# Ollama Open-Webui

Notes only for now:

https://github.com/open-webui/open-webui
https://docs.openwebui.com/getting-started/env-configuration/#general

Nvidia GPU, linux:

```
mkdir /opt/ollama
mkdir /opt/open-webui
mkdir /opt/openedai-speech/tts-voices
mkdir /opt/openedai-speech/tts-config
mkdir /opt/pipelines
mkdir /opt/whisper
```

docker-ollama.yml

```
services:
  ollama:
    image: ollama/ollama:latest
    container_name: ollama
    volumes:
      - /opt/ollama:/root/.ollama
    ports:
      - 11434:11434
    #runtime: nvidia
    deploy:
      resources:
        reservations:
          devices:
          - driver: nvidia
            device_ids: ['0']
            capabilities: [gpu]
```

docker-openwebui.yml

```
services:
  open-webui:
    image: ghcr.io/open-webui/open-webui:main
    container_name: open-webui
    deploy:
      resources:
        reservations:
          devices:
          - driver: nvidia
            device_ids: ['0']
            capabilities: [gpu]
```

```
    ports:
      - 3000:8080
    volumes:
      - /opt/open-webui:/app/backend/data
    restart: always
    extra_hosts:
      host.docker.internal: host-gateway
    environment:
      - WEBUI_NAME="CustomGPT"
```

docker-openedai-speech.yml

```
services:
  openedai-speech:
    image: ghcr.io/matatonic/openedai-speech
    container_name: openedai-speech
    ports:
      - "8060:8000"
    volumes:
      - /opt/openedai-speech/tts-voices:/app/voices
      - /opt/openedai-speech/tts-config:/app/config
    restart: unless-stopped
    deploy:
      resources:
        reservations:
          devices:
          - driver: nvidia
            device_ids: ['0']
            capabilities: [gpu]
    environment:
      - TZ=Europe/London
      - TTS_HOME=voices
      - HF_HOME=voices
      # - PRELOAD_MODEL=xtts
      # - PRELOAD_MODEL=xtts_v2.0.2
      # - PRELOAD_MODEL=parler-tts/parler_tts_mini_v0.1
```

In open-webui under settings → audio

1. set to openai
2. API Base URL: http://host.docker.internal:8060/v1
3. API Key: sk-111111111 (note: this is a dummy API key, no key required)
4. Under TTS Voice within the same audio settings menu in the admin panel, you can set the TTS Model to use from the following choices below that openedai-speech supports. The voices of these models are optimized for the English language.
   tts-1 or tts-1-hd: alloy, echo, echo-alt, fable, onyx, nova, and shimmer (tts-1-hd is configurable; uses OpenAI samples by default)

docker-pipelines.yml

```
services:
  pipelines:
    image: ghcr.io/open-webui/pipelines:main
    container_name: pipelines
    volumes:
      - /opt/pipelines:/app/pipelines
    ports:
      - 9099:9099
    restart: always
    extra_hosts:
      host.docker.internal: host-gateway
    deploy:
      resources:
        reservations:
          devices:
          - driver: nvidia
            device_ids: ['0']
            capabilities: [gpu]
```

Under settings→connections set:

1. OPENAI API host: http://host.docker.internal:9099
2. OPENAI API Key: 0p3n-w3bu!

docker-whisper.yml

```
services:
  faster-whisper:
    image: lscr.io/linuxserver/faster-whisper:gpu
    container_name: faster-whisper
    environment:
      - PUID=1000
      - PGID=1000
      - TZ=Europe/London
      - WHISPER_MODEL=tiny-int8
      - WHISPER_BEAM=1 #optional
      - WHISPER_LANG=en #optional
    volumes:
      - /opt/whisper:/config
    ports:
      - 10300:10300
    restart: unless-stopped
    deploy:
      resources:
        reservations:
          devices:
          - driver: nvidia
            device_ids: ['0']
            capabilities: [gpu]
```

go to settings → audio, set

1. OPENAI API host: http://host.docker.internal:10300/v1
2. OPENAI API Key: sk-something
3. model: hasspy:faster-whisper-tiny-int8

possible alternative: https://github.com/fedirz/faster-whisper-server

AMD GPU on Windows:

docker-ollama.yml

```
name: ollama
services:
  ollama:
    image: ollama/ollama:rocm
    container_name: ollama
    volumes:
      - /p/Docker_Volumes/ollama:/root/.ollama
    ports:
      - 11434:11434
    deploy:
      resources:
        reservations:
          devices:
            - capabilities: [gpu]
```

docker-openwebui.yml

```
name: webui
services:
  open-webui:
    image: ghcr.io/open-webui/open-webui:main
    container_name: open-webui
    deploy:
      resources:
        reservations:
          devices:
          - capabilities: [gpu]
    ports:
      - 3000:8080
    volumes:
      - /p/Docker_Volumes/openwebui:/app/backend/data
    restart: always
    extra_hosts:
      host.docker.internal: host-gateway
    environment:
      - WEBUI_AUTH=false
```

docker install - WSL2 backend cmd line

```
docker compose -f docker-openwebui.yml up -d
docker compose -f docker-ollama.yml up -d
```

mkdir p/Docker_Volumes = P:\Docker_Volumes