AI-04557 Open WebUI Information

Explain Open WebUI? Why people use it? Features, Chat Interface, KnowledgeBase? What hardware specs needed to run at my home dockerlab? What GCP specs needed to run? 4vcpu, 16GB, 80GB GCP enough. Say more.

Open WebUI is an open-source, feature-rich, self-hosted web interface designed to act as your personal "ChatGPT-style" dashboard.

Originally built as a frontend for **Ollama**, it has evolved into a complete, vendor-agnostic AI platform. It lets you connect local AI engines, cloud providers (OpenAI, Anthropic, Mistral, Groq), or open-source local API hosts (LM Studio, vLLM) under one interface.

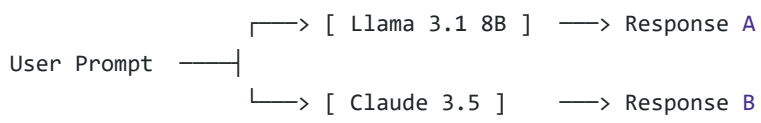
Why Do People Use It?

- **Privacy & Data Control:** Keep chat histories, user management, and internal documents strictly on local hardware or private servers without sending data to third parties.
- **No Subscription Lock-In:** Replace monthly SaaS AI subscriptions by leveraging open-weight models (like Llama 3, Qwen 2.5, DeepSeek) or cheap API keys.
- **Multi-Model Orchestration:** Combine different models in a single chat (e.g., using a fast local model for summarizing, but calling Claude 3.5 Sonnet or GPT-4o for complex coding).
- **Multi-User Access Control:** Run a single server instance with fine-grained Role-Based Access Control (RBAC) so friends, family, or lab team members get their own private accounts.

Key Features, Chat Interface & KnowledgeBase

1. Modern Chat Interface

- **ChatGPT-Like UI:** Familiar workspace featuring Markdown/LaTeX rendering, code highlighting, web searching capabilities, and prompt history.
- **Voice & Video Interactivity:** Hands-free voice mode leveraging local or cloud Speech-to-Text (Whisper) and Text-to-Speech engines.
- **Model Branching & Side-by-Side:** Split a prompt to evaluate responses from two different models simultaneously.



2. Built-in Knowledge Base (RAG)

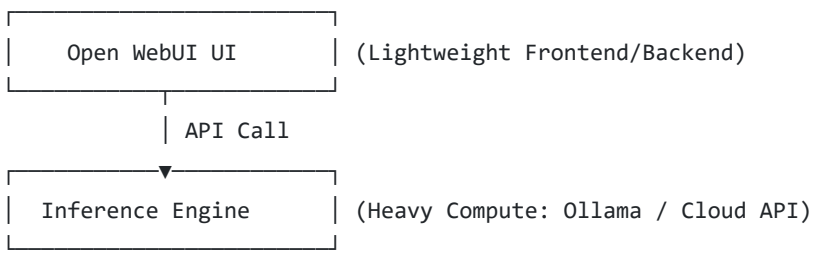
- **Document Processing:** Upload PDFs, TXT files, CSVs, or websites directly into the UI.
- **Vector Database Integration:** Automatically chunks texts, calculates embeddings (using built-in models or Ollama), and stores them natively.
- **Document Grounding:** Select specific `#KnowledgeBase` files during a chat to force the AI to answer exclusively using your personal documents.

3. Modding & Ecosystem (Pipes, Actions & MCP)

- **Tool Calling & Code Execution:** Connect tools like Model Context Protocol (MCP) servers or Python runtimes to let models run shell commands, browse the web, or read external databases.
- **Prompts & Agents Marketplace:** Import or create system prompts, personas, and custom workflows via the community hub.

Hardware Specs Needed

It is critical to distinguish between **Open WebUI itself** and the **AI Model Inference Engine** (like Ollama or vLLM). Open WebUI acts strictly as a lightweight web server/web backend.



1. Home DockerLab Specs

If you run Open WebUI in Docker alongside your models:

- **For Open WebUI Alone (Using External/Cloud APIs):**
  - **CPU:** 1-2 vCPUs
  - **RAM:** 1 GB–2 GB (idle footprint is ~500 MB)
  - **Disk:** ~10 GB (for Docker images, chat logs, vector database index)
- **For Open WebUI + Local AI Inference (Ollama on Home Hardware):**
  - **CPU:** 4+ Cores (AVX2 / AVX512 support recommended for CPU fallback)
  - **RAM:** 16 GB to 32 GB (to run quantized 8B-14B models on RAM)
  - **GPU (Recommended):** NVIDIA GPU with **8GB–16GB VRAM** (e.g., RTX 3060 12GB / RTX 4070 / RTX 4090) for real-time generation speed.
  - **Disk:** 50 GB+ NVMe SSD (model weights range from 4GB to 25GB each).

GCP Specs (Evaluating 4 vCPU / 16GB RAM / 80GB Disk)

> **Verdict:** A GCP VM with **4 vCPUs, 16 GB RAM, and 80 GB SSD is an EXCELLENT sweet spot** for host deployment—with a few structural caveats depending on how you plan to process inference.

Scenario A: Web UI + RAG + Cloud/External APIs

If this GCP instance hosts Open WebUI, handles document embedding/vector storage, and connects out to external APIs (OpenAI, Anthropic, Groq) or a dedicated GPU box in your home lab:

- **Performance: Flawless & Over-provisioned.**
- **Capacity:** Easily handles 20–50 concurrent users without slowing down.
- **Storage:** 80 GB is plenty for Docker containers, PostgreSQL databases, vector stores, and thousands of PDFs.

Scenario B: Web UI + Local CPU Inference (Running Ollama on GCP CPU)

If you expect this GCP VM to generate model tokens purely on its 4 vCPUs without a GPU:

- **Capabilities:** 16 GB RAM allows loading small/medium 4-bit quantized models like `Llama-3.1-8B` or `Phi-3.5`.
- **The Catch: Generation speed will be slow** (~2 to 6 tokens/second). CPU matrix math in cloud environments lacks GPU parallelism.
- **Cost Caution:** Running standard Compute Engine instances 24/7 with 4 vCPU/16GB costs ~$100–$140/month unless managed with preemptible/spot instances or committed use discounts.

Deployment Recommendation on GCP

If deploying to GCP, run Open WebUI as a Docker container using a simple compose layout:

YAML

```yaml
version: '3.8'

services:
  open-webui:
    image: ghcr.io/open-webui/open-webui:main
    container_name: open-webui
    ports:
      - "80:8080"
    volumes:
      - open-webui_data:/app/backend/data
    environment:
      - OLLAMA_BASE_URL=http://your-home-ip-or-tailscale:11434
      - OPENAI_API_KEY=your_api_key_here
    restart: always

volumes:
  open-webui_data:
```

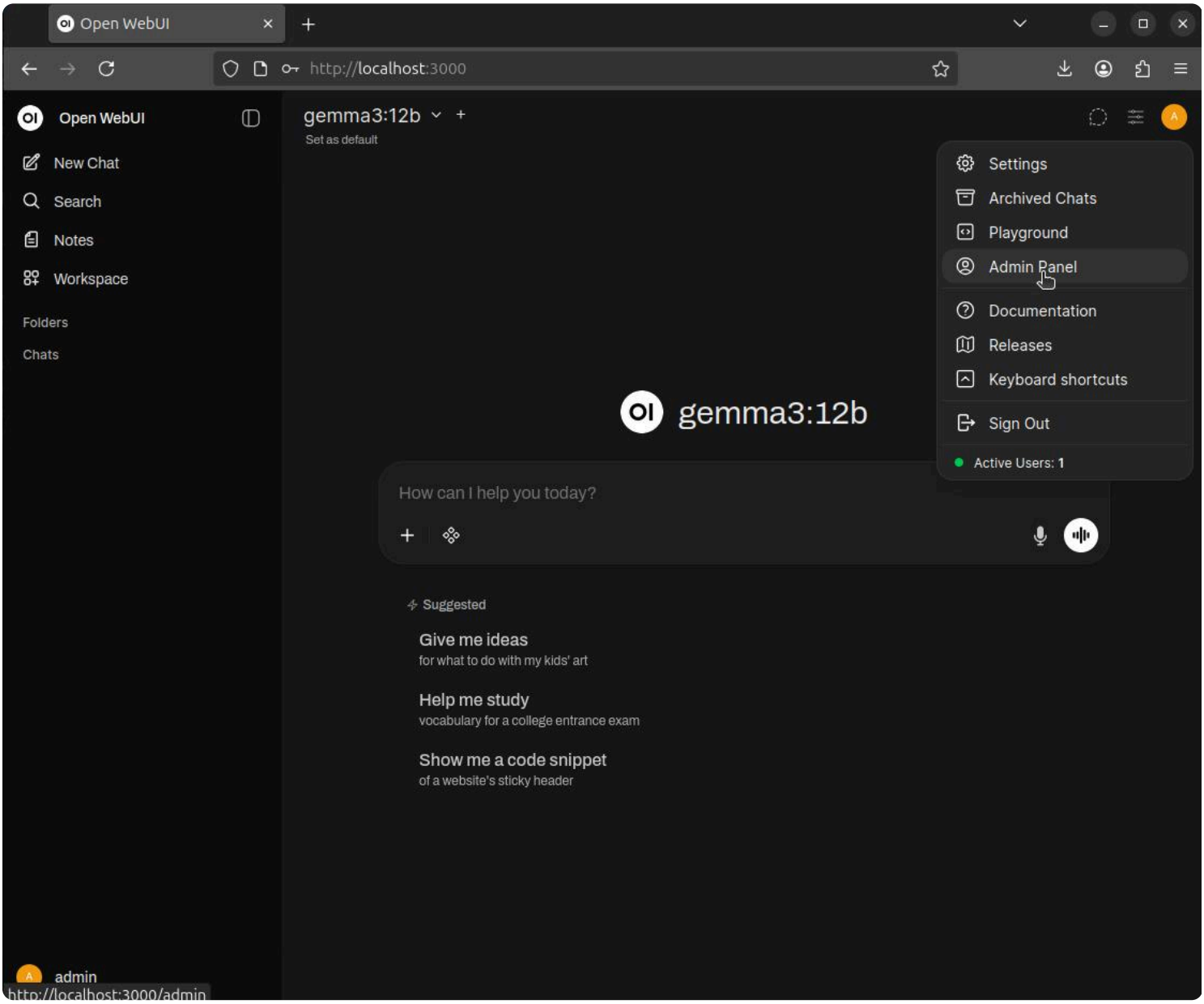
Pro Tips for GCP Deployment

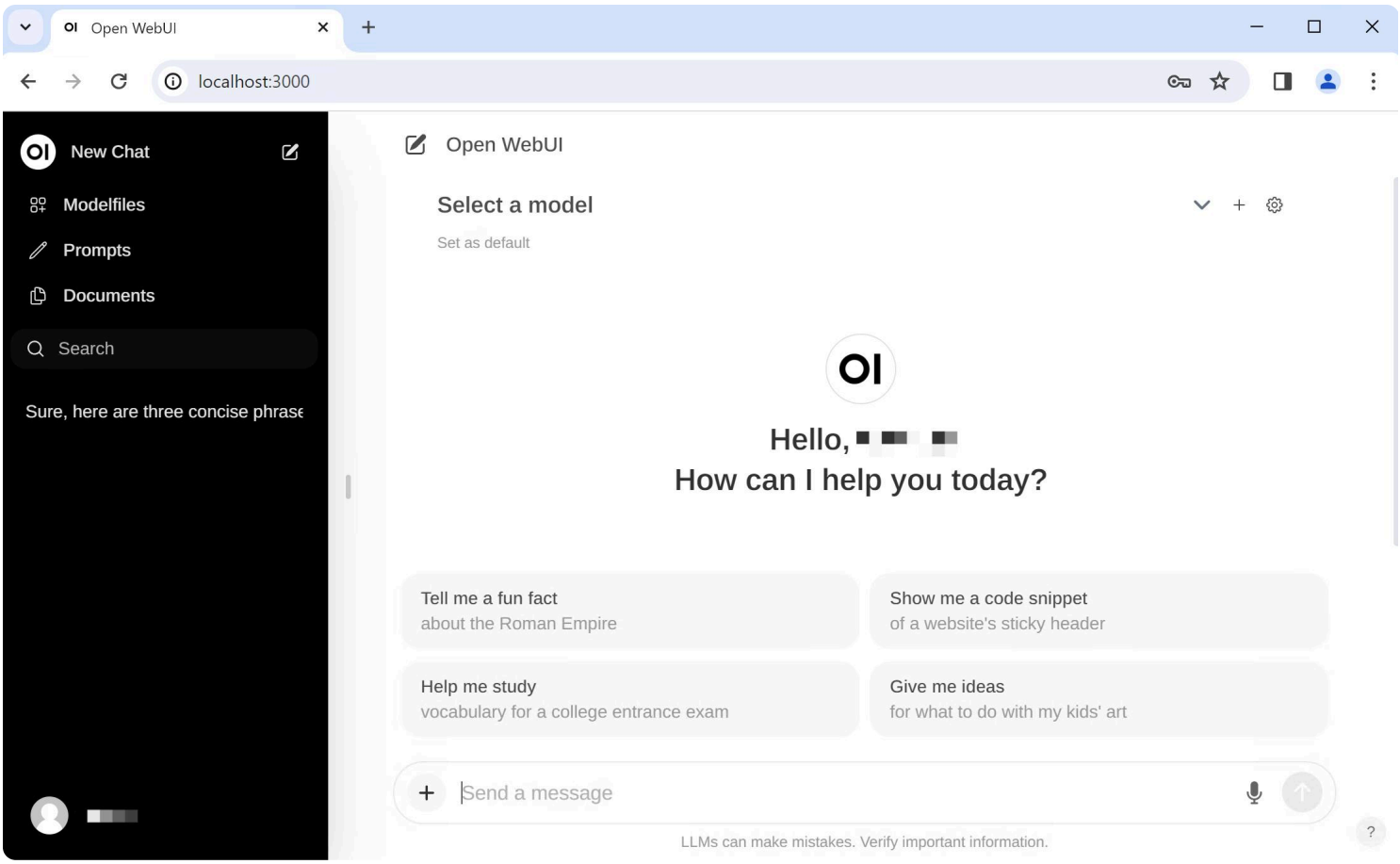
1. **Use Cloud Run (Serverless) instead of VM:** If you only use cloud APIs, deploy Open WebUI to **GCP Cloud Run**. It scales to zero when not in use, cutting costs down to cents/month.
2. **Setup Tailscale/WireGuard:** If your home lab has a powerful GPU running Ollama, keep Open WebUI on GCP and connect it back to your home machine over a private mesh network (Tailscale). This gives you a fast, accessible public Web UI powered by your zero-cost home GPU!

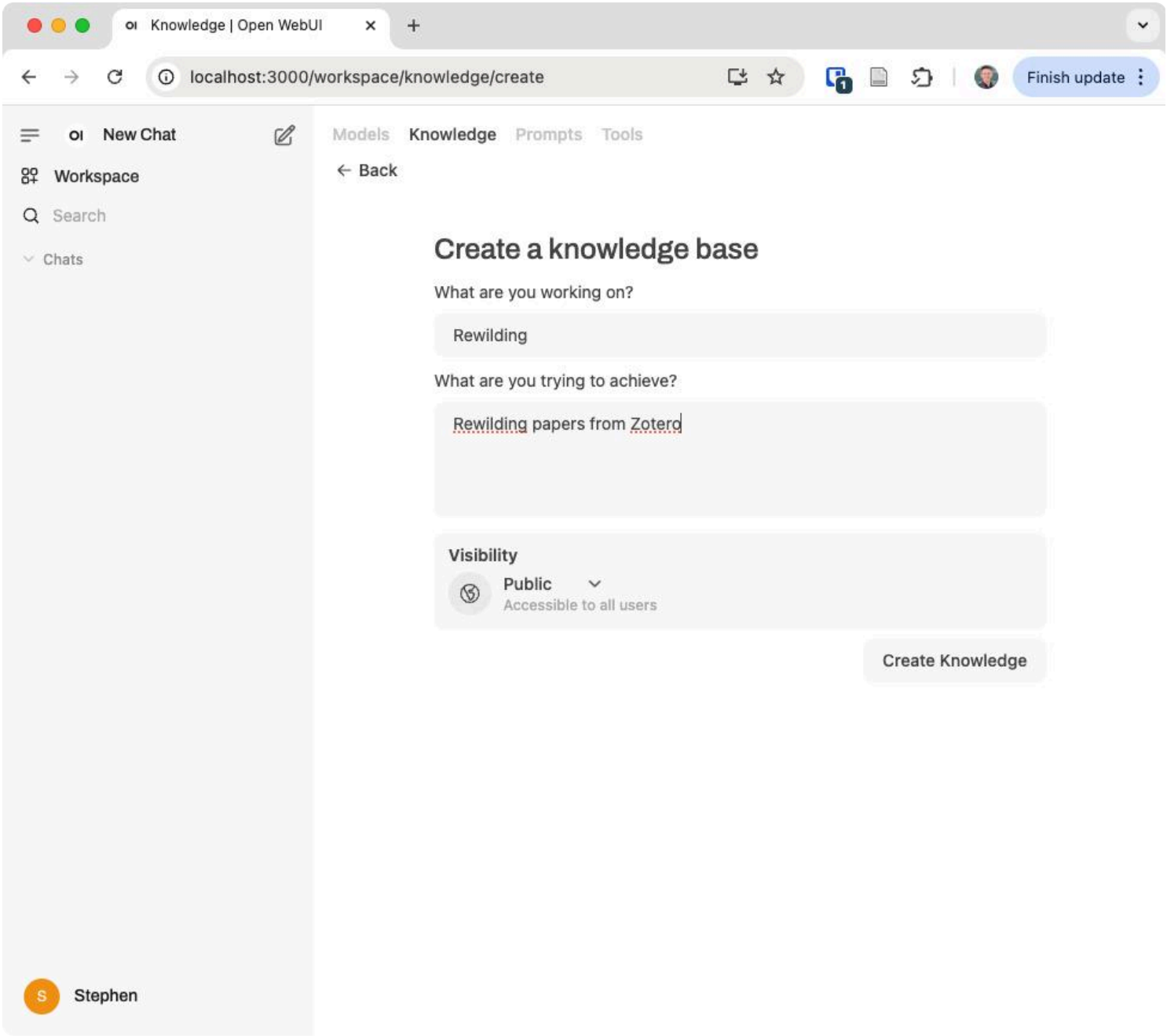
ChatGPT

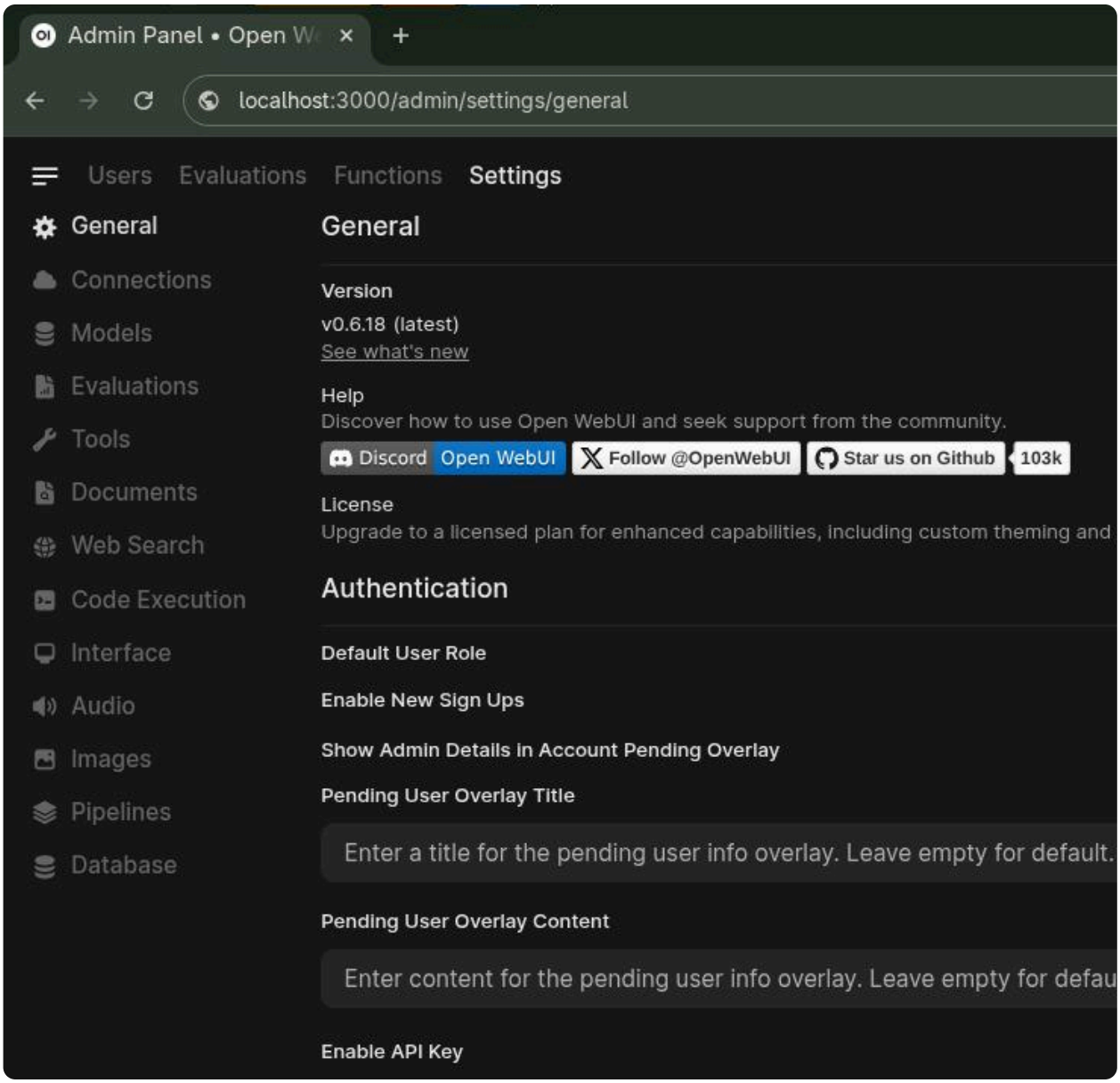
Given your current interests in Docker, GCP, self-hosting, AI, and building a portable homelab, **Open WebUI** is one of the most useful Docker applications you can add. It essentially transforms local or self-hosted LLMs into something that feels similar to ChatGPT, while giving you complete control over the models, documents, users, and data.

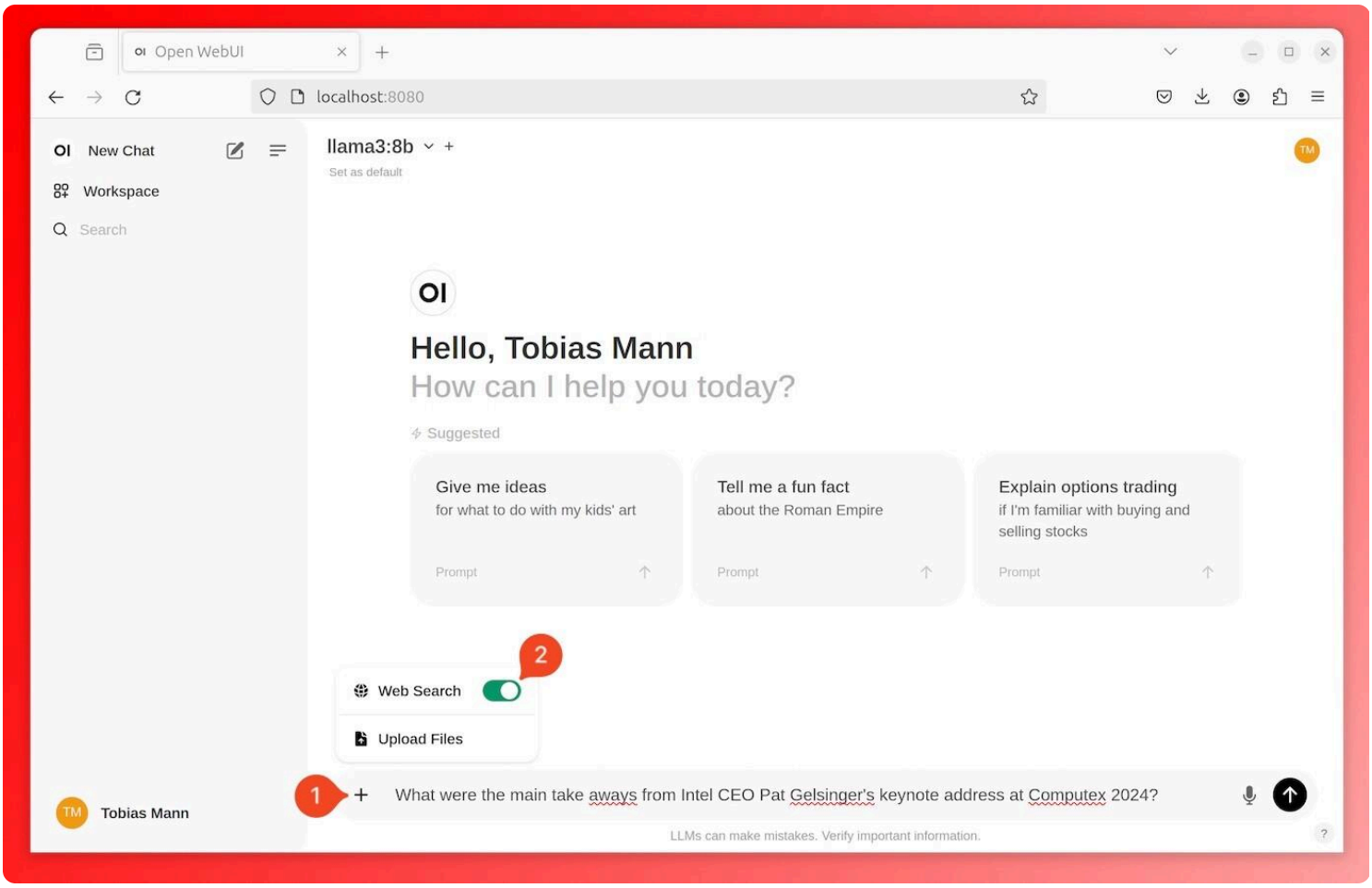
What is Open WebUI?

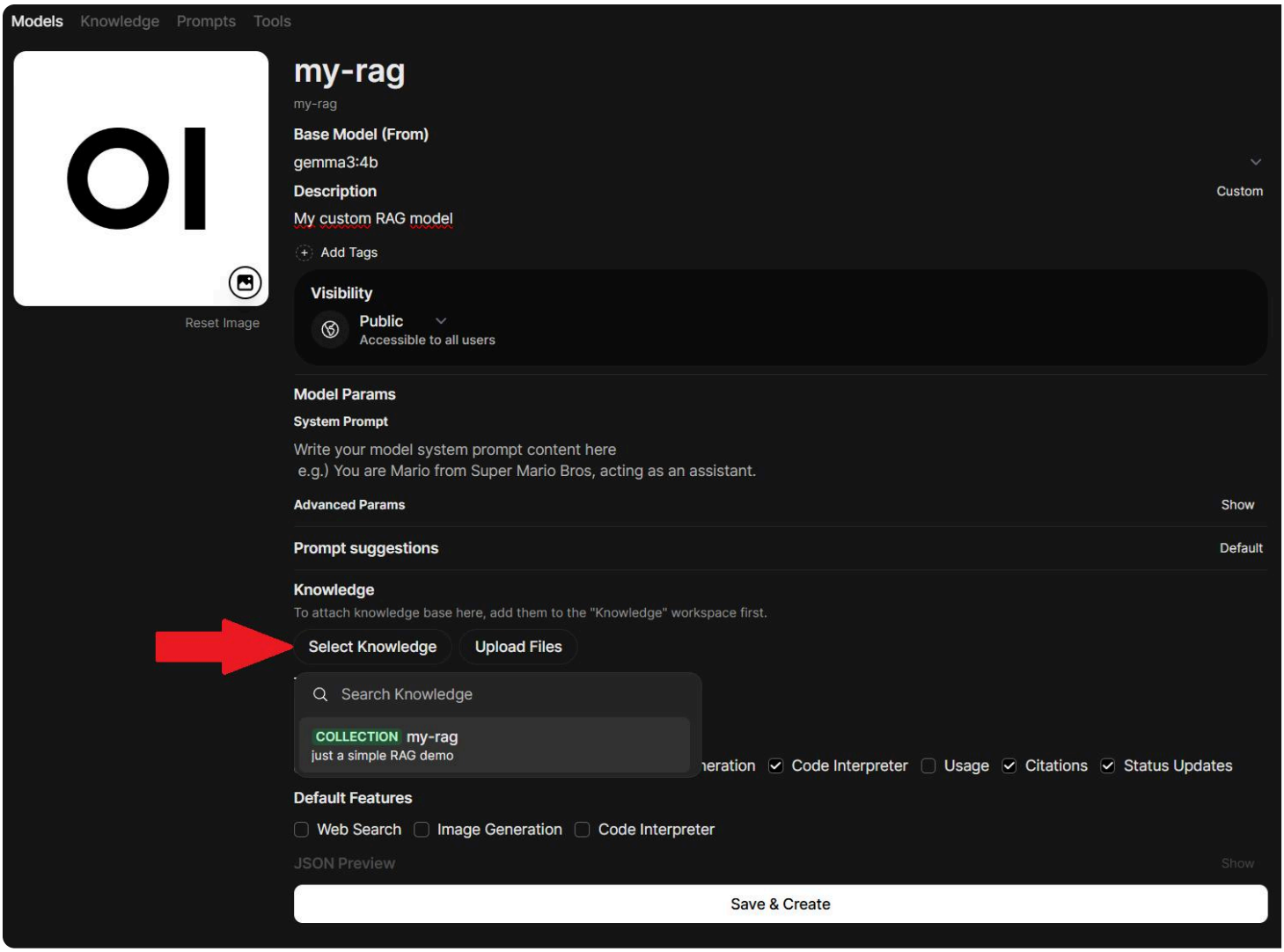



Open WebUI
localhost:3000
New Chat
Modelfiles
Prompts
Documents
Search
Sure, here are three concise phrase
Open WebUI
Select a model
Set as default
Hello,
How can I help you today?
Tell me a fun fact
about the Roman Empire
Show me a code snippet
of a website's sticky header
Help me study
vocabulary for a college entrance exam
Give me ideas
for what to do with my kids' art
Send a message
LLMs can make mistakes. Verify important information.


Knowledge | Open WebUI
localhost:3000/workspace/knowledge/create
Finish update
New Chat
Workspace
Search
Chats
Stephen
Models
Knowledge
Prompts
Tools
Back
Create a knowledge base
What are you working on?
Rewilding
What are you trying to achieve?
Rewilding papers from Zotero
Visibility
Public
Accessible to all users
Create Knowledge

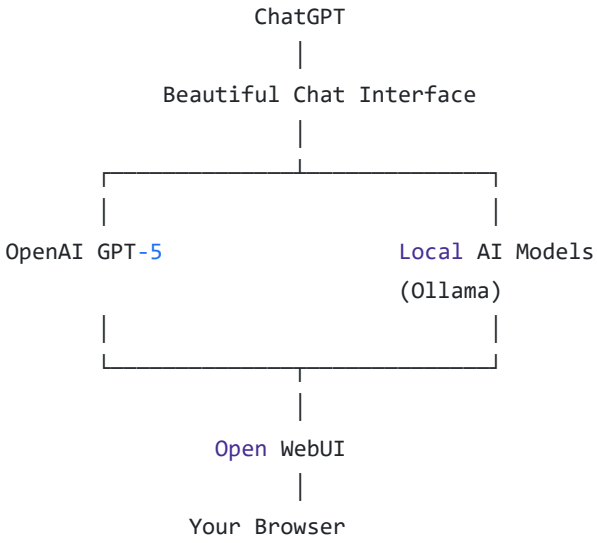

Admin Panel • Open W
localhost:3000/admin/settings/general
Users
Evaluations
Functions
Settings
General
Connections
Models
Evaluations
Tools
Documents
Web Search
Code Execution
Interface
Audio
Images
Pipelines
Database
General
Version
v0.6.18 (latest)
See what's new
Help
Discover how to use Open WebUI and seek support from the community.
Discord Open WebUI
Follow @OpenWebUI
Star us on Github 103k
License
Upgrade to a licensed plan for enhanced capabilities, including custom theming and
Authentication
Default User Role
Enable New Sign Ups
Show Admin Details in Account Pending Overlay
Pending User Overlay Title
Enter a title for the pending user info overlay. Leave empty for default.
Pending User Overlay Content
Enter content for the pending user info overlay. Leave empty for defau
Enable API Key


Open WebUI
localhost:8080
New Chat
Workspace
Search
llama3:8b
Set as default
Hello, Tobias Mann
How can I help you today?
Suggested
Give me ideas
for what to do with my kids' art
Prompt
Tell me a fun fact
about the Roman Empire
Prompt
Explain options trading
if I'm familiar with buying and selling stocks
Prompt
2
Web Search
Upload Files
1
What were the main take aways from Intel CEO Pat Gelsinger's keynote address at Computex 2024?
Tobias Mann
LLMs can make mistakes. Verify important information.



Open WebUI is an open-source AI web application that provides a beautiful ChatGPT-like interface for running AI models.

Think of it like this:



It is **not an AI model itself.**

Instead, it is a **front-end** that connects to:

- OpenAI
- Anthropic
- Google Gemini
- Ollama
- vLLM
- LM Studio
- OpenRouter
- LiteLLM
- Local Llama models
- DeepSeek
- Mistral
- Qwen

It can connect to one or many models simultaneously.

Why has it become so popular?

People like it because it combines many AI capabilities into a single interface.

Instead of

- ChatGPT website
- Claude website
- Gemini website
- Ollama CLI

you have

```
One browser window
```

where everything lives.

Main Features

1. ChatGPT-style Interface

Very similar to ChatGPT.

Features include:

- folders
- conversations
- markdown
- syntax highlighting

- code blocks
- copy buttons
- regenerate response
- edit prompts
- dark mode
- mobile responsive

For many users it actually feels more polished than early ChatGPT versions.

---

2. Multiple AI Models

Example:

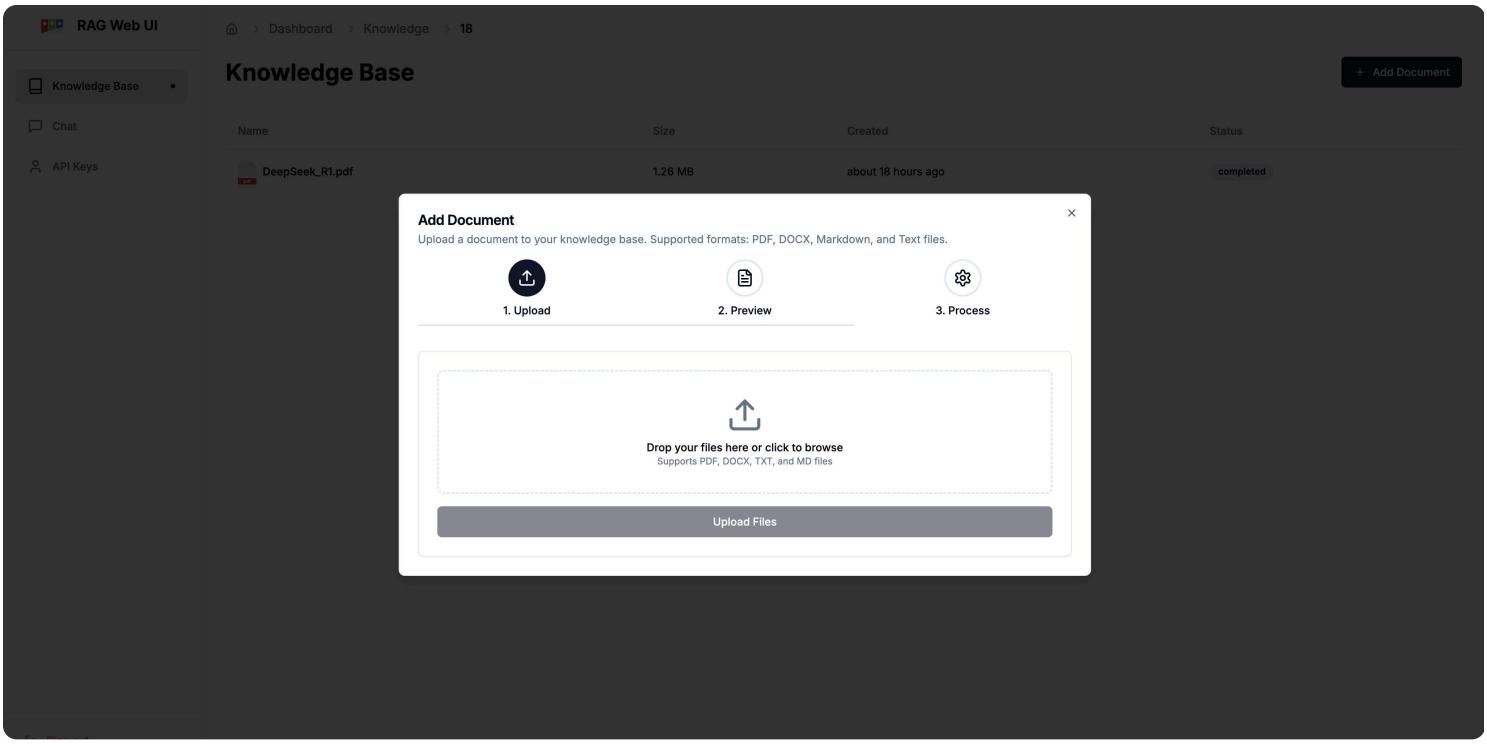
```
GPT-5

Claude Sonnet

Gemini 2.5

DeepSeek

Llama 3.3

Qwen

Mistral
```

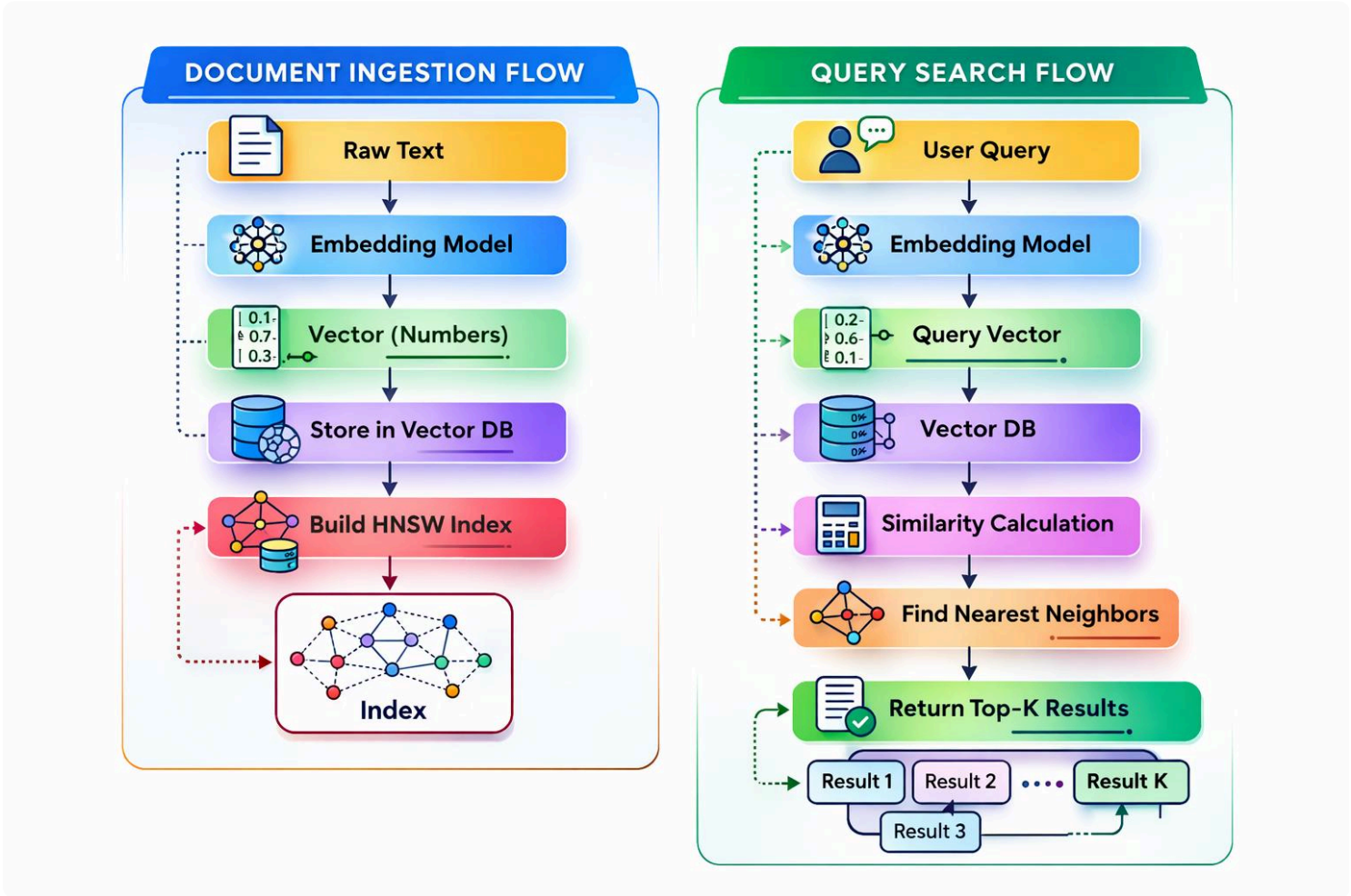
Switch models instantly.

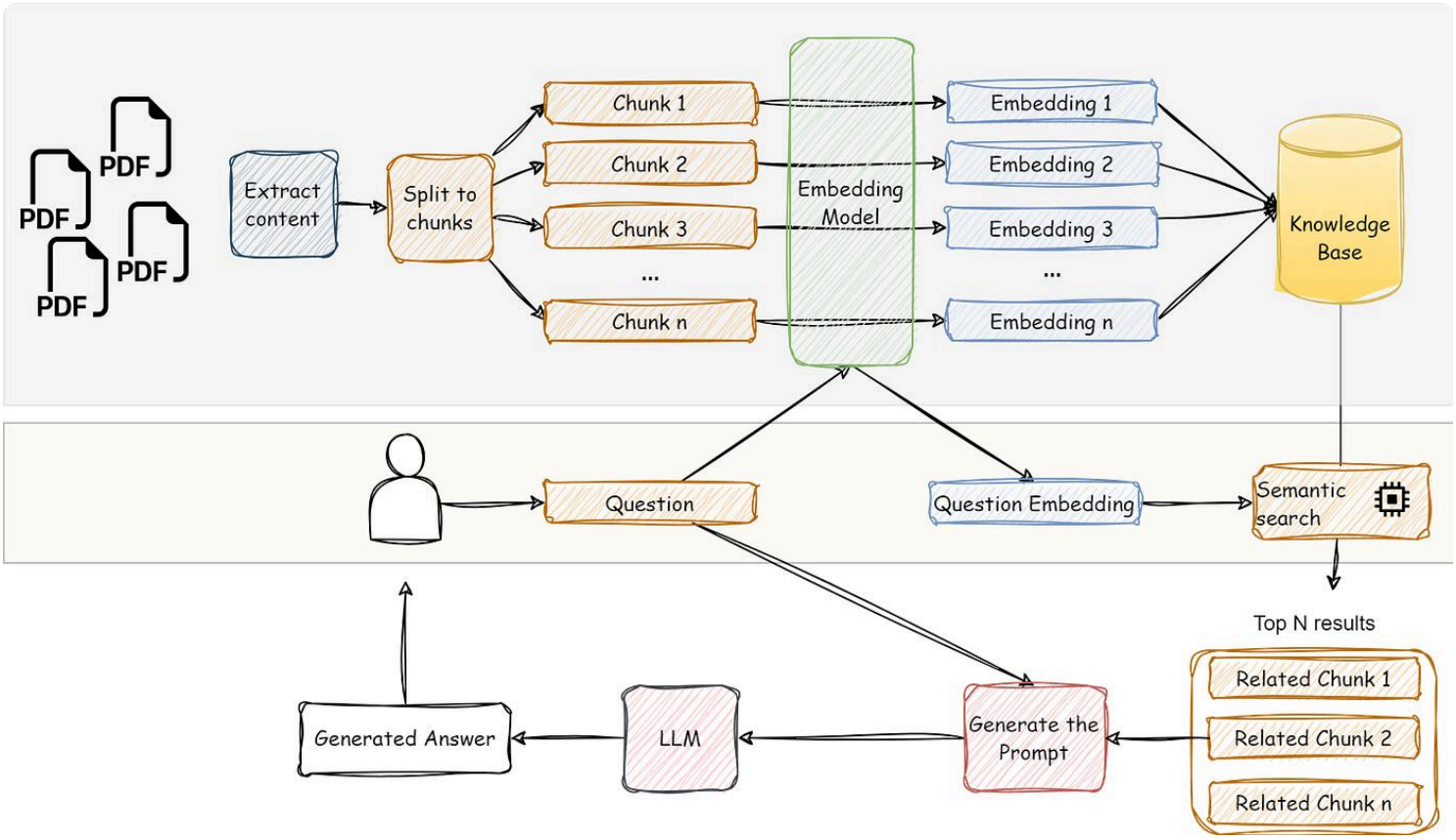
---

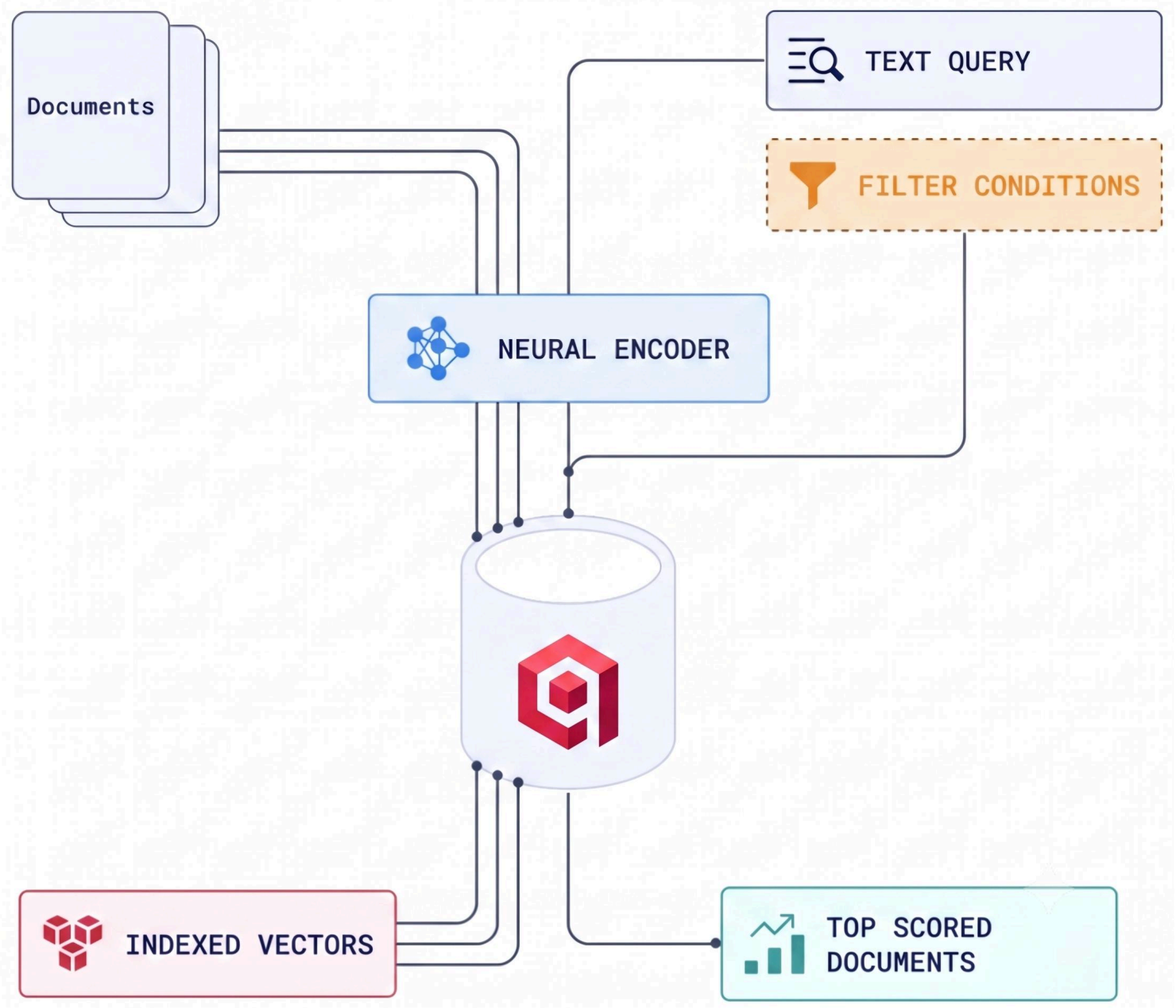
3. Knowledge Base (RAG)

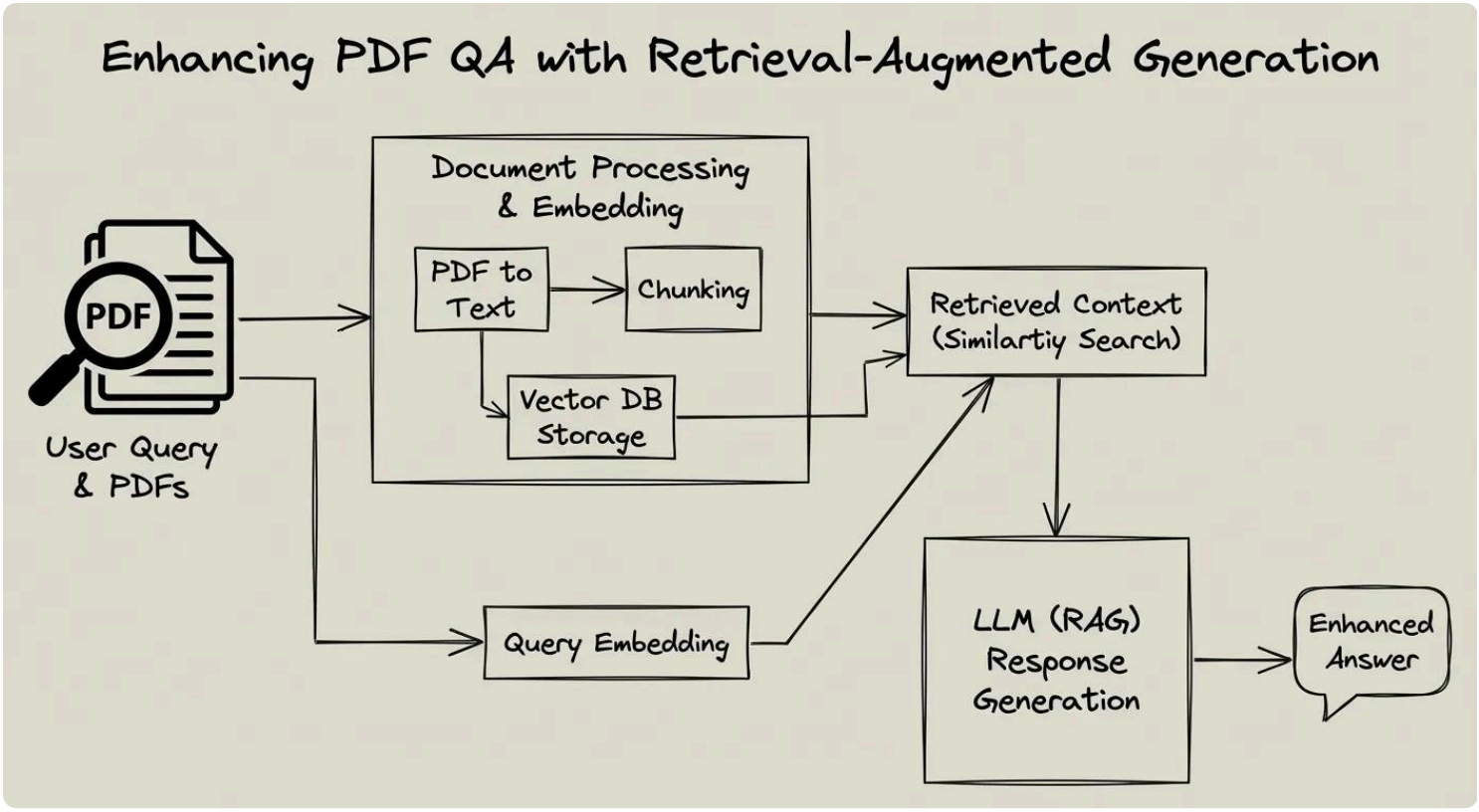
This is one of the biggest reasons people deploy Open WebUI.


RAG Web UI
Dashboard
Knowledge
18
Knowledge Base
Chat
API Keys
Knowledge Base
Add Document
Name
Size
Created
Status
DeepSeek_R1.pdf
1.26 MB
about 18 hours ago
completed
Add Document
Upload a document to your knowledge base. Supported formats: PDF, DOCX, Markdown, and Text files.
1. Upload
2. Preview
3. Process
Drop your files here or click to browse
Supports PDF, DOCX, TXT, and MD files
Upload Files


DOCUMENT INGESTION FLOW
Raw Text
Embedding Model
Vector (Numbers)
Store in Vector DB
Build HNSW Index
Index
QUERY SEARCH FLOW
User Query
Embedding Model
Query Vector
Vector DB
Similarity Calculation
Find Nearest Neighbors
Return Top-K Results
Result 1
Result 2
Result K
Result 3


PDF
PDF
PDF
PDF
Extract content
Split to chunks
Chunk 1
Chunk 2
Chunk 3
...
Chunk n
Embedding Model
Embedding 1
Embedding 2
Embedding 3
...
Embedding n
Knowledge Base
Question
Question Embedding
Semantic search
Top N results
Related Chunk 1
Related Chunk 2
Related Chunk n
Generate the Prompt
LLM
Generated Answer


Documents
TEXT QUERY
FILTER CONDITIONS
NEURAL ENCODER
INDEXED VECTORS
TOP SCORED DOCUMENTS



Imagine uploading

```
100 PDFs

500 Word docs

Medical textbooks

Financial reports

Docker documentation

Google Cloud docs

PostgreSQL manuals
```

Open WebUI indexes everything.

Then you ask:

> How do I configure PostgreSQL replication?

Instead of searching the web...

The AI searches YOUR documents.

This is called

RAG

Retrieval-Augmented Generation.

It supports:

- PDF
- Word
- Markdown
- HTML
- TXT
- CSV
- PowerPoint
- Websites
- GitHub repositories

Example

You upload

```
Docker Compose

Ubuntu docs

OpenEMR docs

MariaDB manual

Google Cloud documentation

Portainer docs

Nginx Proxy Manager
```

Now ask:

> Show me how Portainer communicates with Docker Engine.

It answers using your uploaded documentation.

Knowledge Collections

You can organize documents into collections.

Example

```
Medical

Finance
```

```
Docker

Linux

Google Cloud

Programming

Investing
```

Each collection can have its own chatbot.

Search

Semantic search

instead of keyword search.

Example

Search:

> Docker reverse proxy

It finds documents even if those exact words never appear together.

Memory

Each chat remembers context.

```
Yesterday

Continue conversation
```

Works similarly to ChatGPT conversations.

User Accounts

Supports

- Admin
- User
- Teams

Perfect for families or small businesses.

Sharing

Share chats with others.

Great for organizations.

Plugins / Tools

Can integrate with

- Web Search
- Python
- Image generation
- APIs
- MCP servers
- Custom tools

Voice

Supports

Speech-to-text

Text-to-speech

Microphone input

Image Support

Upload

```
Photos

Charts

Diagrams

Medical scans

Screenshots
```

Ask questions about them (subject to the capabilities of the connected model).

Code Assistant

Excellent for

Docker

Python

SQL

Terraform

Ansible

GCP

Kubernetes

Linux

API Keys

You can connect simultaneously to

OpenAI

Anthropic

Gemini

OpenRouter

DeepSeek

Ollama

etc.

Authentication

Supports

LDAP

OAuth

SSO

Google login

GitHub login

Docker Deployment

Usually only

```
services:

  open-webui:
```

```
    ollama:
```

Sometimes

```
Open WebUI

+

Ollama

+

Qdrant

+

Redis

+

Postgres
```

depending on features.

Hardware Requirements

This depends on whether you are **only hosting the interface** or also **running local AI models**.

Option 1: Open WebUI + Cloud APIs (OpenAI, Gemini, Claude)

This is lightweight because the heavy AI computation happens in the cloud.

Recommended:

- 2 vCPUs
- 4–8 GB RAM
- 20–40 GB SSD

CPU usage is minimal.

Option 2: Open WebUI + Ollama + 7–8B model (Llama, Qwen, Mistral)

Recommended:

- 8 CPU cores
- 16–32 GB RAM
- 100 GB SSD

- No GPU required, but responses will be slower.

Option 3: Serious Local AI

For 13B–70B models, a dedicated GPU (such as NVIDIA RTX 4090 or enterprise GPUs with large VRAM) becomes the practical choice. CPU-only systems can run some larger models with quantization, but performance drops significantly.

Your Home DockerLab

Based on the DockerLab you've described, a practical architecture would be:

```
Ubuntu Server

Docker

Portainer

Nginx Proxy Manager

MariaDB

PostgreSQL

OpenEMR

OpenProject

Docmost

Inventree

Open WebUI

Ollama

Qdrant
```

The largest resource consumer will be Ollama and the models you load. Open WebUI itself is relatively light.

GCP

Your current GCP VM:

```
4 vCPU
```

```
16 GB RAM

80 GB SSD
```

is **more than adequate** for:

- Open WebUI
- Ollama with one 7B–8B quantized model loaded at a time
- RAG with a modest knowledge base
- Docker, Portainer, and NGINX Proxy Manager

It will not provide fast inference compared with a GPU, but it is excellent for learning, experimentation, coding assistance, and document Q&A. If you use cloud APIs (OpenAI, Gemini, Claude) through Open WebUI instead of local models, this VM has ample capacity because the AI inference runs on the provider's infrastructure.

Why it aligns well with your environment

Given your focus on Docker, GCP, DevOps, databases, and self-hosting, Open WebUI can become the **single AI portal** for your lab. You could centralize documentation for Docker, PostgreSQL, MariaDB, Google Cloud, OpenEMR, and your own notes, then query all of it through one interface. Combined with cloud models for maximum capability or Ollama for offline experimentation, it integrates naturally into the stack you've already been building.