. Lenguajes para el Desarrollo de Páginas Web*. https://geoinnova.org/blog-territorio/html-css-y-javascript-lenguajes-para-el-desarrollo-de-paginas-web/

Telescope Guide. (2021). *Astronomy for Kids: The Journey Begins*. Telescope Guide. https://www.telescopeguide.org/wp-content/uploads/2021/12/Astronomy-for-Kids-TelescopeGuide-org.pdf