

# DOC-BOT

## (Bot de pediatría y crianza)

06/11/2024

Emanuel A. Teper

Buenos Aires, Argentina.

## Introducción

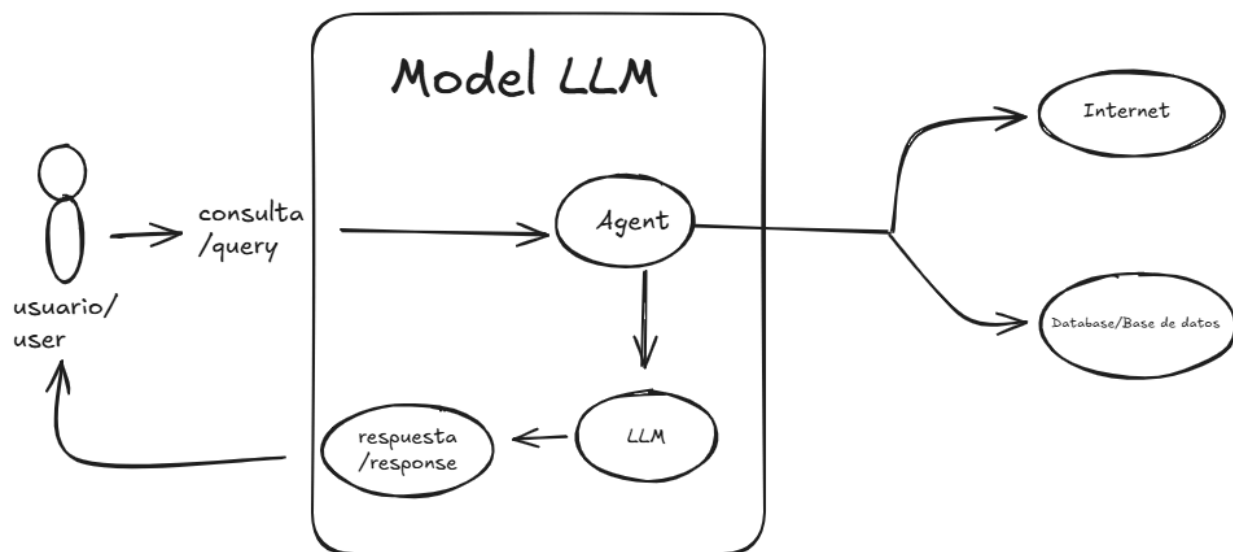
**Descripción general:** Este bot está diseñado para proporcionar consejos de salud pediátrica, responder preguntas frecuentes sobre enfermedades comunes, guiar a los padres, madres y cuidadores de niños sobre cuidados básicos y derivar casos graves a profesionales médicos.

**Objetivo:** Mejorar el acceso a información médica confiable para padres, madres y cuidadores de niños.

**Público objetivo:** Madres, padres, cuidadores, tutores y personal de la salud

## Arquitectura

1. **Interfaz de usuario (UI):** Contiene una interfaz de usuario simple. Un chat con preguntas y respuestas bien definidas. Botones de acceso rápido a consultas y un botón para borrar el historial.
2. **Backend:** En esta sección se encuentran 3 archivos un main.py que se encarga de reunir todos los módulos del backend y aplicar la toma de decisiones y generar las respuestas. Un archivo embeddings que se encarga de gestionar el **proceso de creación y uso de embeddings** (representaciones vectoriales) de texto, que son fundamentales para el funcionamiento del bot. Y un archivo de web scraping que se encarga de adquirir información de una web en específica para mantener actualizado siempre al bot
3. **Bot de IA:** El motor de inteligencia artificial es GPT-3.5-Turbo.



## Interacción del usuario con el bot


**Flujo de conversación:** Muestra un diagrama de flujo o ejemplos de cómo el bot interactúa con los usuarios. Por ejemplo, si un usuario pregunta "¿cómo saber si mi hijo tiene fiebre?", el bot podría responder con una serie de preguntas relacionadas y luego proporcionar consejos.

**Sugerencias de preguntas frecuentes:** Incluir ejemplos de preguntas comunes que los usuarios pueden hacer al bot, como:

- "¿Qué hacer si mi hijo tiene tos?"
- "¿Cómo saber si mi bebé tiene cólicos?"
- "¿Cuál es la dosis correcta de paracetamol para un niño?"

## Instalación y requisitos

- I. Abrir terminal y ejecutar: `python -m venv env`
- II. Activar el env: `env/scripts/activate`
- III. Si nos da un error por falta de autorización ejecutamos como administrador el siguiente comando en el PowerShell de Windows:  
`Set-ExecutionPolicy RemoteSigned -Scope LocalMachine`  
 luego apretamos "s" para confirmar, luego repetimos el paso 2

- 
- IV. Instalar dependencias con el comando: `pip install -r requirements.txt`
  - V. Ejecutar el `main.py` para crear la carpeta VECTORS

Para exportar un `requirements.txt` se hace de la siguiente manera: `pip freeze > requirements.txt`

Recuerden poner su Api Key en el archivo `.env`.

**!!!Muchas gracias por usar DOC-BOT!!!**