fanlabs.streamlit.app

**Everytime I open Terminal:**

cd ~/Desktop/fanlabs-gpt

source venv/bin/activate

[**https://fanlabs.streamlit.app/**](https://fanlabs.streamlit.app/)

fanlabs.streamlit.app

**Everytime I open Terminal:**

cd ~/Desktop/fanlabs-gpt

source venv/bin/activate

**For Adding Docs - run in terminal!**

cd ~/Desktop/fanlabs-gpt/fanlabs\_rag  
source venv/bin/activate  
source ~/.zshrc

**To open GPT**

streamlit run app.py

**New Push Token (newest 5/30 8:50am) – in use**

ghp\_rmlPMDYGbxml3xV9U5eeTdOrfrJoRo4Da8hD

---------

**Fan Labs Push Push API –** [**https://fanlabs.streamlit.app/**](https://fanlabs.streamlit.app/) **--> three dots bottom right –> settings 🡪 secrets**

sk-proj-TVScrpKDNUtrWQI53HVpjeHutx8LM\_awjuodk3Try7IAEgxAG-tYJBEAZPh63SXOr5u\_FGFyScT3BlbkFJTxvxcO\_DfzCqrVoCSiGKWAyjRV-0aALU1QWC3HF11788QbXLKW5\_NNyJkE-bpsUPpMbNW7eZoA

**✅ 1. How the System Works**

**GitHub**  
Stores all your app files:

* app.py (chat app)
* embed.py (document embedding)
* .faiss and .json index files

**Streamlit Cloud**  
Hosts your chatbot publicly at: <https://fanlabs.streamlit.app>  
It auto-updates when you push code or data changes to GitHub.

**OpenAI API**  
Provides:

* Embedding of your documents (via embed.py)
* Chat response generation (via app.py)

**You**  
Run embed.py when adding/changing documents. Push files to GitHub. Streamlit does the rest.

**✏️ 2. Update FanLabs GPT Prompt**

**Where?**

Inside app.py, look for:

prompt = (

"You are a helpful assistant. Answer the question based only on the following context:\n\n"

+ "\n\n---\n\n".join(retrieved\_chunks)

+ f"\n\nQuestion: {query}"

)

**How?**

1. Open app.py in a code editor
2. Edit the system prompt to change tone, rules, etc.
3. Save the file
4. Run:
5. git add app.py
6. git commit -m "Update prompt"

git push

1. ✅ Streamlit auto-refreshes

**📄 3. Add Documents (Step-by-Step)**

**🧹 A. Clean & Chunk Your Docs**

Each document must be broken into small text chunks. Use this structure:

[

{"text": "First chunk of your document."},

{"text": "Second chunk with meaningful content."}

]

* Save this as fanlabs\_chunks.json in your project folder
* You can write a script to clean and chunk long docs automatically

**🧠 B. Embed Your Document**

1. Activate your Python environment:

source venv/bin/activate

1. Run the embedder:

python embed.py

1. ✅ Confirm this appears:

✅ SentenceTransformer embeddings complete. Index and metadata saved.

**📤 C. Push to GitHub**

git add fanlabs\_vector\_index.faiss fanlabs\_chunk\_metadata.json

git commit -m "Add new doc"

git push

* This uploads your document index to GitHub
* Streamlit automatically updates the live chatbot

**🧪 4. Confirm It’s Working**

Ask a question related to your doc. ✅ If the answer is specific and relevant → it worked  
❌ If answer is vague or off-topic → the data wasn’t embedded or indexed properly

Check:

* fanlabs\_vector\_index.faiss exists and is pushed
* fanlabs\_chunk\_metadata.json exists and is pushed
* You ran embed.py successfully

**🔁 5. Notes & Maintenance**

* **No need to rerun or restart manually.** Streamlit refreshes with every push
* To edit chatbot logic, update app.py
* To retrain on new docs, update fanlabs\_chunks.json + rerun embed.py
* The .env file stays local; your **Streamlit secret** handles the live key