

**Proyecto integrador MNA Grupo10.**

**Avance 3. Baseline.**

**Equipo #16:**

**Kevin Brandon Cruz Mejía, Matricula A01794176.**

**José Alberto Almada Gerardo, Matricula A01794415.**

13 de octubre del 2024.

Índice.

Introducción .............................................................................................. 3

Baseline...................................................................................................... 3

Evaluacion ................................................................................................ 6

Conclusiones............................................................................................. 6

Bibliografía ................................................................................................ 7

**Introducción.**

Este avance debe culminar en un modelo baseline que actúe como referencia, proporcionando un punto de partida para evaluar la viabilidad del problema. Debido la naturaleza de nuestro proyecto integrador el cual es el de desarrollar un show room digital donde se pueda analizar el uso de Asistentes AI en los diferentes sectores y temas industriales con el fin de identificar áreas de oportunidad en cada industria donde sea factible desarrollar un asistente basado agentesAI con arquitectura RAG, estos asistentes podrán incorporar conocimiento personalizado, integrar funciones de llamada, y ser accesible a través de una API desde código en Python.

Debido a que muchos de los casos siguen en construcción, no podremos definir aun todas las características, los posibles problemas de ajuste y la métrica de evaluación adecuada, pero a continuación abordaremos el código baseline que se usara en nuestra investigación.

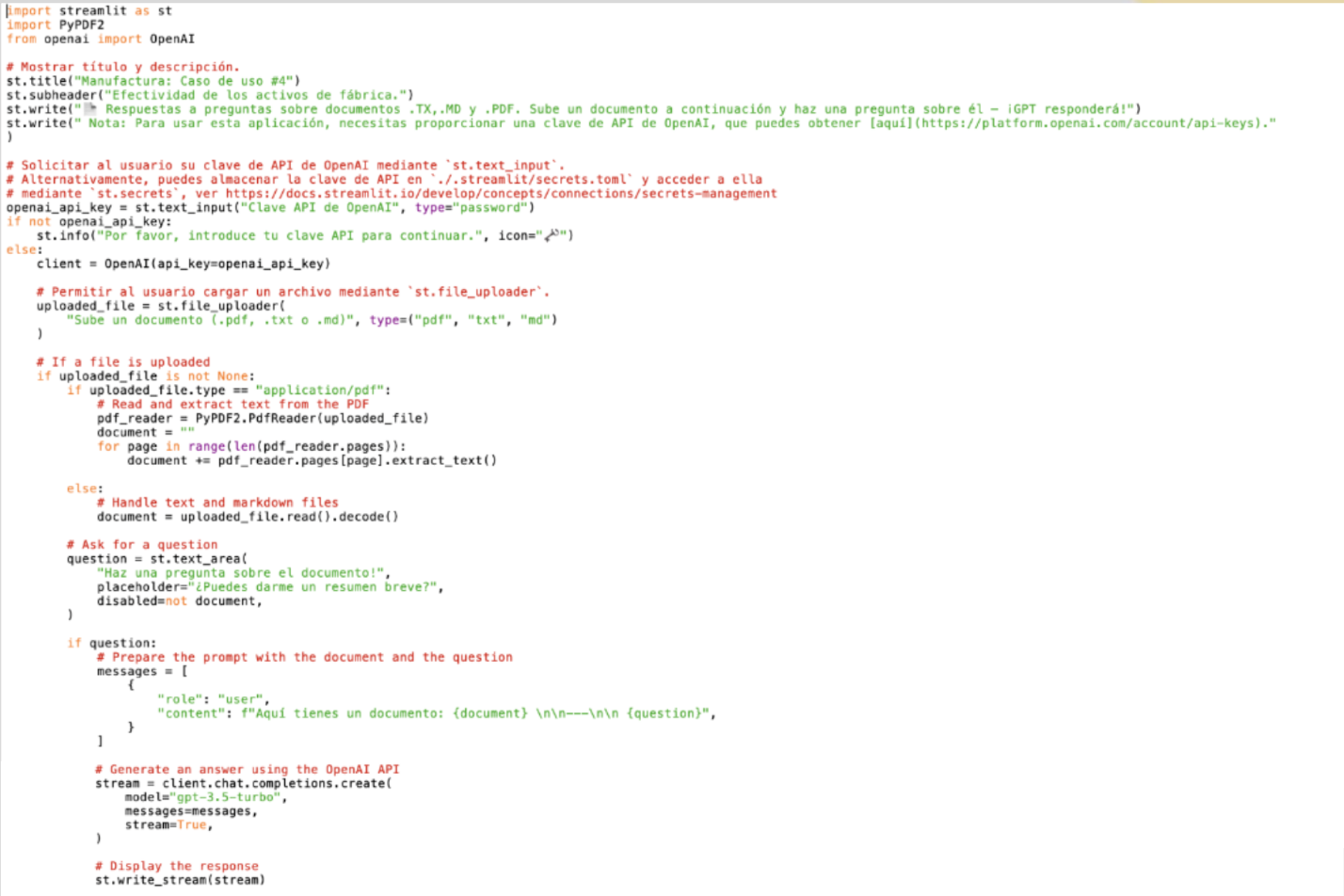
**Punto de partida (Baseline).**

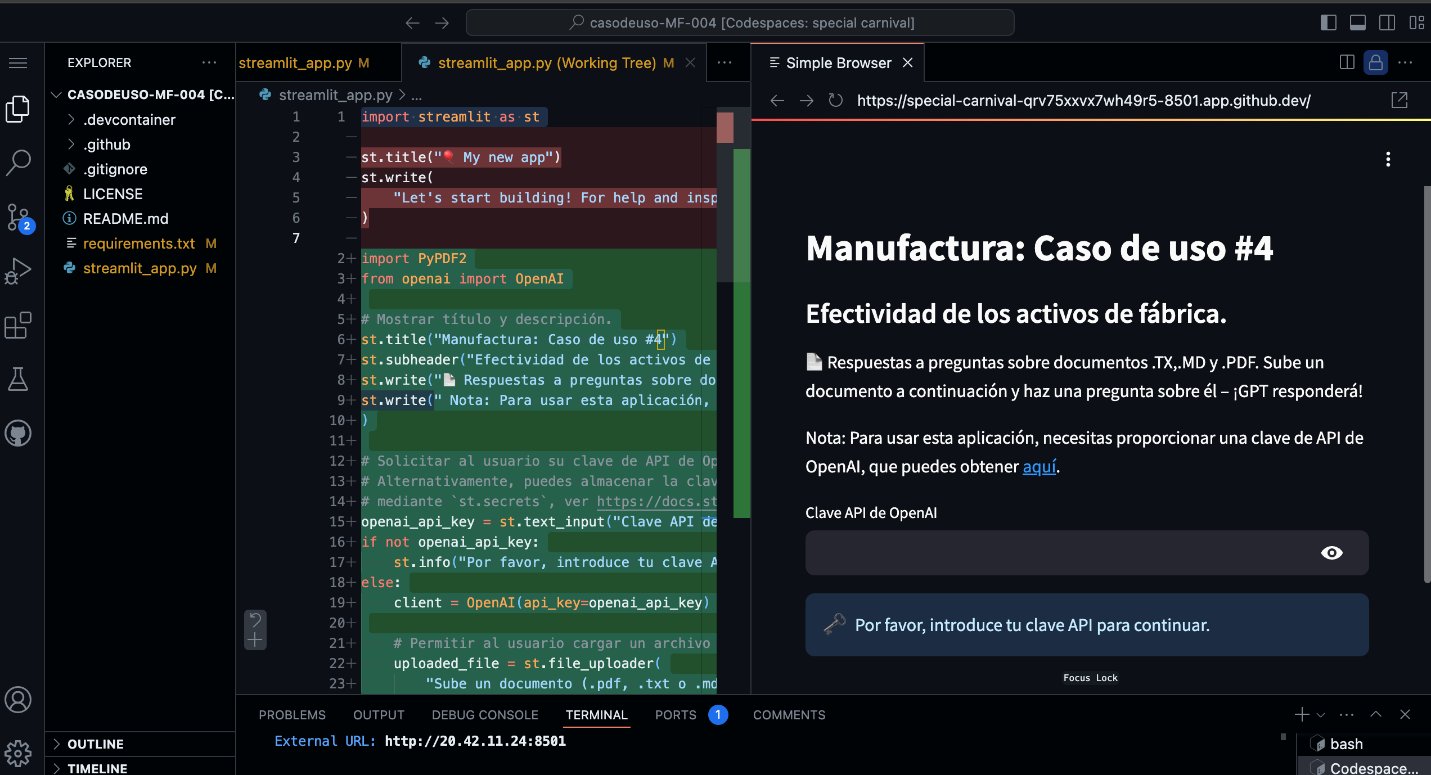
Para crear un baseline de un Asistente AI basado en API de OpenAI y ejecutado en Streamlit con arquitectura RAG (Retrieval-Augmented Generation), el objetivo del baseline es evaluar un Asistente AI sencillo, que utilice la API de OpenAI para responder preguntas a partir de un conjunto de datos o documentos. Este baseline servirá para medir el rendimiento de un modelo básico, y proporcionar una comprensión inicial antes de implementar soluciones más complejas.

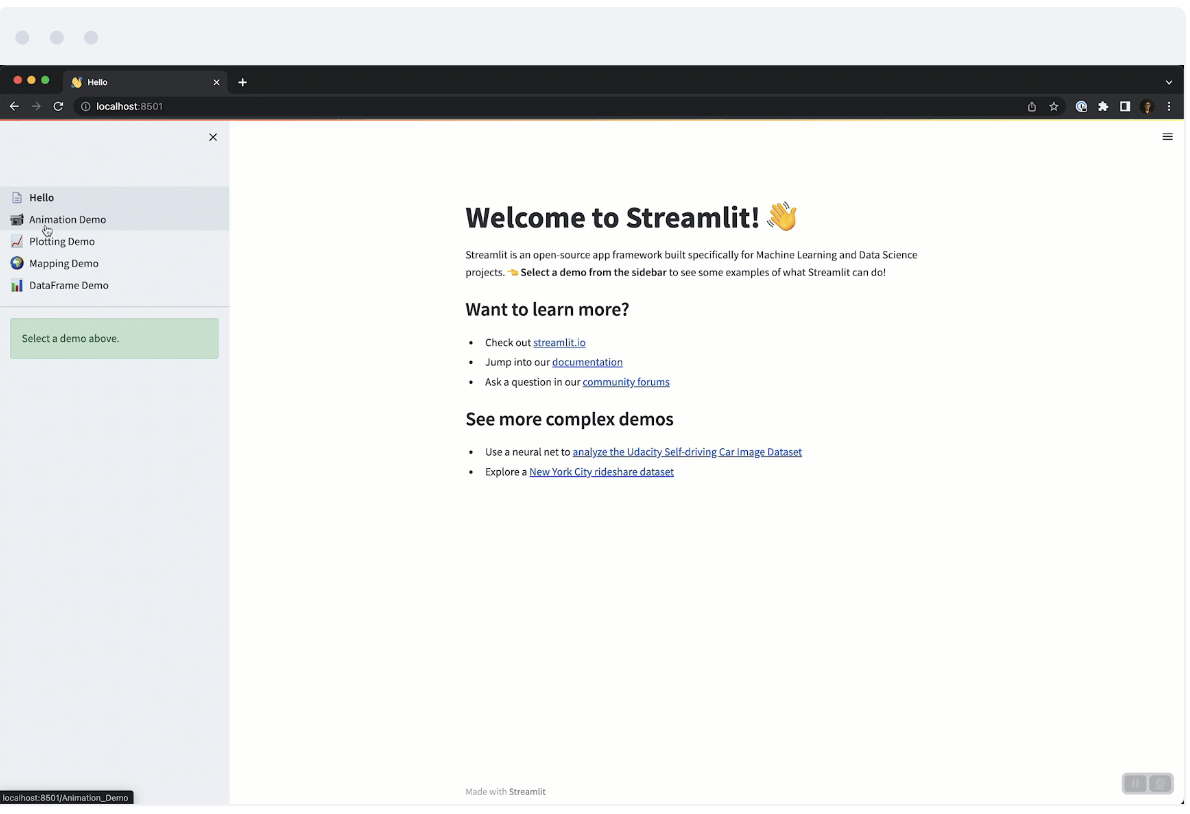
Para un Asistente AI de baseline que use RAG, se planea emplear un modelo de lenguaje pre-entrenado (GPT-3.5-turbo) como el generador, y utilizar búsqueda basada en palabras clave o similitud de texto para la fase de recuperación.

Los pasos para seguir son los siguientes.

* Modelo. Utiliza un modelo pre-entrenado de OpenAI para la generación de respuestas.
* Algoritmo. GPT-3.5-turbo OpenAI assistant AP I.
* Simulación de Recuperación. Usar funciones de búsqueda de texto en un conjunto de documentos local para simular la fase de recuperación.
* Librería para recuperación. Se planea utilizar emplear TfidfVectorizer de sklearn para implementar una búsqueda básica de documentos, pero el entrenamiento RAG de cada asistente se plane realizarlo de manera individual en la aplicación de OPENAI.
* Importancia de Características. Dado que el modelo es un generador de lenguaje pre-entrenado, no se podrá calcular directamente la importancia de las características, pero se podrá evaluar qué tan relevantes son las respuestas generadas con respecto a las consultas y los documentos.
* Ajuste del Modelo. Se evaluará si el modelo está "sobre ajustado" o "sub ajustado" revisando la coherencia de las respuestas en diferentes preguntas y documentos.
* Métrica de Evaluación. Se considera utilizar Exact Match (qué tanto la respuesta generada coincide con la información relevante) o BLEU score para medir la similitud de las respuestas con respecto a un conjunto de referencias.
* Desempeño Mínimo. El desempeño mínimo esperado es que el asistente pueda recuperar información relevante del documento con una precisión básica, proporcionando respuestas coherentes.

Figura1.1. Código baseline que se está tomando como referencia actual mente.

Figura1.2. Código baseline para streamlit.

Figura1.3. Modelo multi página a utilizar en nuestra aplicación de show room.

**Evaluación.**

Importancia de características. No es aplicable en este baseline, pero se puede evaluar la relevancia de las respuestas.

* Ajuste del modelo. Analiza la coherencia y relevancia de las respuestas.
* Métricas. Evaluar con Exact Match o BLEU Score.
* Desempeño mínimo. Debe ser capaz de dar respuestas coherentes y útiles.

**Conclusiones.**

En conclusión, este baseline ofrece una implementación sencilla para evaluar la viabilidad del asistente AI antes de avanzar a modelos más complejos.

Bibliografía.

Assistants API Overview. (n.d.). Retrieve from: <https://platform.openai.com/docs/assistants/overview>

Nantasenamat, C. (2023, July 20). Beginner’s guide to OpenAI API: Build your own LLM tool from scratch. Retrieve from: <https://blog.streamlit.io/beginners-guide-to-openai-api/>

Donato, V. (2022, June 2). Introducing multipage apps! Vincent Donato. Retrieve from: <https://blog.streamlit.io/introducing-multipage-apps/>

Build a basic LLM chat app. (n.d.). Retrieved from: <https://docs.streamlit.io/develop/tutorials/llms/build-conversational-apps>