Open WebUI Installation Guide for Windows 11

Overview

This guide documents the complete installation process for setting up Open WebUI as a ChatGPT-style interface to your local Ollama server, with remote access via Cloudflare Tunnel.

Prerequisites

- Windows 11 machine
- Ollama already installed and running (e.g., at `localhost:11434`)
- Cloudflare tunnel already configured
- Administrative access to Windows

Step 1: Install Python 3.11 (Side-by-side with existing Python)

Why Python 3.11?

Open WebUI requires Python 3.11 or 3.12. If you have Python 3.13 (or any other version), install 3.11 alongside it without interfering with existing Python scripts.

Installation Steps

1. Download Python 3.11.x from python.org/downloads
2. During installation:
   - ✅**CHECK** "Install launcher for all users"
   - ❌**UNCHECK** "Add python.exe to PATH" (critical!)
   - Click "Customize installation"
   - Check all optional features
   - On "Advanced Options":
     - Choose custom location: `C:\Python311\` or `D:\Misc\Python311\`
     - **Uncheck** "Add Python to environment variables"
   - Complete installation
3. Verify both Python versions work:

```cmd
# Your default Python (3.13 or whatever you have)
python --version

# Python 3.11 specifically
py -3.11 --version

# Or use full path
C:\Python311\python.exe --version
```

Step 2: Install Open WebUI

Installation Command

```cmd
py -3.11 -m pip install open-webui
```

Set Environment Variable

```cmd
setx OLLAMA_BASE_URL "http://localhost:11434"
```

Note: Close and reopen Command Prompt after setting environment variable.

Step 3: Test Open WebUI Locally

Start Open WebUI Manually

```cmd
# Using Python launcher
py -3.11 -m open-webui serve --host 0.0.0.0 --port 8080

# Or using full path
D:\Misc\Python311\Scripts\open-webui.exe serve --host 0.0.0.0 --port 8080
```

Verify Local Access

1. Open browser to `http://localhost:8080`
2. Create your first account (becomes admin)
3. Verify you can see your Ollama models
4. Test a chat message

Keep this window open for now

Step 4: Configure Cloudflare Tunnel

Add DNS Record in Cloudflare

1. Go to Cloudflare Dashboard
2. Select your domain (e.g., `ldmathes.cc`)
3. Click **DNS** in left menu
4. Click **Add record**
5. Configure:
   - **Type:** `CNAME`
   - **Name:** `chat`
   - **Target:** `[your-tunnel-id].cfargotunnel.com`
   - **Proxy status:** Proxied (orange cloud ON)
   - **TTL:** Auto
6. Click **Save**

Update Tunnel Config File

Location: `C:\Users\[YourUsername]\.cloudflared\config.yml`

Add the chat hostname to your ingress rules:

yaml

```yaml
tunnel: [your-tunnel-id]
credentials-file: C:\Users\[YourUsername]\.cloudflared\[your-tunnel-id].json
protocol: http2

ingress:
  # Your existing hostnames...
  - hostname: ollama.yourdomain.com
    service: http://localhost:11434
    originRequest:
      httpHostHeader: "localhost:11434"

  # Add this for Open WebUI
  - hostname: chat.yourdomain.com
    service: http://localhost:8080

  # Catch-all rule (must be last)
  - service: http_status:404
```

Validate and Restart Tunnel

cmd

```cmd
# Validate configuration
cloudflared tunnel ingress validate

# Restart cloudflared service
net stop cloudflared
net start cloudflared
```

Or if running manually:

cmd

```cmd
taskkill /F /IM cloudflared.exe
cloudflared tunnel run [your-tunnel-id]
```

Step 5: Test Remote Access

Wait for DNS Propagation

DNS should propagate in 2-10 minutes. Test with:

cmd

```cmd
# From any machine
nslookup chat.yourdomain.com 8.8.8.8
```

You should see Cloudflare IPs (104.x.x.x or 172.x.x.x)

Access Your Chat Interface

Visit `https://chat.yourdomain.com` from any device

Troubleshooting DNS Cache

If it doesn't work on a specific device:

Windows:

```cmd
ipconfig /flushdns
```

macOS:

```bash
sudo dscacheutil -flushcache; sudo killall -HUP mDNSResponder
```

Or: Cycle VPN connection if using one

Step 6: Set Up as Windows Service (Auto-start)

Download NSSM

1. Download from nssm.cc/download
2. Extract to a permanent location (e.g., `D:\Misc\nssm.exe`)

Install Service

Open Command Prompt **as Administrator**:

```cmd
# If you get "service already exists" error, delete it first:
sc delete OpenWebUI

# Wait 10-30 seconds, then install:
d:\Misc\nssm.exe install OpenWebUI2 "D:\Misc\Python311\Scripts\open-webui.exe" "serve --host 0.0.0.0 --port 8080"

d:\Misc\nssm.exe set OpenWebUI2 AppDirectory "D:\Misc\Python311\Scripts"

d:\Misc\nssm.exe set OpenWebUI2 AppEnvironmentExtra OLLAMA_BASE_URL=http://localhost:11434

d:\Misc\nssm.exe set OpenWebUI2 AppStdout "D:\Misc\openwebui-stdout.log"

d:\Misc\nssm.exe set OpenWebUI2 AppStderr "D:\Misc\openwebui-stderr.log"

d:\Misc\nssm.exe start OpenWebUI2
```

Verify Service is Running

```cmd
d:\Misc\nssm.exe status OpenWebUI2

# Should show: SERVICE_RUNNING
```

Test Remote Access Again

Visit `https://chat.yourdomain.com` - should work from all devices now

Installation Complete! ✅

You now have:

- ✅Python 3.11 installed alongside your existing Python
- ✅Open WebUI installed and running as a Windows service
- ✅ChatGPT-style interface at `https://chat.yourdomain.com`
- ✅Accessible from any device (desktop, mobile, tablet)
- ✅Auto-starts when Windows boots

Post-Installation Configuration

First-Time Setup

1. Visit `https://chat.yourdomain.com`
2. Create admin account (first user becomes admin)
3. Configure settings:
   - Default model selection
   - User registration (enable/disable)
   - Authentication requirements
   - Interface preferences

Admin Panel Access

- Click profile icon → **Admin Panel**
- Manage users, models, and system settings

Key File Locations

Item	Location
Python 3.11	`D:\Misc\Python311\`
Open WebUI executable	`D:\Misc\Python311\Scripts\open-webui.exe`
Database & data	`D:\Misc\Python311\Lib\site-packages\open_webui\data\`
Service logs	`D:\Misc\openwebui-stdout.log` and `openwebui-stderr.log`
Cloudflare config	`C:\Users\[Username]\.cloudflared\config.yml`
NSSM	`D:\Misc\nssm.exe`

Common Installation Issues

Issue: Python version conflict

Solution: Install Python 3.11 to custom directory, don't add to PATH, use `py -3.11` explicitly

Issue: "Service marked for deletion"

Solution: Run `sc delete OpenWebUI`, wait 30 seconds, try again

Issue: 502 Bad Gateway after service install

Solution: Check logs at `D:\Misc\openwebui-stderr.log`, verify service is running with `nssm status OpenWebUI2`

Issue: Models not appearing

Solution: Verify Ollama is running, check `OLLAMA_BASE_URL` environment variable, restart service

Issue: DNS not resolving

Solution: Wait 5-10 minutes, test with `nslookup chat.yourdomain.com 8.8.8.8`, flush DNS cache

Updating Open WebUI

When updates are available:

```cmd
# Stop service
d:\Misc\nssm.exe stop OpenWebUI2

# Update Open WebUI
py -3.11 -m pip install --upgrade open-webui

# Start service
d:\Misc\nssm.exe start OpenWebUI2
```

Alternative: Manual Startup (Without Service)

If you prefer not to use NSSM, create a batch file:

File: `start-openwebui.bat`

```batch
@echo off
set OLLAMA_BASE_URL=http://localhost:11434
D:\Misc\Python311\Scripts\open-webui.exe serve --host 0.0.0.0 --port 8080
pause
```

Place in Windows Startup folder: `C:\Users\[Username]\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup`

Architecture Diagram

```
User Device (Browser)
   ↓
https://chat.yourdomain.com
   ↓
Cloudflare Edge Servers (SSL, DDoS protection)
   ↓
Cloudflare Tunnel (encrypted)
   ↓
Windows 11 Machine
   ├── localhost:8080 (Open WebUI - Python 3.11)
   └── localhost:11434 (Ollama AI Models)
```

Security Notes

- First user registered becomes admin automatically
- Consider disabling public registration after creating accounts
- All traffic encrypted via Cloudflare SSL
- Cloudflare provides DDoS protection
- No ports exposed directly to internet (tunnel only)
- Keep Open WebUI and Ollama updated regularly

Next Steps

1. Explore Open WebUI features (document upload, multi-model chat, etc.)
2. Download additional Ollama models as needed
3. Configure user preferences and default settings
4. Set up regular backups of the database
5. Share access with team members if desired

Support Resources

- **Open WebUI Docs:** https://docs.openwebui.com
- **Ollama Docs:** https://ollama.ai/docs
- **Cloudflare Tunnel Docs:** https://developers.cloudflare.com/cloudflare-one/
- **Python Launcher:** https://docs.python.org/3/using/windows.html#launcher