Setting Up My Own Private AI Chatbot

Why Bother Setting Up a Personal AI?

Privacy. That's the #1 reason.

When you use a cloud-based chatbot (like ChatGPT, Gemini, etc.), your data — and metadata — is stored on external servers. But if you run your AI **locally**, nothing leaves your system. It's **yours, fully private**.

This matters when you're:

- Working on confidential projects
- Collaborating on sensitive documents
- Building internal tools
- Or just want to **own your data**

Tools We'll Use

- **Ollama** – lightweight way to run LLMs locally
- **LLaMA 3** – the actual AI model
- **Open WebUI** – gives your AI chatbot a clean GUI (instead of just terminal use)
- **Docker** – helps run apps in isolated containers (no messy setup)

Why Use Docker?

Docker makes it **easy to run services** like Open WebUI without installing tons of dependencies or messing with your local system. You can start/stop it like flipping a switch all contained, no chaos.

Why LLMs Use GPUs

LLMs do heavy math — mainly matrix operations — which are perfect for **parallel processing**. That's where GPUs shine.

Performance Note:

- **Recommended:** 16GB VRAM GPU
- **Below 16GB VRAM?** It'll **still work**, but you'll notice some slowdowns.
- **No GPU?** It defaults to your **CPU**, but expect **slower generation** and lags in output.

The AI will still run — just not as fast or smooth.

Step-by-Step Setup

Step 1: Install Ollama

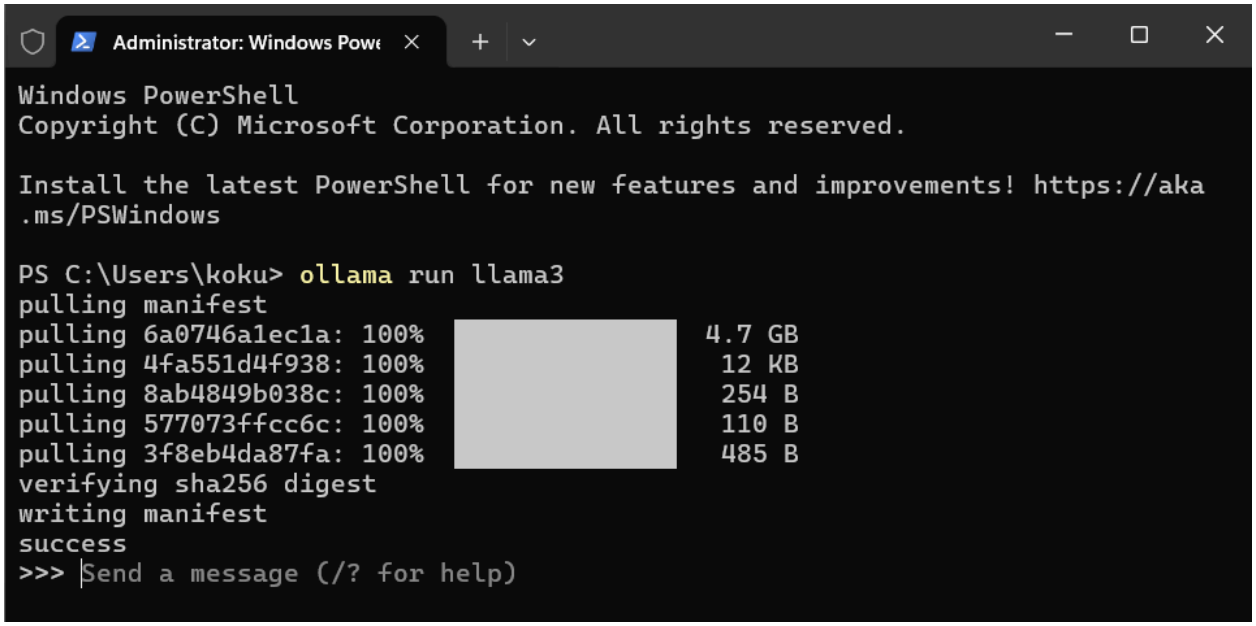
Download and Install Ollama

- *Visit 👉 https://ollama.ai*
- *Download and install Ollama based on your operating system.*

Once installed, open your terminal and run:

```
ollama run llama3
```

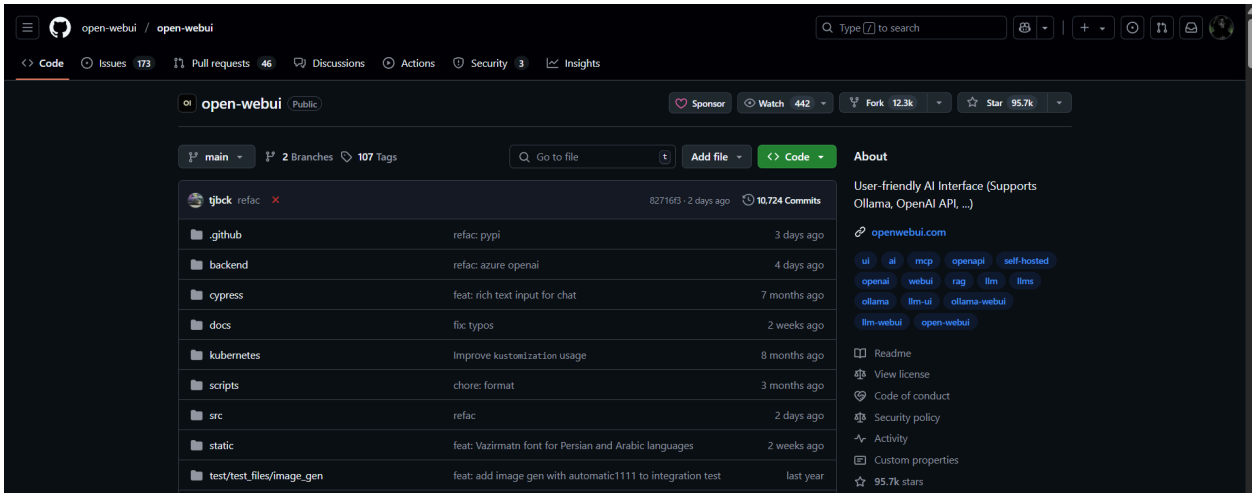
🔍 ***Why LLaMA 3?***
It's the latest open-source large language model (LLM), trained on massive datasets and optimized for local use. LLaMA 3.1 or 3.3 are great picks for better context understanding.



Step 2: Set Up Open WebUI

Running AI from the terminal isn't fun. Let's make it visual.

- Head over to 👉 https://openwebui.com
- Click **"Get OpenWebUI"** — it'll redirect to the official GitHub repo.
- OpenWebUI gives you a **chat-style interface** for interacting with your private AI.



Step 3: Install Docker

We use Docker to easily launch OpenWebUI with the right settings, libraries, and integrations — without breaking anything on our local system.

To get started:

- Go to 👉 https://docker.com and install Docker Desktop.
- During setup, **you might hit a WSL error** (common on Windows). Fix it via:

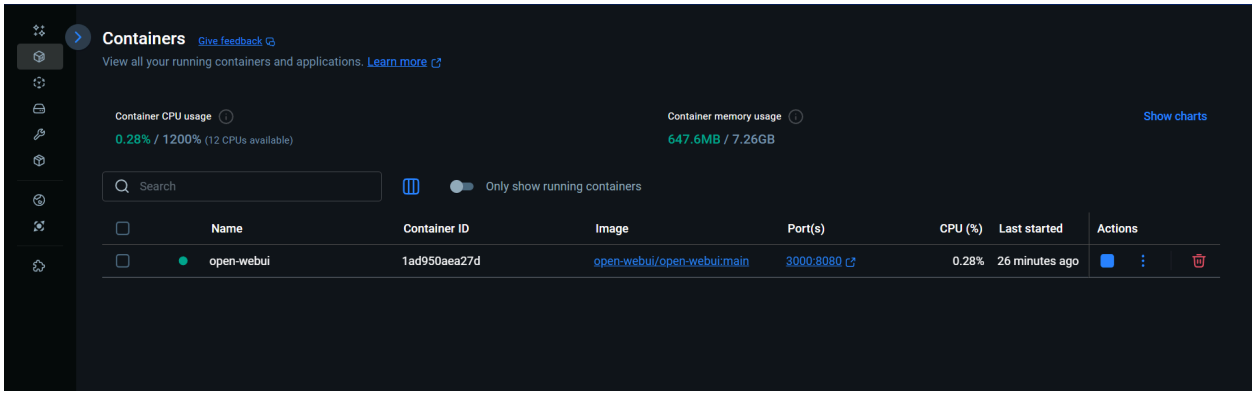
```
# If WSL isn't installed
     wsl --install

# If WSL is installed, just update it
     wsl --update
```

Once Docker is installed and your account is ready, move on to the next step.

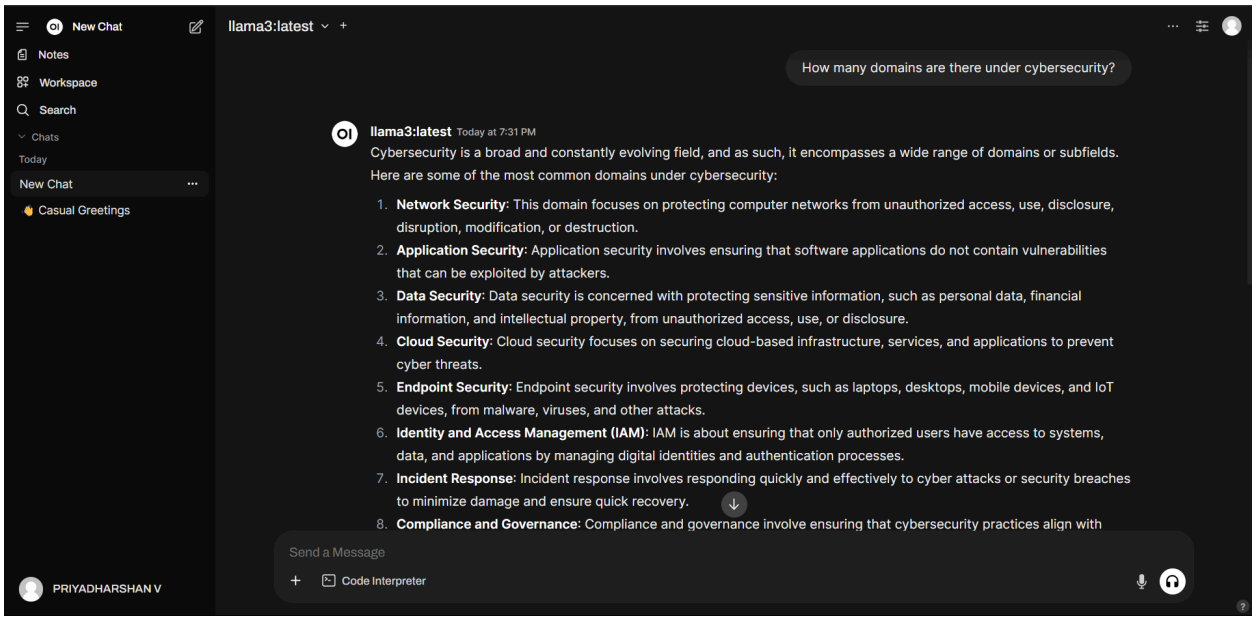
Step 4: Launch Your Private Chatbot

- *Go back to the **GitHub page** of OpenWebUI.*
- *Scroll to the **"Quick Start with Docker"** section in the README.*
- *Under "Installation with Default Configuration," copy the provided Docker command and paste it into your Docker terminal.*

Docker will create a container for the web UI and assign a port (e.g., `localhost:3000`).



*Paste that into your browser — your **private AI chatbot UI is live**.*



Final Tips

- Save your local AI link as a browser bookmark
- Try different LLMs via Ollama like `llama2`, `mistral`, `codellama` depending on your use case
- You can extend this setup with plugins or integrate it into your dev workflow

---

Why I Shared This

Most people don't realize they can **own and control their AI**.

This isn't just about privacy — it's about **independence**, **data control**, and using AI as a **power tool**, not just a toy.

Priyadharshan Vadivel

(Pentester | Red Team)