Builder

AI powered application building pipeline

Installation

From the root of the project execute:

```
./Scripts/install.sh
```

or

```
./Scripts/install.sh CATEGORY
```

where `CATEGORY` can be on of the following categories:

- `General`
- `Coder`
- `Tester`
- `Translation`
- `Generative/Animation`
- `Generative/Audio`
- `Generative/JPEG`
- `Generative/PNG`
- `Generative/SVG`

Note: If none has been passed to the script, the `General` category will be used.

Documentation

All documentation is located here milosvasic@thinker □ ~/…/HelixDevelopment/Builder □ □ main ● □ ./Scripts/install.sh Generative/Audio /usr/local/bin/ollama /usr/bin/docker 16171387e706 ghcr.io/open-webui/open-webui:main "bash start.sh" 2

weeks ago Up 51 minutes (healthy) 0.0.0.0:8085->8080/tcp, [::]:8085->8080/tcp open-webui Installation completed 🎵 Detected audio category, using specialized audio installer... 🎵 Audio Models Installation System

🔧 Checking system dependencies... 📦 Installing audio generation dependencies... ⚠ Using system Python with --break-system-packages flag 📦 Installing Python packages... ⚠ Virtual environment pip not found, using system pip 📦 Using basic pip flags (no system packages support) 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/python3.12/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. 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/python3.12/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. 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/python3.12/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. 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/python3.12/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. 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/python3.12/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. 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/python3.12/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. 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/python3.12/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. 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/python3.12/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. ✅ Audio dependencies installed 🔍 Detecting system capabilities... GPU VRAM: 6.0 GB 💻 Limited GPU/CPU setup. Using small audio models. ⚠ Using system Python for model installation

Models for installation:

Audio Generation Models - 7B Category

Format: model_name:type:repository_id

Types: musicgen, tts, bark

These models are installed via specialized audio framework

musicgen-small:musicgen:facebook/musicgen-small speech-t5:tts:microsoft/speecht5_tts bark-small:bark:suno/bark 2025-09-12 19:43:47 - Skipping: # Audio Generation Models - 7B Category 2025-09-

12 19:43:47 - Skipping: # Format: model_name:type:repository_id 2025-09-12 19:43:47 - Skipping: # Types: musicgen, tts, bark 2025-09-12 19:43:47 - Skipping: # These models are installed via specialized audio framework 2025-09-12 19:43:47 - Processing audio model: musicgen-small:musicgen:facebook/musicgen-small 🎼 Installing MusicGen model: musicgen-small (facebook/musicgen-small) Downloading MusicGen model: facebook/musicgen-small Fetching 1 files: 100%| [redacted] | 1/1 [00:00<00:00, 26379.27it/s] ✅ MusicGen model cached successfully ✅ Success: musicgen-small:musicgen:facebook/musicgen-small

2025-09-12 19:43:54 - Processing audio model: speech-t5:tts:microsoft/speecht5_tts 🗣 Installing TTS model: speech-t5 (microsoft/speecht5_tts) Downloading TTS model: microsoft/speecht5_tts × Error downloading model: SpeechT5Tokenizer requires the SentencePiece library but it was not found in your environment. Check out the instructions on the installation page of its repo: https://github.com/google/sentencepiece#installation and follow the ones that match your environment. Please note that you may need to restart your runtime after installation.

⚠ TTS model download failed, but continuing... ✅ Success: speech-t5:tts:microsoft/speecht5_tts

2025-09-12 19:43:58 - Processing audio model: bark-small:bark:suno/bark 🐕 Installing Bark model: bark-small (suno/bark) ✅ Bark already available 📥 Downloading Bark model components (this may take a while)... ⚠ Bark setup completed with warnings: 'str' object has no attribute '**module**' ✅ Success: bark-small:bark:suno/bark

== === AUDIO INSTALLATION SUMMARY: Total models processed: 3 Successfully installed: 3 Failed: 0

🎵 All audio models installed successfully! 🎵 Audio models installation completed! 📝 Creating audio model usage scripts... ✅ MusicGen script created from template ✅ TTS script created from template ✅ Usage scripts created in /home/milosvasic/Projects/HelixDevelopment/Builder/Scripts/../AudioModels/scripts/

📋 Usage Examples: Generate Music: python3 ../AudioModels/scripts/generate_music.py 'happy upbeat electronic music' Text to Speech: python3 ../AudioModels/scripts/text_to_speech.py 'Hello, this is a test'

🎯 Audio models are ready for use! 🎵 Audio models have been installed successfully ℹ Note: Audio models run independently of Ollama/WebUI 📋 Use the provided Python scripts to generate audio content