

open-webui / open-webui

🔍

👤

▼

📁

⛶

<> Code 167 Issues Pull requests 25 Discussions Actions Security 1 Insights

👁

❤

🔑

☆

User-friendly AI Interface (Supports Ollama, OpenAI API, ...)

🔗 openwebui.com

📄 BSD-3-Clause license

📄 Code of conduct

📄 Security policy

★ 82.7k stars 🍴 10k forks 👁 391 watching 🌿 Branches 🏠 Activity 📋 Custom properties 🏷 Tags

🌐 Public repository

🌿 main 4 Branches 96 Tags 🔑 📁 🔍 Go to file t Go to file + Add file <> Code ...

👤 tjbck

Update bug_report.yaml

✖

b03fc97 · 4 days ago 🕒

📁 .github	Update bug_report.yaml	4 days ago
📁 backend	chore: format	last week
📁 cypress	feat: rich text input for chat	5 months ago
📁 docs	docs/apache.md: Add websocket proxy	2 months ago
📁 kubernetes	Improve kustomization usage	6 months ago
📁 scripts	chore: format	3 weeks ago
📁 src	fix: pinyin keyboard	last week
📁 static	fix: site.webmanifest	last week
📁 test/test_files/image_gen	feat: add image gen with automatic1111 to in...	10 months ago
📄 .dockerignore	feat: Initial support for pgvector	4 months ago
📄 .env.example	refac: byebye litellm	10 months ago
📄 .eslintignore	chat feature added	2 years ago
📄 .eslintrc.cjs	feat: add basic cypress test as initial work tow...	11 months ago
📄 .gitattributes	fixed Docker problem some users experienced	last year
📄 .gitignore	feat(config): exclude .vscode/settings.json fro...	9 months ago
📄 .npmrc	chat feature added	2 years ago
📄 .prettierignore	chore: format	10 months ago
📄 .prettierrc	chat feature added	2 years ago
📄 CHANGELOG.md	doc: changelog	last week
📄 CODE_OF_CONDUCT.md	[DATALAD RUNCMD] run codespell throughou...	2 months ago
📄 Caddyfile.localhost	feat: update .env.example and add Caddyfile	2 years ago
📄 Dockerfile	refac	5 months ago

 INSTALLATION.md	rename	last year
 LICENSE	Update LICENSE	2 months ago
 Makefile	Chose between "docker-compose" and "docke...	11 months ago
 README.md	chore: format	3 weeks ago
 TROUBLESHOOTING.md	fix: Fix typos	5 months ago
 confirm_remove.sh	Chose between "docker-compose" and "docke...	11 months ago
 cypress.config.ts	feat: add basic cypress test as initial work tow...	11 months ago
 demo.gif	doc: demo.gif	10 months ago
 docker-compose.a1111-test.yaml	feat: add image gen with automatic1111 to in...	10 months ago
 docker-compose.amdgpu.yaml	Add variables	11 months ago
 docker-compose.api.yaml	Removed version synatax as its no longer nee...	10 months ago
 docker-compose.data.yaml	Removed version synatax as its no longer nee...	10 months ago
 docker-compose.gpu.yaml	Removed version synatax as its no longer nee...	10 months ago
 docker-compose.playwright.yaml	refac: RAG_WEB_LOADER -> RAG_WEB_LOADE...	3 weeks ago
 docker-compose.yaml	Removed version synatax as its no longer nee...	10 months ago
 hatch_build.py	refac: rename build hash vars	10 months ago
 i18next-parser.config.ts	fix: configure i18next to not return empty stri...	last year
 package-lock.json	chore: format	last week
 package.json	chore: format	last week
 postcss.config.js	chore: format	3 weeks ago
 pyproject.toml	chore: pyproject	last week
 run-compose.sh	Introduce docker-compose.playwright.yaml + ...	last month
 run-ollama-docker.sh	fix: run-ollama-docker.sh	last year
 run.sh	Update run.sh	last year
 svelte.config.js	refac: tool calls	last month
 tailwind.config.js	refac: styling	last month
 tsconfig.json	chat feature added	2 years ago
 update_ollama_models.sh	fix: chmod +x update_ollama_models.sh	3 months ago
 uv.lock	increment unstructured package version	last month
 vite.config.ts	refac: transformer.js	last month

Open WebUI

 Stars  83k  Forks  10k  Watchers  391

repo size

168.1 MiB

languages

11

javascript

40.4%

last commit

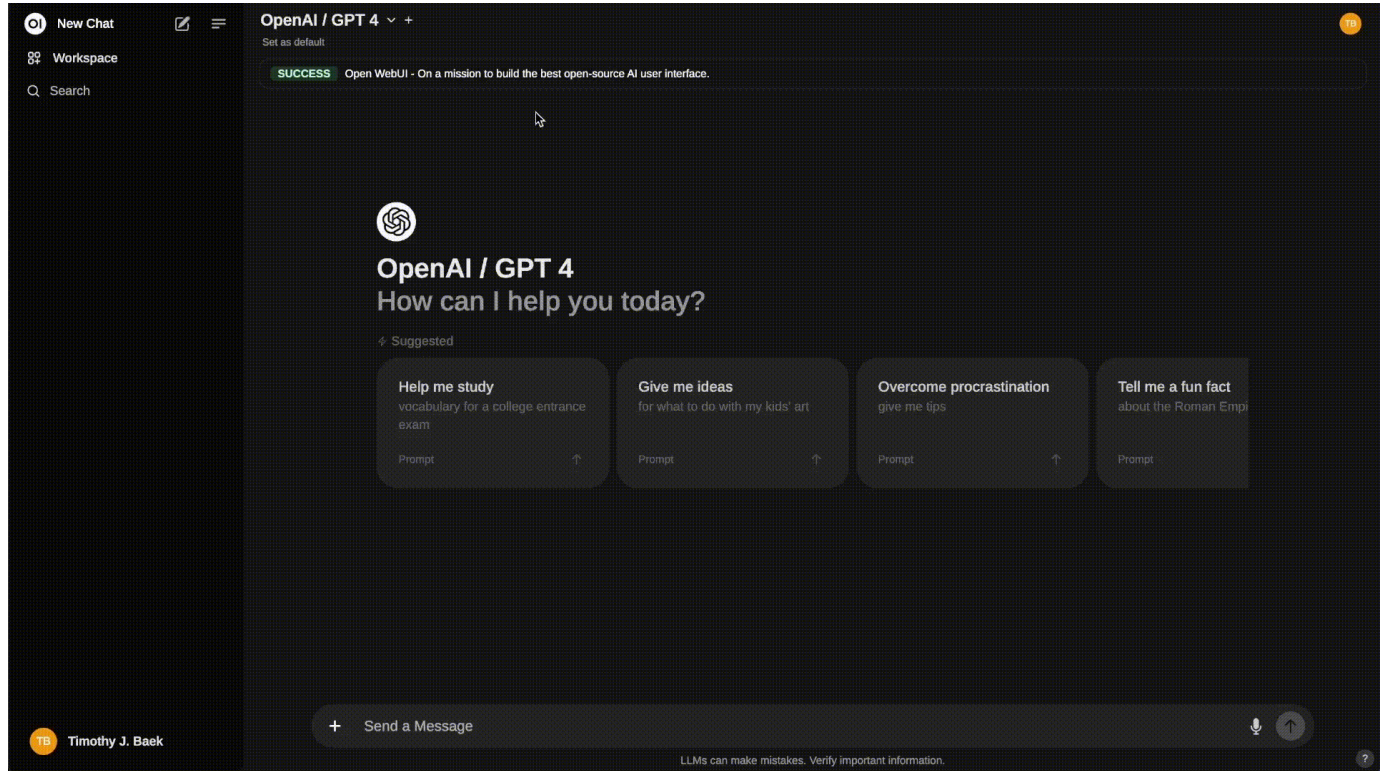
last saturday

hits

34140 / 11997999

 Discord  Open WebUI  Sponsor 

Open WebUI is an [extensible](#), feature-rich, and user-friendly self-hosted AI platform designed to operate entirely offline. It supports various LLM runners like **Ollama** and **OpenAI-compatible APIs**, with **built-in inference engine** for RAG, making it a **powerful AI deployment solution**.



Tip

Looking for an [Enterprise Plan](#)? – [Speak with Our Sales Team Today!](#)

Get **enhanced capabilities**, including **custom theming and branding**, **Service Level Agreement (SLA) support**, **Long-Term Support (LTS) versions**, and **more!**

For more information, be sure to check out our [Open WebUI Documentation](#).

Key Features of Open WebUI ☆

- 🔧 **Effortless Setup:** Install seamlessly using Docker or Kubernetes (kubectrl, kustomize or helm) for a hassle-free experience with support for both `:ollama` and `:cuda` tagged images.
- 🔗 **Ollama/OpenAI API Integration:** Effortlessly integrate OpenAI-compatible APIs for versatile conversations alongside Ollama models. Customize the OpenAI API URL to link with **LMStudio**, **GroqCloud**, **Mistral**, **OpenRouter**, and **more**.
- ♥ **Granular Permissions and User Groups:** By allowing administrators to create detailed user roles and permissions, we ensure a secure user environment. This granularity not only enhances security but also allows for customized user experiences, fostering a sense of ownership and responsibility amongst users.
- 📱 **Responsive Design:** Enjoy a seamless experience across Desktop PC, Laptop, and Mobile devices.
- 📱 **Progressive Web App (PWA) for Mobile:** Enjoy a native app-like experience on your mobile device with our PWA, providing offline access on localhost and a seamless user interface.
- 📄 **Full Markdown and LaTeX Support:** Elevate your LLM experience with comprehensive Markdown and LaTeX capabilities for enriched interaction.
- 🗣️ **Hands-Free Voice/Video Call:** Experience seamless communication with integrated hands-free voice and video call features, allowing for a more dynamic and interactive chat environment.
- ✂️ **Model Builder:** Easily create Ollama models via the Web UI. Create and add custom characters/agents, customize chat elements, and import models effortlessly through [Open WebUI Community](#) integration.

- **🔗 Native Python Function Calling Tool:** Enhance your LLMs with built-in code editor support in the tools workspace. Bring Your Own Function (BYOF) by simply adding your pure Python functions, enabling seamless integration with LLMs.

- **📄 Local RAG Integration:** Dive into the future of chat interactions with groundbreaking Retrieval Augmented Generation (RAG)

📖 README 🔒 Code of conduct 📄 BSD-3-Clause license 🔒 Security

the chat or add them to your document library, effortlessly accessing them using the `#` command before a query.

- **🔍 Web Search for RAG:** Perform web searches using providers like `SearXNG`, `Google PSE`, `Brave Search`, `serpstack`, `serper`, `Serply`, `DuckDuckGo`, `TavilySearch`, `SearchApi` and `Bing` and inject the results directly into your chat experience.
- **🌐 Web Browsing Capability:** Seamlessly integrate websites into your chat experience using the `#` command followed by a URL. This feature allows you to incorporate web content directly into your conversations, enhancing the richness and depth of your interactions.
- **🖼️ Image Generation Integration:** Seamlessly incorporate image generation capabilities using options such as AUTOMATIC1111 API or ComfyUI (local), and OpenAI's DALL-E (external), enriching your chat experience with dynamic visual content.
- **⚙️ Many Models Conversations:** Effortlessly engage with various models simultaneously, harnessing their unique strengths for optimal responses. Enhance your experience by leveraging a diverse set of models in parallel.
- **🛡️ Role-Based Access Control (RBAC):** Ensure secure access with restricted permissions; only authorized individuals can access your Ollama, and exclusive model creation/pulling rights are reserved for administrators.
- **🌍 Multilingual Support:** Experience Open WebUI in your preferred language with our internationalization (i18n) support. Join us in expanding our supported languages! We're actively seeking contributors!
- **🔗 Pipelines, Open WebUI Plugin Support:** Seamlessly integrate custom logic and Python libraries into Open WebUI using [Pipelines Plugin Framework](#). Launch your Pipelines instance, set the OpenAI URL to the Pipelines URL, and explore endless possibilities. [Examples](#) include **Function Calling**, User **Rate Limiting** to control access, **Usage Monitoring** with tools like Langfuse, **Live Translation with LibreTranslate** for multilingual support, **Toxic Message Filtering** and much more.
- **🔄 Continuous Updates:** We are committed to improving Open WebUI with regular updates, fixes, and new features.

Want to learn more about Open WebUI's features? Check out our [Open WebUI documentation](#) for a comprehensive overview!

🔑 Also Check Out Open WebUI Community!

Don't forget to explore our sibling project, [Open WebUI Community](#), where you can discover, download, and explore customized Modelfiles. Open WebUI Community offers a wide range of exciting possibilities for enhancing your chat interactions with Open WebUI! 🚀

How to Install 🚀

Installation via Python pip 🐍

Open WebUI can be installed using pip, the Python package installer. Before proceeding, ensure you're using **Python 3.11** to avoid compatibility issues.

1. **Install Open WebUI:** Open your terminal and run the following command to install Open WebUI:

```
pip install open-webui
```

2. **Running Open WebUI:** After installation, you can start Open WebUI by executing:

```
open-webui serve
```

This will start the Open WebUI server, which you can access at <http://localhost:8080>

Quick Start with Docker 🐳

Note

Please note that for certain Docker environments, additional configurations might be needed. If you encounter any connection issues, our detailed guide on [Open WebUI Documentation](#) is ready to assist you.

⚠ Warning

When using Docker to install Open WebUI, make sure to include the `-v open-webui:/app/backend/data` in your Docker command. This step is crucial as it ensures your database is properly mounted and prevents any loss of data.

💡 Tip

If you wish to utilize Open WebUI with Ollama included or CUDA acceleration, we recommend utilizing our official images tagged with either `:cuda` or `:ollama`. To enable CUDA, you must install the [Nvidia CUDA container toolkit](#) on your Linux/WSL system.

Installation with Default Configuration

- **If Ollama is on your computer**, use this command:

```
docker run -d -p 3000:8080 --add-host=host.docker.internal:host-gateway -v open-webui:/app/backend/data --name op
```

- **If Ollama is on a Different Server**, use this command:

To connect to Ollama on another server, change the `OLLAMA_BASE_URL` to the server's URL:

```
docker run -d -p 3000:8080 -e OLLAMA_BASE_URL=https://example.com -v open-webui:/app/backend/data --name open-web
```

- **To run Open WebUI with Nvidia GPU support**, use this command:

```
docker run -d -p 3000:8080 --gpus all --add-host=host.docker.internal:host-gateway -v open-webui:/app/backend/dat
```

Installation for OpenAI API Usage Only

- **If you're only using OpenAI API**, use this command:

```
docker run -d -p 3000:8080 -e OPENAI_API_KEY=your_secret_key -v open-webui:/app/backend/data --name open-webui --
```

Installing Open WebUI with Bundled Ollama Support

This installation method uses a single container image that bundles Open WebUI with Ollama, allowing for a streamlined setup via a single command. Choose the appropriate command based on your hardware setup:

- **With GPU Support:** Utilize GPU resources by running the following command:

```
docker run -d -p 3000:8080 --gpus=all -v ollama:/root/.ollama -v open-webui:/app/backend/data --name open-webui -
```

- **For CPU Only:** If you're not using a GPU, use this command instead:

```
docker run -d -p 3000:8080 -v ollama:/root/.ollama -v open-webui:/app/backend/data --name open-webui --restart al
```

Both commands facilitate a built-in, hassle-free installation of both Open WebUI and Ollama, ensuring that you can get everything up and running swiftly.

After installation, you can access Open WebUI at <http://localhost:3000>. Enjoy! 😊

Other Installation Methods

We offer various installation alternatives, including non-Docker native installation methods, Docker Compose, Kustomize, and Helm. Visit our [Open WebUI Documentation](#) or join our [Discord community](#) for comprehensive guidance.

Troubleshooting

Encountering connection issues? Our [Open WebUI Documentation](#) has got you covered. For further assistance and to join our vibrant community, visit the [Open WebUI Discord](#).

Open WebUI: Server Connection Error

If you're experiencing connection issues, it's often due to the WebUI docker container not being able to reach the Ollama server at 127.0.0.1:11434 (host.docker.internal:11434) inside the container . Use the `--network=host` flag in your docker command to resolve this. Note that the port changes from 3000 to 8080, resulting in the link: `http://localhost:8080` .

Example Docker Command:

```
docker run -d --network=host -v open-webui:/app/backend/data -e OLLAMA_BASE_URL=http://127.0.0.1:11434 --name open-we
```

Keeping Your Docker Installation Up-to-Date

In case you want to update your local Docker installation to the latest version, you can do it with [Watchtower](#):

```
docker run --rm --volume /var/run/docker.sock:/var/run/docker.sock containrrr/watchtower --run-once open-webui
```

In the last part of the command, replace `open-webui` with your container name if it is different.

Check our Updating Guide available in our [Open WebUI Documentation](#).

Using the Dev Branch 🐛

⚠ Warning

The `:dev` branch contains the latest unstable features and changes. Use it at your own risk as it may have bugs or incomplete features.

If you want to try out the latest bleeding-edge features and are okay with occasional instability, you can use the `:dev` tag like this:

```
docker run -d -p 3000:8080 -v open-webui:/app/backend/data --name open-webui --add-host=host.docker.internal:host-gat
```

Offline Mode

If you are running Open WebUI in an offline environment, you can set the `HF_HUB_OFFLINE` environment variable to `1` to prevent attempts to download models from the internet.

```
export HF_HUB_OFFLINE=1
```

What's Next? 🌟

Discover upcoming features on our roadmap in the [Open WebUI Documentation](#).

License 📄

This project is licensed under the [BSD-3-Clause License](#) - see the [LICENSE](#) file for details. 📄

Support 💬

If you have any questions, suggestions, or need assistance, please open an issue or join our [Open WebUI Discord community](#) to connect with us! 🗨

Star History

 [Star History Chart](#)

Created by [Timothy Jaeryang Baek](#) - Let's make Open WebUI even more amazing together! 

Releases 96

 **v0.5.20** Latest
last week

[+ 95 releases](#)

Sponsor this project

 **tjbck** Timothy Jaeryang Baek

 Sponsor

[Learn more about GitHub Sponsors](#)

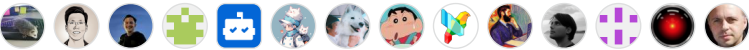
Packages 1

 **open-webui**

Used by 100

 + 92

Contributors 418



[+ 404 contributors](#)

Deployments 400

 **pypi**

[+ 399 deployments](#)

Languages

