

Installation avec pip

Source : [GitHub - open-webui/open-webui: User-friendly AI Interface \(Supports Ollama, OpenAI API, ...\)](#)

ATTENTION !

L'installation avec `pip` rend la mise à jour du logiciel complexe. Faites cette installation pour des tests, ou du développement spécifique et non une utilisation en production.

Installation via Python pip

Open WebUI peut être installé via pip, l'installateur de paquets Python. Avant de procéder, assurez-vous d'utiliser Python 3.11 pour éviter les problèmes de compatibilité.

Installer Open WebUI :

Ouvrez votre terminal et exécutez la commande suivante pour installer Open WebUI :

```
pip install open-webui
```

Exécuter Open WebUI :

Une fois l'installation terminée, vous pouvez démarrer Open WebUI en exécutant la commande suivante :

```
open-webui serve
```

Cela démarrera le serveur Open WebUI, que vous pourrez accéder à l'adresse suivante :

<http://localhost:8080>.

Erreur possible :

Erreur lors de l'installation de `open-webui` via `pip` :

```
root@srv-ia:/home/ia# pip install open-webui
error: externally-managed-environment

× This environment is externally managed
└─> To install Python packages system-wide, try apt install
```

python3-xyz, where xyz is the package you are trying to install.

If you wish to install a non-Debian-packaged Python package, create a virtual environment using `python3 -m venv path/to/venv`. Then use `path/to/venv/bin/python` and `path/to/venv/bin/pip`. Make sure you have python3-full installed.

If you wish to install a non-Debian packaged Python application, it may be easiest to use `pipx install xyz`, which will manage a virtual environment for you. Make sure you have pipx installed.

See `/usr/share/doc/python3.11/README.venv` for more information.

note: If you believe this is a mistake, please contact your Python installation or OS distribution provider. You can override this, at the risk of breaking your Python installation or OS, by passing `--break-system-packages`.

hint: See PEP 668 for the detailed specification.

Utiliser un environnement virtuel

La meilleure pratique pour installer des paquets Python sans affecter l'environnement système est de créer un **environnement virtuel**. Cela permet d'avoir une installation isolée pour vos paquets Python.

Étapes pour créer un environnement virtuel :

1. **Installez** `python3-venv` (si ce n'est pas déjà fait) :

```
sudo apt install python3-venv
```

2. **Créez un environnement virtuel :**

Dans le répertoire où vous souhaitez installer vos paquets, créez un environnement virtuel :

```
python3 -m venv mon_venv
```

Cette commande crée un dossier `mon_venv` qui contient une installation isolée de Python.

3. **Activez l'environnement virtuel :**

```
source mon_venv/bin/activate
```

Vous verrez probablement un changement dans l'invite de commande, indiquant que l'environnement virtuel est activé (par exemple, `(mon_venv)` au début de la ligne).

4. Installez `open-webui` dans l'environnement virtuel :

Une fois l'environnement activé, vous pouvez installer `open-webui` comme d'habitude :

```
pip install open-webui
```

5. Lancez l'application avec `nohup` et redirigez la sortie vers un fichier de log, puis lancez-le en arrière-plan :

```
nohup open-webui serve > open-webui.log 2>&1 &
```

- `nohup` permet au processus de ne pas être tué lorsqu'une session se termine.
- `> open-webui.log 2>&1` redirige les sorties standard (stdout) et les erreurs (stderr) vers un fichier `open-webui.log`.
- `&` place le processus en arrière-plan.

Vous pouvez quitter la session SSH ou votre terminal normalement, l'application continuera de fonctionner.

Pour voir si le processus est toujours en cours d'exécution :

```
ps aux | grep open-webui
```

6. Désactivez l'environnement virtuel quand vous avez terminé :

```
deactivate
```

Cela vous ramènera à votre environnement système Python.

Revision #5

Created 29 November 2024 08:49:14 by Maxime

Updated 3 December 2024 09:43:20 by Maxime