GUIDE D'INSTALLATION

Intelligence Artificielle Locale
(Linux)

Ollama + AnythingLLM

Service Écoles-Médias (SEM) – Genève
Janvier 2026

1.Installation d’Ollama

Étape 1 – Télécharger Ollama

1. Ouvrez votre navigateur web.

Allez sur l’adresse suivante :
`https://ollama.com/download`

2. Cliquez sur **Linux** pour obtenir la commande d’installation.

Étape 2 – Lancer l’installation

1. Ouvrez un terminal sous Linux.
2. Collez la commande fournie par le site (généralement) :

```
curl -fsSL https://ollama.com/install.sh | sh
```

3. Appuyez sur **Entrée** et laissez le script installer Ollama.

Étape 3 – Vérifier l’installation

Dans le terminal, tapez :

```
ollama --version
```

Si une version s’affiche (par exemple `ollama version x.x.x`), alors Ollama est correctement installé.

2. Téléchargement du modèle IA

Nous allons installer **Mistral 7B**, un excellent modèle léger et performant.

Étape 1 – Ouvrir le terminal

Ouvrez une fenêtre de terminal sous Linux.

Étape 2 – Télécharger le modèle

Tapez la commande suivante :

```
ollama pull mistral
```

Ollama télécharge automatiquement le modèle.
Cela peut prendre quelques minutes selon votre connexion.

Étape 3 – Tester le modèle

Une fois le téléchargement terminé, lancez le modèle :

```
ollama run mistral
//
ollama run llama3.2:1b
```

Vous verrez apparaître un prompt :

```
>>>
```

Tapez une question, par exemple :

```
>>> Explique-moi ce qu'est Docker en quelques mots.
```

Puis appuyez sur **Entrée**.
Le modèle vous répond directement dans le terminal.

Pour quitter le modèle :

- Appuyez sur `Ctrl + D`, ou
- Tapez :

```
/exit
```

3. Installation d'AnythingLLM sur Linux

Étape 1 – Télécharger le script d'installation

Ouvrez votre navigateur et allez sur :
`https://anythingllm.com/download`

- Cliquez sur **Download for Linux**.
- Cela vous redirigera vers GitHub pour récupérer le script d'installation.

Étape 2 – Télécharger et rendre le script exécutable

Ouvrez un terminal et tapez :

```
# Télécharger le script
curl -fsSL https://cdn.anythingllm.com/latest/installer.sh -o installer.sh

# Rendre le script exécutable
chmod +x installer.sh
```

Étape 3 – Exécuter le script d'installation

`./installer.sh`

- Le script détecte automatiquement l'architecture de votre système (`aarch64` pour ARM64 comme Mac M2, `x86_64` pour Intel/AMD).
- Il télécharge la **AppImage correspondante** dans votre répertoire `$HOME`.
- Il propose de créer automatiquement un **profil AppArmor** pour permettre à AnythingLLM de s'exécuter sans problèmes de permissions.

Le terminal affiche :

```
AppArmor is enabled on this system.

[Warning] You will get an error about SUID permission issues when running the AppImage without creating an AppArmor profile.

This requires sudo privileges. If you are unsure, you can create an AppArmor profile manually.

Do you want to auto-create an AppArmor profile for AnythingLLM now? (y/n):
```

➡️ vous devez répondre :

```
y
```

Puis le système demande le mot de passe :

```
[sudo] password for user:
```

Vous entrez le mot de passe Linux et validez

Qu'avez-vous fait ? :

Cette étape autorise AnythingLLM une seule fois dans le système de sécurité AppArmor afin qu'il puisse s'exécuter normalement, sans blocages ni demandes répétées de droits administrateur.

- Enfin, il crée un fichier `.desktop` pour lancer l'application depuis le menu graphique.

Lancer l'application
Après installation et création du profil AppArmor :

```
cd ~
./AnythingLLMDesktop.AppImage
```

Petites remarques et solutions aux problèmes fréquents

1ère. Erreur `libz.so` manquant

Problème :

```
error while loading shared libraries: libz.so: cannot open
shared object file
```

- La AppImage a besoin de `libz.so`.
- Sur Ubuntu minimal ARM64, souvent seule la version “natif” est installée :

```
sudo apt install zlib1g
```

- Mais la AppImage cherche **spécifiquement `libz.so`**, ce qui nécessite le paquet de développement :

```
sudo apt install zlib1g-dev
```

Solution complète :

```
sudo apt update
sudo apt install -y zlib1g zlib1g-dev

# Vérifier que le lien symbolique existe
ls -l /usr/lib/aarch64-linux-gnu/libz.so
# Devrait pointer vers libz.so.1.x.x

# Si le lien n'existe pas :
sudo ln -s /usr/lib/aarch64-linux-gnu/libz.so.1
/usr/lib/aarch64-linux-gnu/libz.so

# Lancer AnythingLLM
cd ~
./AnythingLLMDesktop.AppImage
```

- Une fois le lien créé correctement, AnythingLLM devrait démarrer sans erreur liée à `libz.so`.

2ème. Erreur `libfuse.so.2` manquant

Problème :

```
dlopen(): error loading libfuse.so.2
AppImages require FUSE to run.
```

- Les AppImages utilisent **FUSE (Filesystem in Userspace)** pour se monter temporairement.
- Si `libfuse` n'est pas installé, l'AppImage ne peut pas s'exécuter.

Solution pour Ubuntu ARM64 :

```
sudo apt update
sudo apt install -y fuse libfuse2

# Vérifier que la bibliothèque existe
ls -l /usr/lib/aarch64-linux-gnu/libfuse.so.2
# Devrait montrer un lien vers libfuse.so.2.x.x

# Lancer AnythingLLM
cd ~
./AnythingLLMDesktop.AppImage
```

- Après cette étape, l'application devrait démarrer correctement sans erreurs de bibliothèques.