

Open WebUI

The large models available in Open WebUI are synchronized with Ollama. Users need to manually pull the large model from Ollama before switching to the corresponding model.

Open WebUI

1. Environment Requirements
2. Docker Setup
 - 2.1 Installing Docker from the Official Website
 - 2.2 Installing Docker on Alibaba Cloud
 - 2.3 Add Access Permissions
 - 2.4 Compatibility Modification
3. Installing Open WebUI
4. Running Open WebUI
 - 4.1 Administrator Account
 - 4.2 Registration and Login
 - 4.3 User Interface
- 5 Model Dialogue
 - 5.1 Switching Models
 - 5.2 Demo Model: minicpm-v:8b
6. Frequently Asked Questions
 - 6.1 Disabling Open WebUI
 - 6.2 Common Errors
 - Unable to start Open WebUI
 - Service Connection Timeout
 - 6.3 Insufficient Memory Error
 - When using a model with a large data volume to generate queries, the following error occurs:

Demo Environment

Development Board: RDK S100(P) Development Board

Open WebUI is an open-source project designed to provide a simple and easy-to-use user interface (UI) for managing and monitoring open-source software and services.

When using Open WebUI, there is a high probability of encountering unresponsive dialogues or timeouts. Try restarting Open webUI or using the ollama tool to run the model!

1. Environment Requirements

Installing Open WebUI on the host machine and Conda: Requires Node.js >= 20.10, Python = 3.11:

Environment Setup Method	Difficulty (relatively speaking)
Host	High
Conda	Medium
Docker	Low

This tutorial demonstrates installing Open WebUI using Docker.

2. Docker Setup

2.1 Installing Docker from the Official Website

If Docker is not installed, you can use a script to install it with one click.

Domestic users may not be able to install Docker through the official methods. We recommend using Alibaba Cloud to install Docker or directly using our image.

- Update local package list

```
sudo apt update
```

- Upgrade installed packages

```
sudo apt upgrade
```

- Download and run script

Download the get-docker.sh file and save it in the current directory.

```
sudo apt install curl
curl -fSSL https://get.docker.com -o get-docker.sh
```

Run the get-docker.sh script file with sudo privileges.

```
sudo sh get-docker.sh
```

2.2 Installing Docker on Alibaba Cloud

If you cannot install it yourself, please use the image we provide.

-Update local package list

```
sudo apt update
```

- Install essential software

```
sudo apt install apt-transport-https ca-certificates curl gnupg2 lsb-release
software-properties-common
```

- Add GPG key for software repository

```
curl -fSSL https://mirrors.aliyun.com/docker-ce/linux/ubuntu/gpg | sudo gpg --
dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
```

- Add Alibaba Cloud mirror software repository

```
echo "deb [arch=arm64 signed-by=/usr/share/keyrings/docker-archive-keyring.gpg]
https://mirrors.aliyun.com/docker-ce/linux/ubuntu bionic stable" | sudo tee
/etc/apt/sources.list.d/docker.list > /dev/null
```

- Install Docker

```
sudo apt update  
sudo apt install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

2.3 Add Access Permissions

Add access permissions for the current system user to the Docker daemon: Docker commands can be used without using the sudo command

```
sudo usermod -aG docker $USER  
newgrp docker
```

2.4 Compatibility Modification

After successful installation, starting Docker may fail:

```
sunrise@ubuntu:~$ docker images  
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?  
sunrise@ubuntu:~$ sudo systemctl start docker  
Job for docker.service failed because the control process exited with error code  
. See "systemctl status docker.service" and "journalctl -xeu docker.service" for details.
```

This issue stems from a compatibility conflict between **Linux network stack architecture updates** and **Docker's network implementation**. This error typically occurs on newer Linux distributions that default to the nf_tables backend, which Docker may have compatibility issues with. Therefore, switching to legacy mode (using iptables-legacy) can resolve most such issues.

1. Switch to legacy iptables

```
sudo update-alternatives --set iptables /usr/sbin/iptables-legacy  
sudo update-alternatives --set ip6tables /usr/sbin/ip6tables-legacy
```

2. Check the current iptables mode

```
sudo iptables --version  
# It should display "legacy" instead of "nf_tables"
```

3. Clean up iptables rules

```
sudo iptables -t nat -F  
sudo iptables -t nat -X  
sudo iptables -F  
sudo iptables -X
```

4. Restart Docker

```
sudo systemctl restart docker
```

5. Check if the Docker service is running successfully

```
sudo systemctl status Docker
```

```
sunrise@ubuntu:~$ sudo systemctl status docker
● docker.service - Docker Application Container Engine
   Loaded: loaded (/lib/systemd/system/docker.service; enabled; vendor preset: on)
   Active: active (running) since Tue 2025-07-01 11:02:00 CST; 45min ago
     Tasks: 12
    Memory: 72.5M
      CGroup: /system.slice/docker.service
              └─10957 /usr/bin/dockerd -H fd:// --containerd=/run/containerd/con...
Jul 01 11:46:40 ubuntu dockerd[10957]: time="2025-07-01T03:46:40.752680093Z" level=info msg="Starting Docker daemon"
Jul 01 11:46:40 ubuntu dockerd[10957]: time="2025-07-01T03:46:40.758599787Z" level=info msg="Docker daemon has started successfully"
Jul 01 11:46:40 ubuntu dockerd[10957]: time="2025-07-01T03:46:40.758646336Z" level=info msg="API endpoint available at https://0.0.0.0:2375"
Jul 01 11:46:40 ubuntu dockerd[10957]: time="2025-07-01T03:46:40.758661286Z" level=info msg="Docker daemon is ready to handle requests"
Jul 01 11:46:40 ubuntu dockerd[10957]: time="2025-07-01T03:46:40.758672061Z" level=info msg="Listening on fd://"
Jul 01 11:46:40 ubuntu dockerd[10957]: time="2025-07-01T03:46:40.758680561Z" level=info msg="Containerd integration initialized"
Jul 01 11:46:41 ubuntu dockerd[10957]: time="2025-07-01T03:46:41.027717943Z" level=info msg="Docker daemon is ready to handle requests"
Jul 01 11:46:41 ubuntu dockerd[10957]: time="2025-07-01T03:46:41.027767992Z" level=info msg="Listening on fd://"
Jul 01 11:46:41 ubuntu dockerd[10957]: time="2025-07-01T03:46:41.027786792Z" level=info msg="Containerd integration initialized"
Jul 01 11:46:41 ubuntu dockerd[10957]: time="2025-07-01T03:46:41.027797442Z" level=info msg="Docker daemon is ready to handle requests"
lines 1-21/21 (END)
```

3. Installing Open WebUI

For systems with Docker already installed, you can directly enter the following command in the terminal: The image shows the result after pulling.

```
docker pull ghcr.io/open-webui/open-webui:main
```

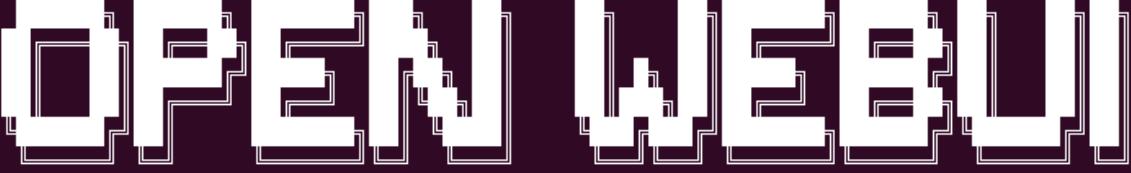
```
sunrise@ubuntu:~$ docker pull ghcr.io/open-webui/open-webui:main
main: Pulling from open-webui/open-webui
34ef2a75627f: Pull complete
6939e8b629d3: Pull complete
8d8c9691b884: Pull complete
4185d6dac69a: Pull complete
94534955d077: Pull complete
4f4fb700ef54: Pull complete
e2fd59778c8f: Pull complete
54d3d80ceb4f: Pull complete
644d18cacda5: Pull complete
2050ba16d878: Pull complete
c1cab6acf868: Pull complete
bd6e015e167e: Pull complete
fb458fc13485: Pull complete
4c66790b05a3: Pull complete
feeee3f706c3a: Pull complete
Digest: sha256:e18a03441385208828d527975e0247590640a72c39f5c1321ff203eec43a924b
Status: Downloaded newer image for ghcr.io/open-webui/open-webui:main
ghcr.io/open-webui/open-webui:main
sunrise@ubuntu:~$
```

4. Running Open WebUI

Enter the following command in the terminal to start the specified Docker container:

```
docker run --network=host -v open-webui:/app/backend/data -e OLLAMA_BASE_URL=http://127.0.0.1:11434 --name open-webui --restart always ghcr.io/open-webui/open-webui:main
```

```
+ sunrise@ubuntu: ~
WARNI [langchain_community.utils.user_agent] USER_AGENT environment variable not
set, consider setting it to identify your requests.
INFO:     Started server process [1]
INFO:     Waiting for application startup.

  
v0.6.15 - building the best AI user interface.  
https://github.com/open-webui/open-webui  
2025-07-01 07:29:32.474 | INFO      | open_webui.utils.logger:start_logger:140 -  
GLOBAL_LOG_LEVEL: INFO - {}  
2025-07-01 07:29:32.475 | INFO      | open_webui.main:lifespan:514 - Installing e  
xternal dependencies of functions and tools... - {}  
2025-07-01 07:29:32.498 | INFO      | open_webui.utils.plugin:install_frontmatter  
_requirements:241 - No requirements found in frontmatter. - {}
```

After successful startup, access it in your browser using the following URL:

```
http://localhost:8080/
```

![image-20250701153046904](i (mage-20250701153046904.png)

Access can be made via the motherboard IP:8080 on the same local area network:

```
Assuming the motherboard IP:192.168.2.105, we can access it via  
192.168.2.105:8080
```

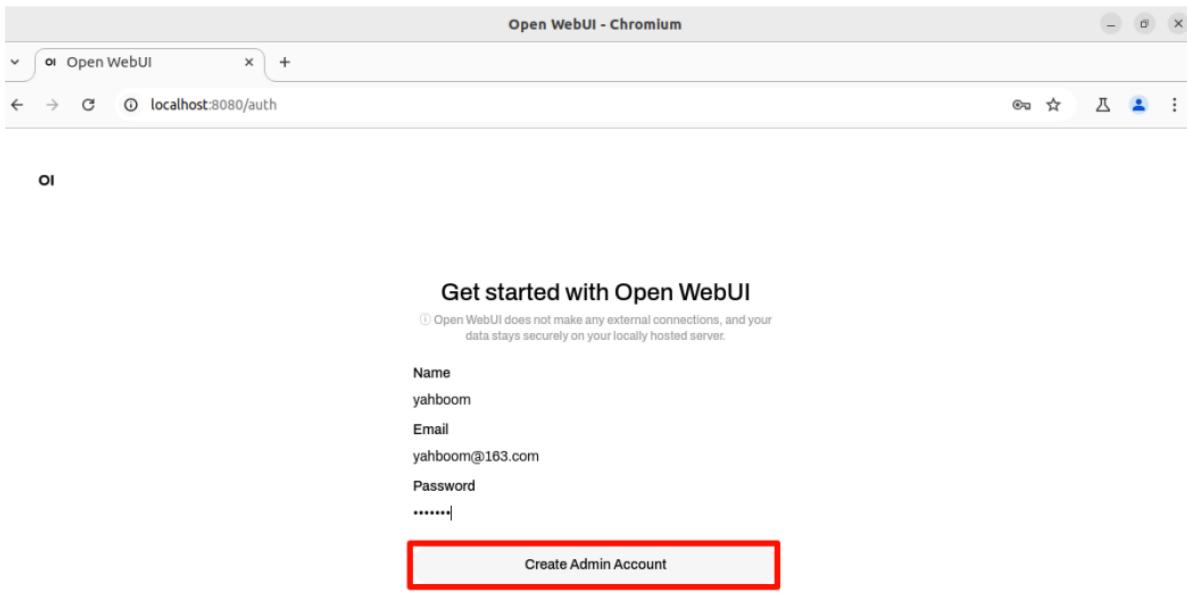
This method is recommended because when accessing the site locally on the motherboard, the browser requires some memory. If you then try to run a 7B or 8B model, it will prompt an insufficient memory error and fail to run.

4.1 Administrator Account

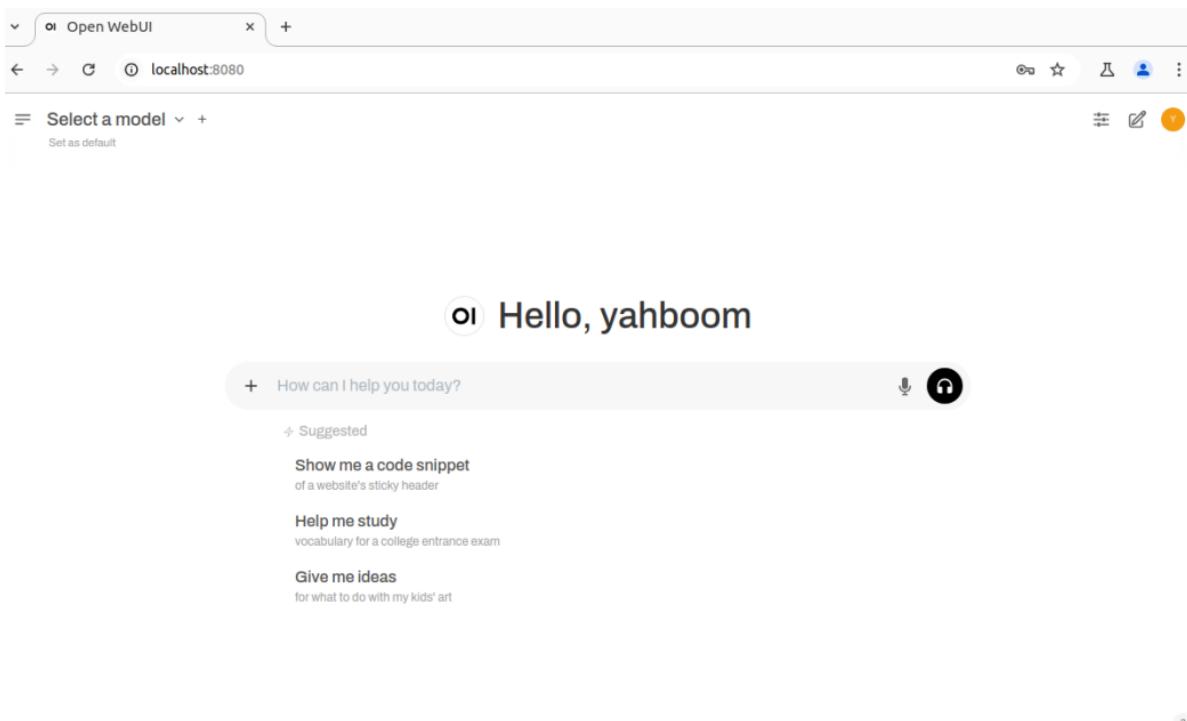
First-time users need to register an account. This account is the administrator account. You can fill in the required information yourself!

```
Since all content in our mirror has been set up and tested, users can log in  
directly using their registered account:  
Username: yahboom  
Email: yahboom@163.com  
Password: yahboom
```

4.2 Registration and Login



4.3 User Interface



5 Model Dialogue

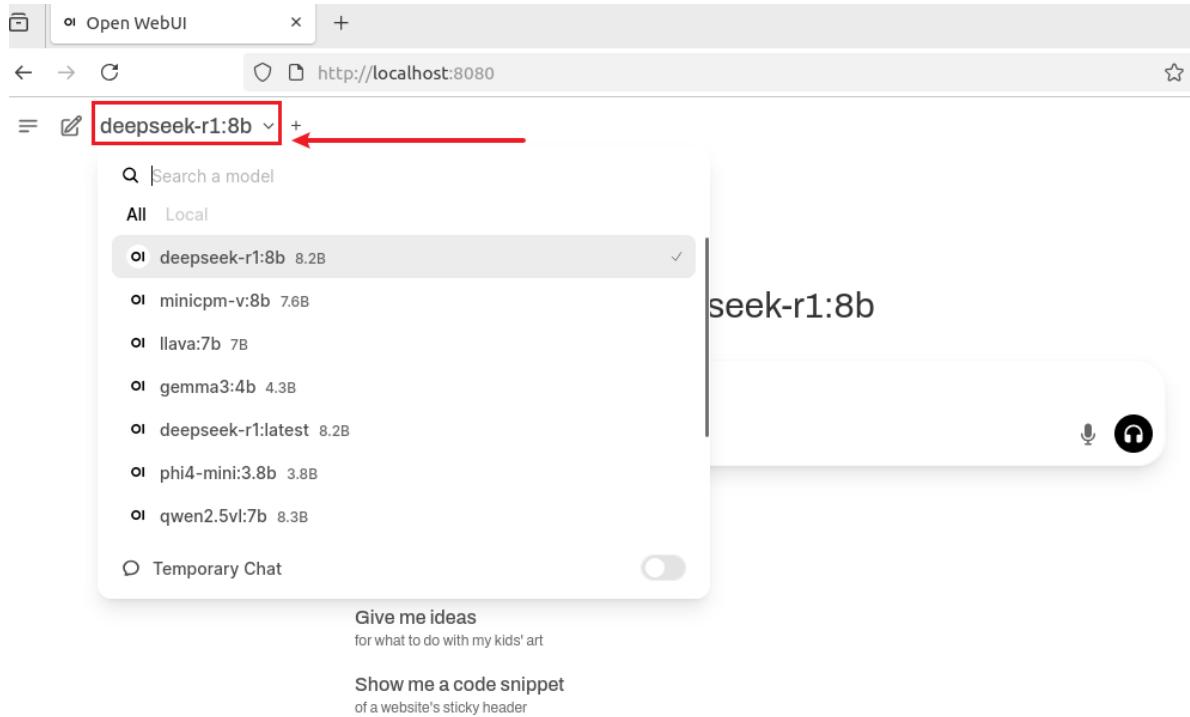
Using the Open WebUI for dialogue will be slower than using the Ollam tool directly, and may even result in timeouts and connection failures. This is related to the Jetson motherboard's memory and is unavoidable!

Users interested can switch to other Linux environments to set up the ollam and Open WebUI tools for interaction.

5.1 Switching Models

Clicking `Select a model` in the upper left corner allows you to select a specific model for interaction.

Models retrieved using Ollam are automatically added to the Open WebUI model options. Refreshing the webpage will display the new model!



5.2 Demo Model: minicpm-v:8b

What do you see in this picture?



What do you see in this picture?

Ol minicpm-v:8b

I see a man wearing two different types of sleeveless shirts, one white tank top and another mesh-like shirt. He is sitting on what appears to be an elevated platform or rooftop with his eyes closed, possibly enjoying the scenery or taking a break from some physical activity. Another person can be seen in the background, also dressed casually and squatting near objects like bottles of water. The sky above them has clouds against a blue backdrop, suggesting it might be daytime.

✍ ⌂ ⏪ ⏴ ⏵ ⏶ ⏷ ⏸

追问

↳ What kind of activities do people usually do on rooftops?



↳ Do you see any other notable features in the image?

输入消息

+ 代码解释器



6. Frequently Asked Questions

6.1 Disabling Open WebUI

Disable the automatically starting Open WebUI.

- View running Docker containers

```
docker ps
```

- Shut down running Docker containers

```
docker stop [CONTAINER ID] # Example docker stop 5f42ee9cf784
```

- View stopped containers

```
docker ps -a
```

- Clean up stopped containers

```
docker rm [CONTAINER ID] # Example docker rm 5f42ee9cf784
```

Clean up all stopped containers:

```
docker container prune
```

6.2 Common Errors

Unable to start Open WebUI

- docker: Error response from daemon: Conflict. The container name "/open-webui" is already in use by container
"cfc05c84f8e38b290337e7178c76fd1c49076f94b11ed3d49d9448be72b7f20f". You have to remove (or rename) that container to reuse that name.

Solution: Close and restart Open WebUI!

Service Connection Timeout

- Open WebUI: Server Connection Error

Close and restart Open WebUI, then try asking the question again, or use the Ollam tool to run the model and ask the question!

6.3 Insufficient Memory Error

When using a model with a large data volume to generate queries, the following error occurs:

- 500: model requires more system memory (6.6 GiB) than is available (5.2 GiB)
 1. It is recommended to use a computer browser accessing the motherboard via IP:8080. Accessing the model locally on the motherboard requires some memory, and running a 7B or 8B model under these conditions will result in an insufficient memory error and the model will fail to run.
 2. Running Open WebUI also consumes some memory, which may prevent some larger models from running, such as Ollama. Instead of running it directly, consider pulling a larger model with a smaller dataset, or migrating it to Ollama to run the larger model.
 3. Increase swap space content.

```
# Create a 4GB swap file
sudo fallocate -l 4G /swapfile
sudo chmod 600 /swapfile
sudo mkswap /swapfile
sudo swapon /swapfile
```