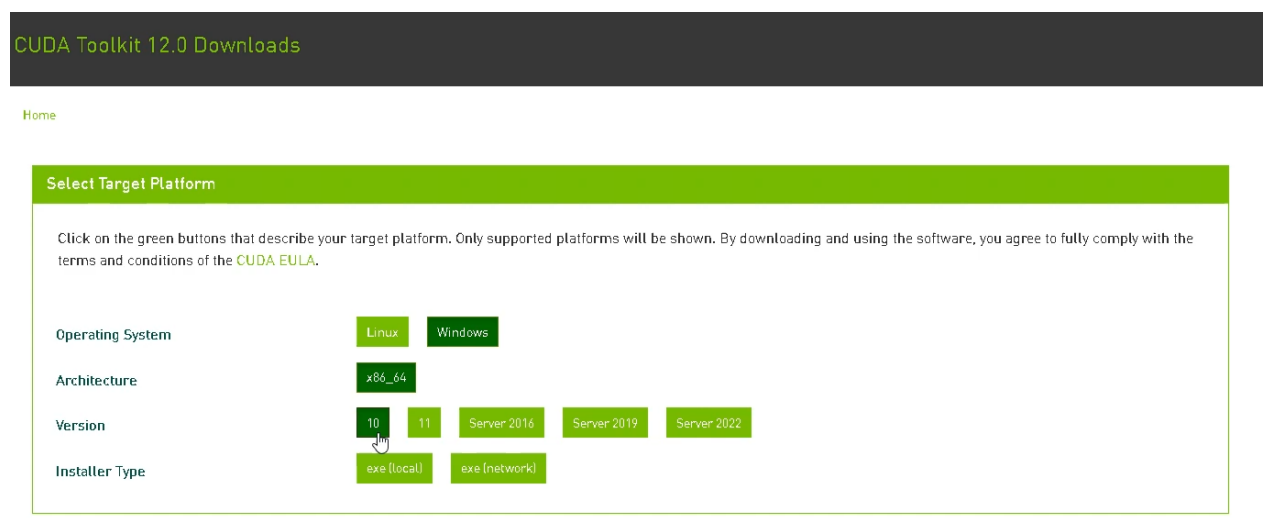


# Installing Whisper WebUI on Windows

## 10/11

### Install Nvidia CUDA Toolkit

First, go to <https://developer.nvidia.com/cuda-downloads> and select “Windows” next to “Operating System”, “x86\_64” next to “Architecture”, “10” or “11” next to “Version” and use the “exe (Network)” installer Type:



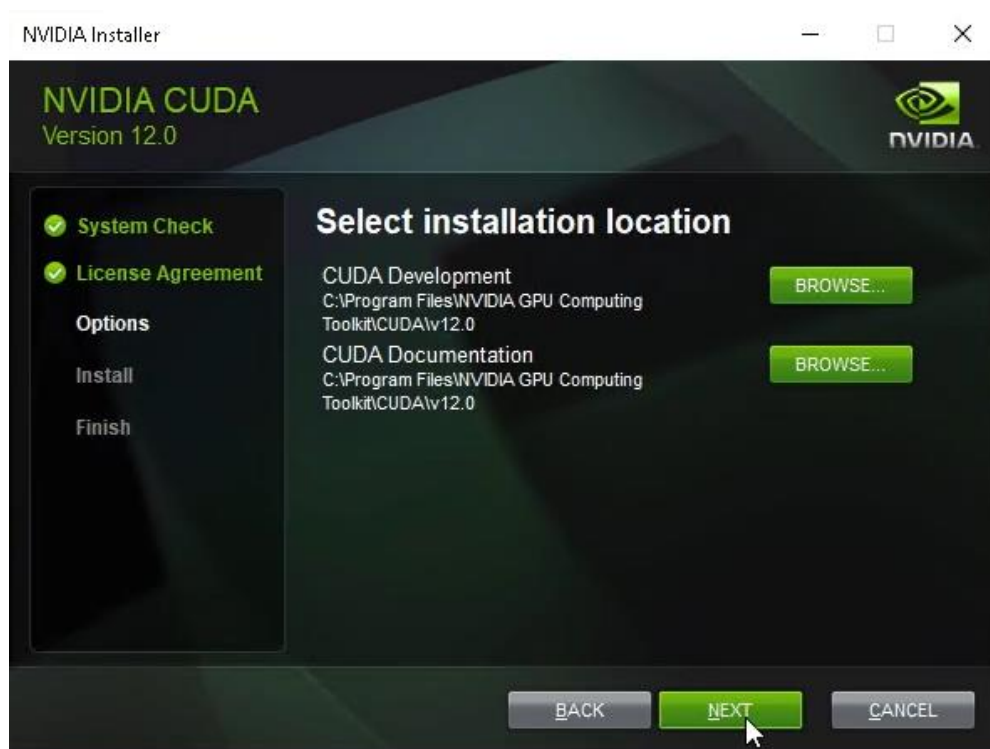
Let the installer Launch:



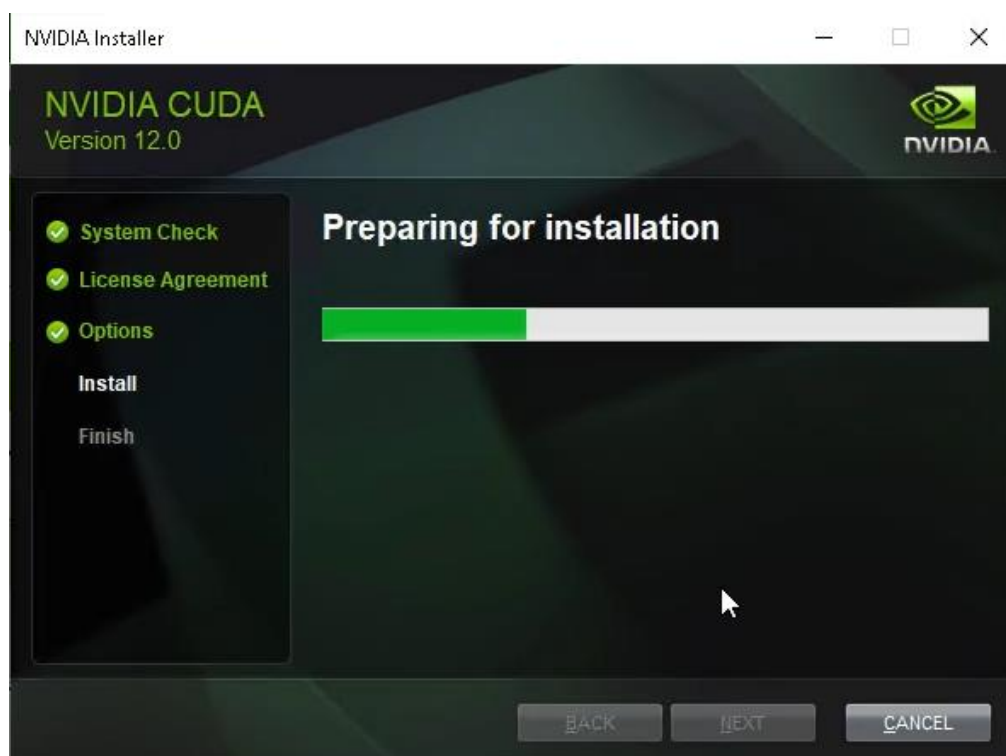
Now, either leave “Express (Recommended)” selected, or choose “Custom (Advanced)”:



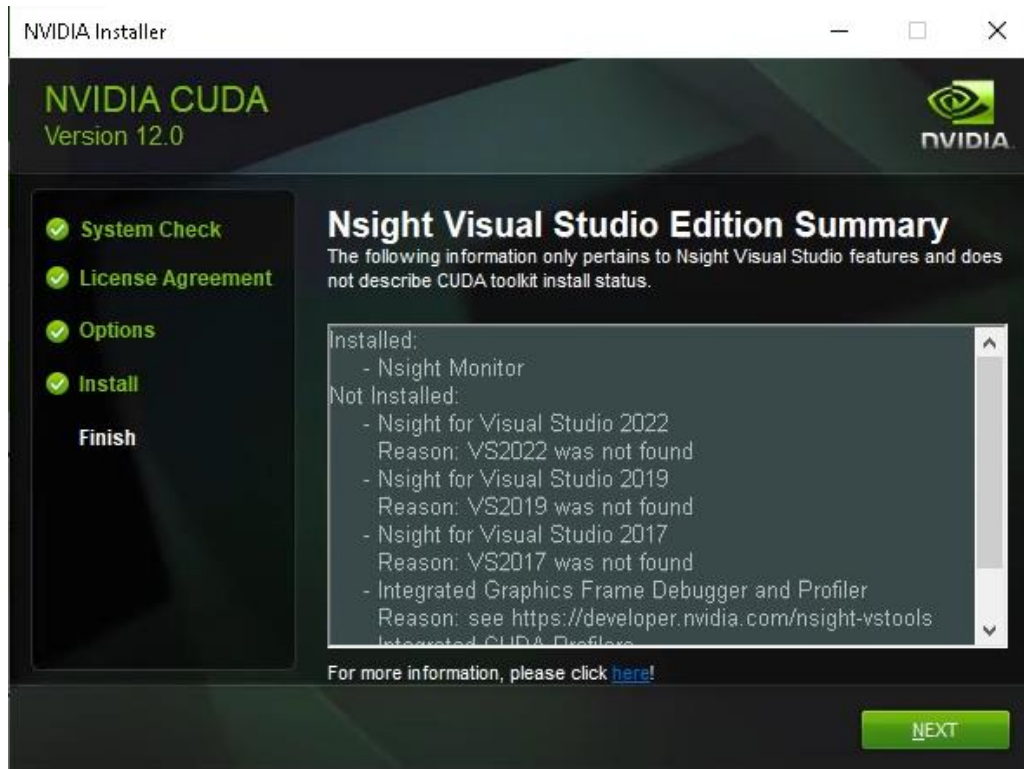
If you chose “Custom (Advanced)”, you may now customize the installation location. Then click “Next”



Wait for the installer to finish (this may take 10-15 minutes depending on your CPU and network speed):



Ignore “Nsight” if it pops up – just click “Next”:



The installer should then be finished – click “Close” (you may also want to deselect “Launch NVIDIA GeForce Experience”)



## Install Anaconda 3

Next, we will install the Python virtual environment manager Anaconda 3, to make managing the Python environment easier and without any conflicts.

Go to <https://www.anaconda.com/> and click “Download”:

# Data science technology for groundbreaking research.

Anaconda offers the easiest way to perform Python/R data science and machine learning on a single machine. Start working with thousands of open-source packages and libraries today.



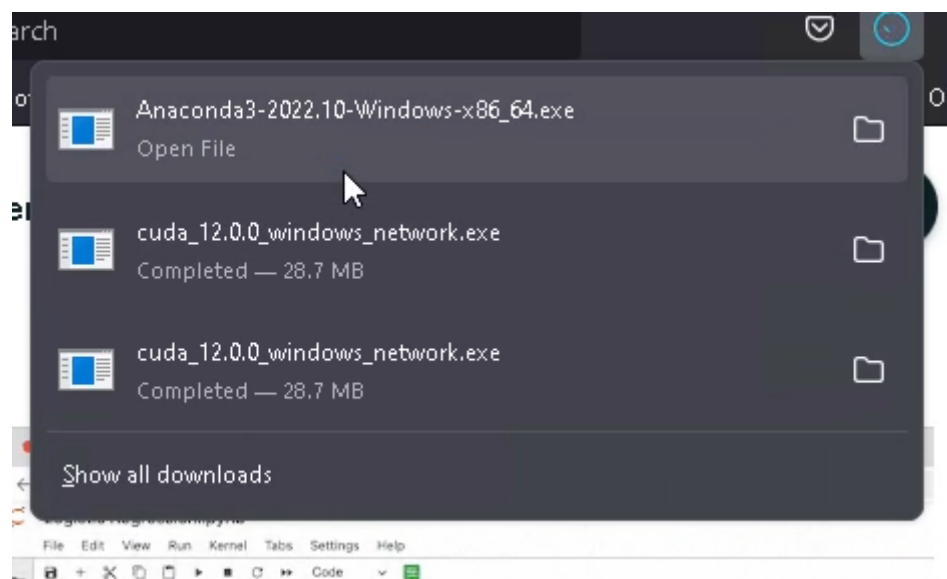
For Windows

Python 3.9 • 64-Bit Graphical Installer • 621 MB

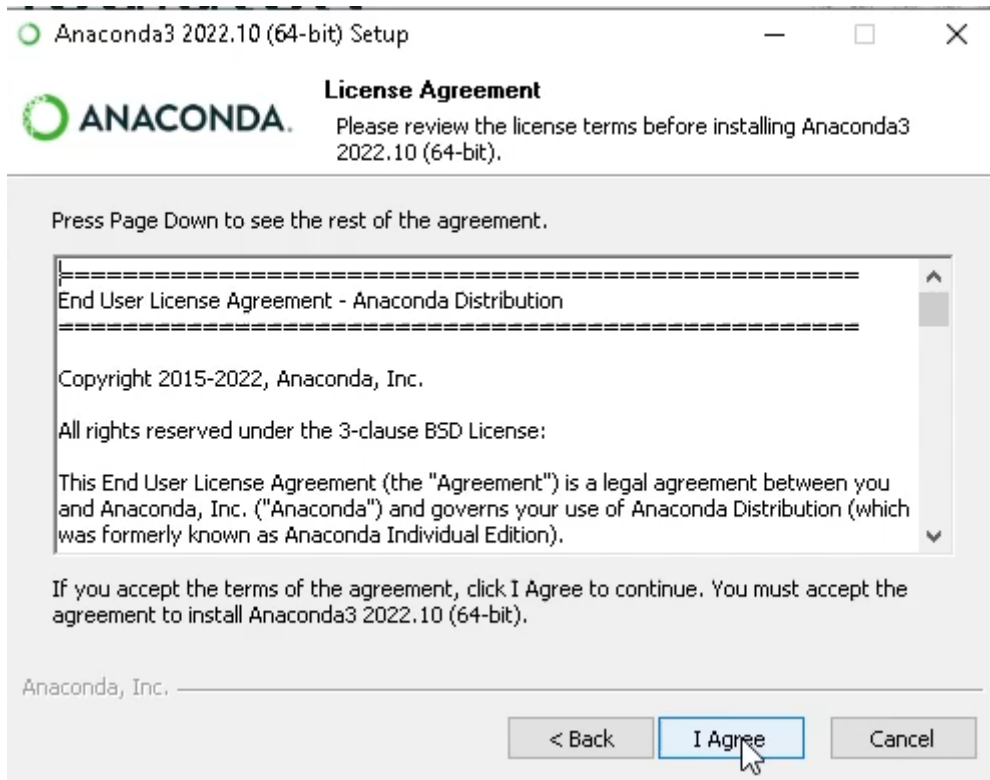
Get Additional Installers



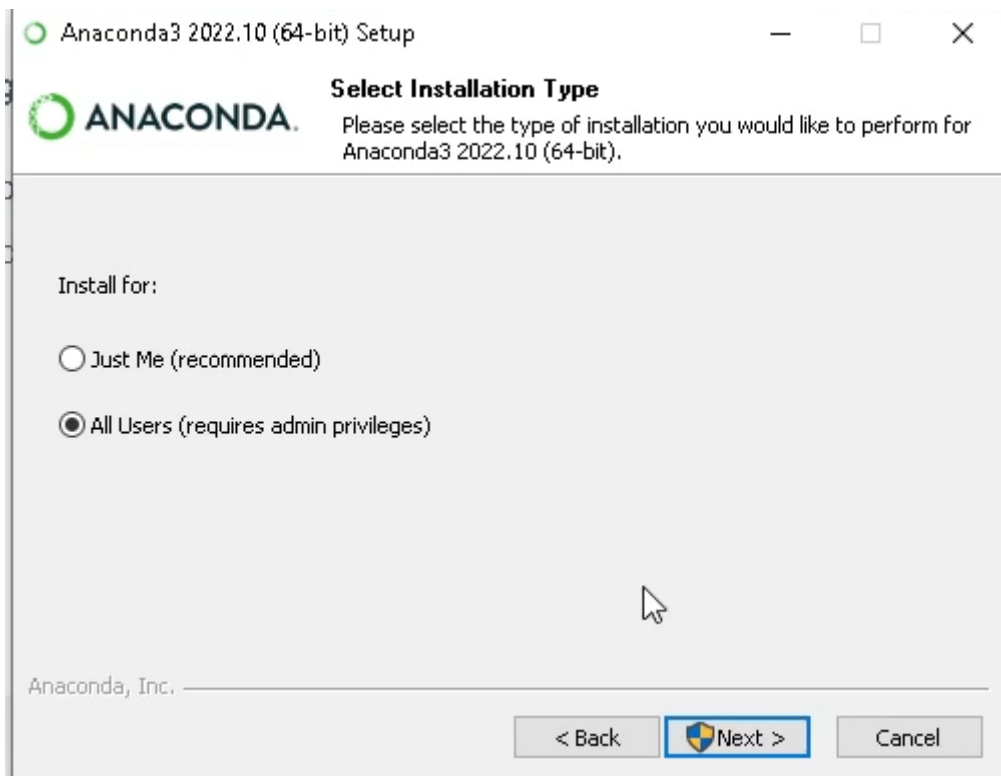
Open the downloaded EXE:



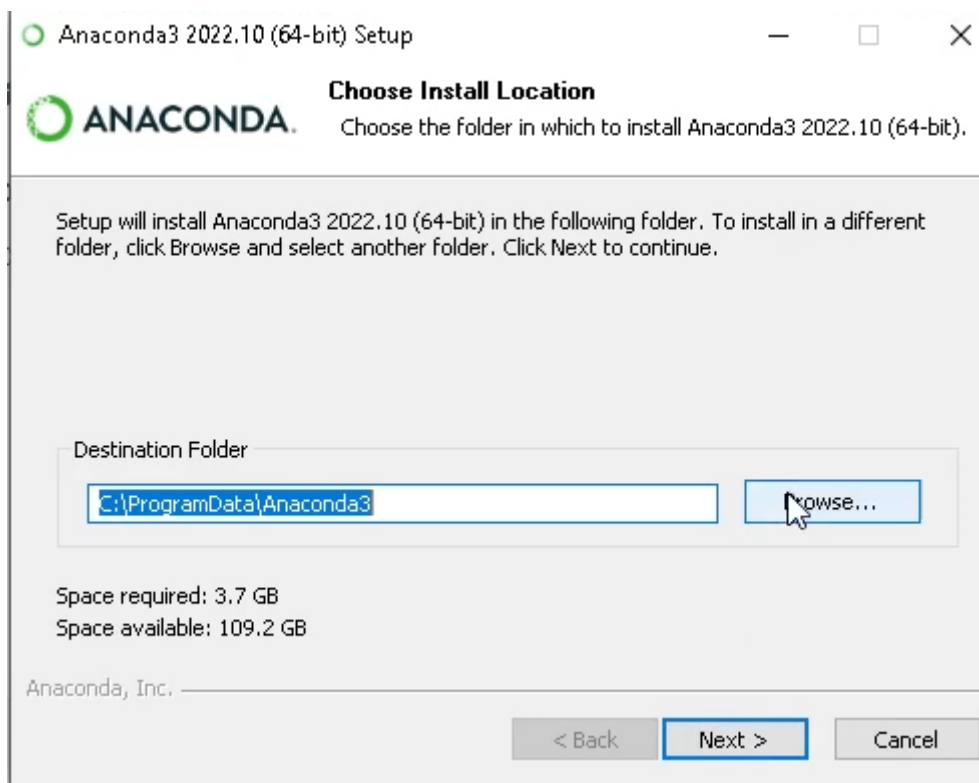
Agree to the EULA:



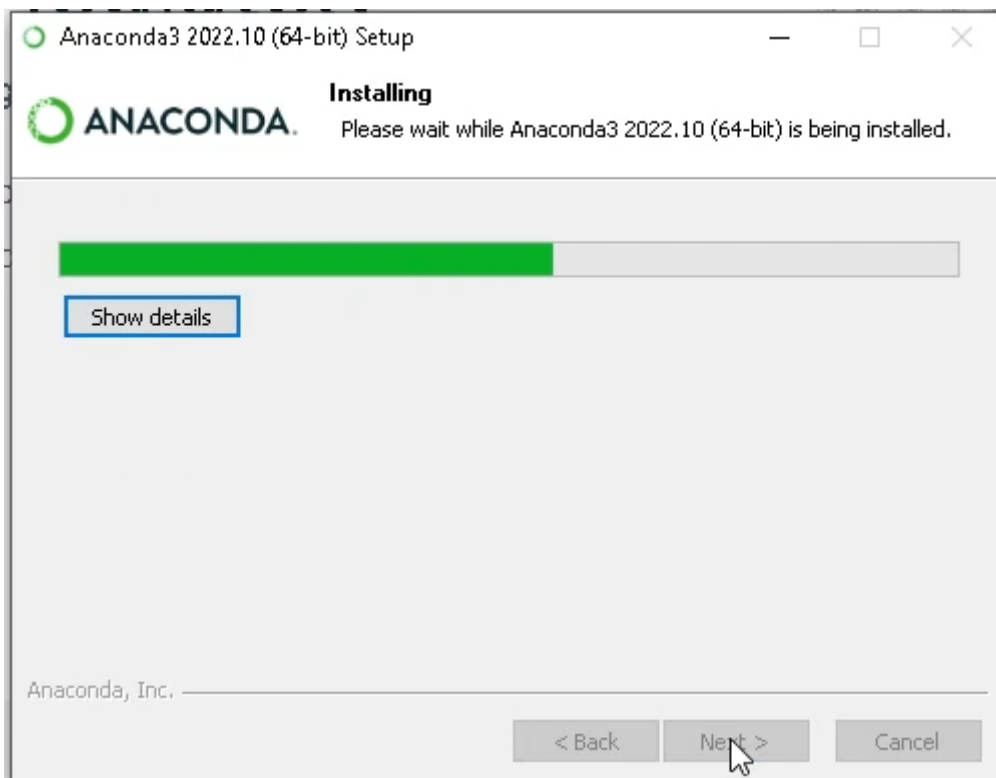
You may then either install Anaconda for the current user or all users (requires admin):



Customize the installation folder:

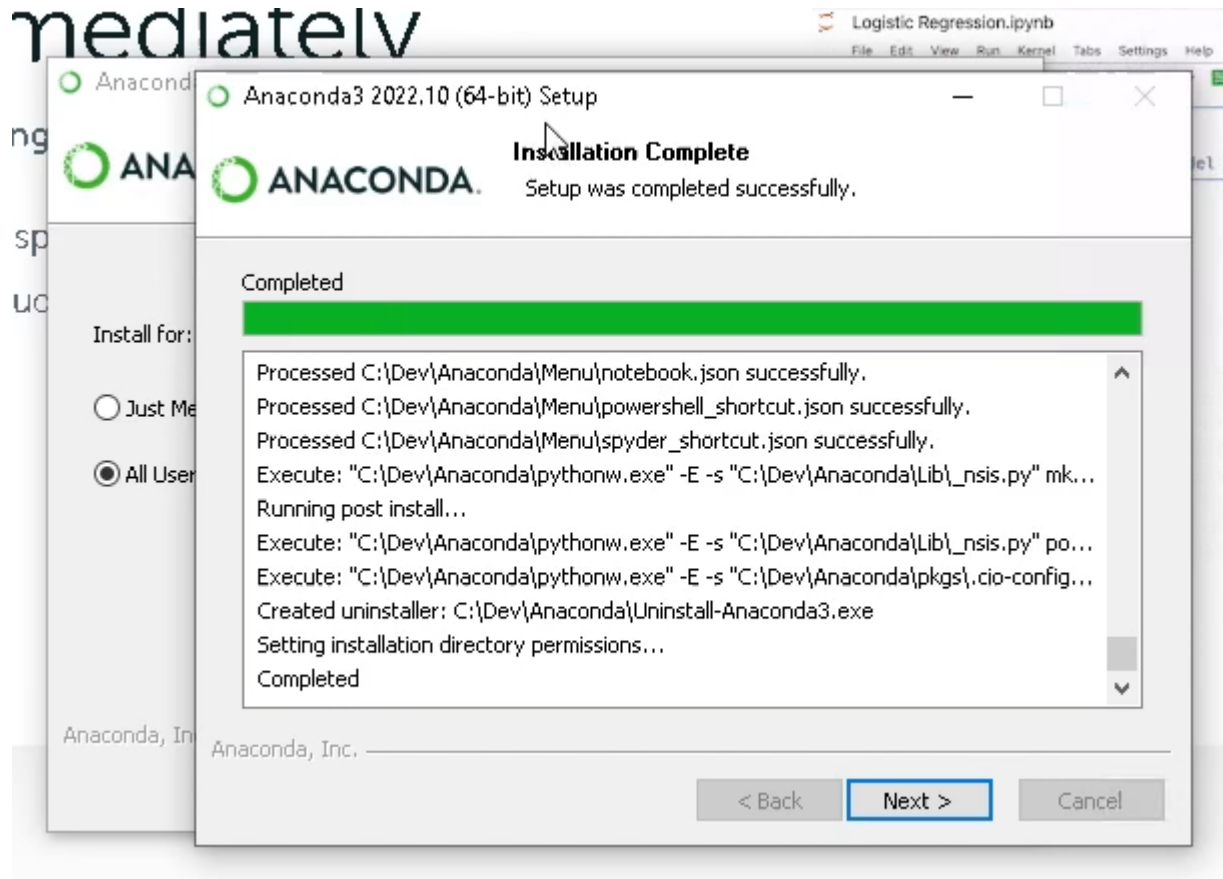


Then start the installation process:



This may take a couple of minutes. Then click Next:

mediately





Git for Windows

Then, install Git for Windows. Go to <https://gitforwindows.org/> and click Download:



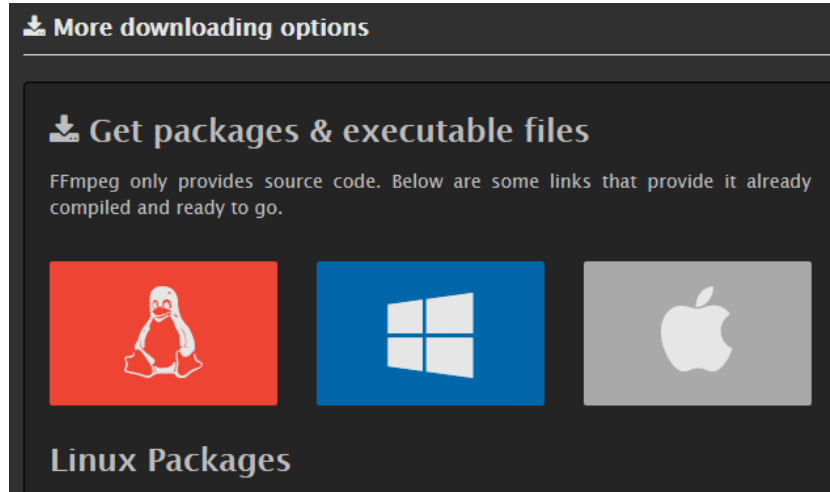
The screenshot shows the homepage of the Git for Windows website. At the top left, it says "git for windows" with "VERSION 2.39.0" below it. To the right are links for "FAQ", "REPOSITORY", and "MAILING LIST". A large orange diamond logo with a white Git branching diagram is on the left. To its right, the text reads "We bring the awesome **Git** SCM to Windows". Below this are two blue buttons: "Download" and "Contribute". A blue horizontal bar below the main content contains the text "Tools & Features". Underneath this bar, on the left, is a paragraph: "Git for Windows focuses on offering a lightweight, native set of tools that bring the full feature set of the [Git SCM](#) to Windows while providing appropriate user interfaces for experienced Git users and novices alike." On the right side of this section is a small image of a terminal window showing command-line output.

And follow the installation wizard.

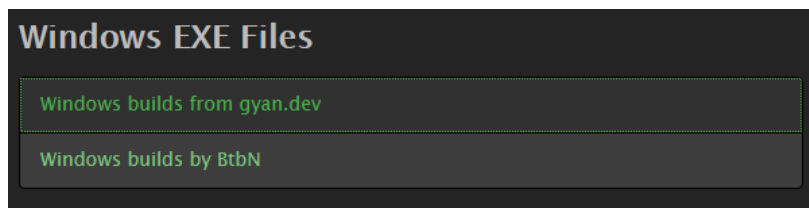
## FFMPEG

If you'd like to use the transcribe URL feature (such as a YouTube URL), you will also need to install FFMPEG.

Navigate to <https://ffmpeg.org/download.html> and click on the Window icon:



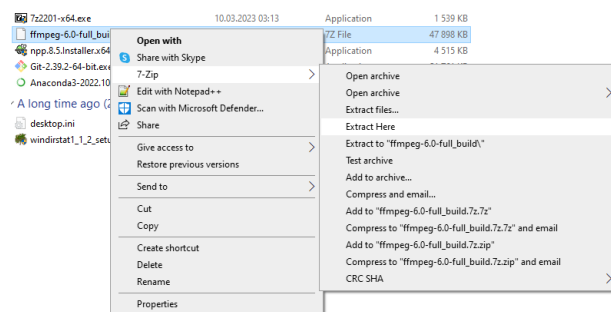
Get the builds from gyan.dev:



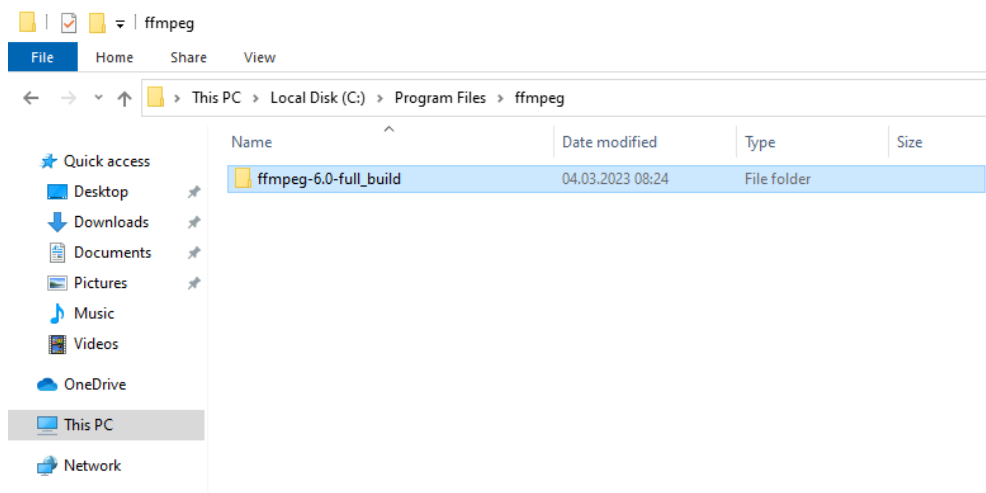
Scroll down to the release builds and download ffmpeg-release-full.7z:



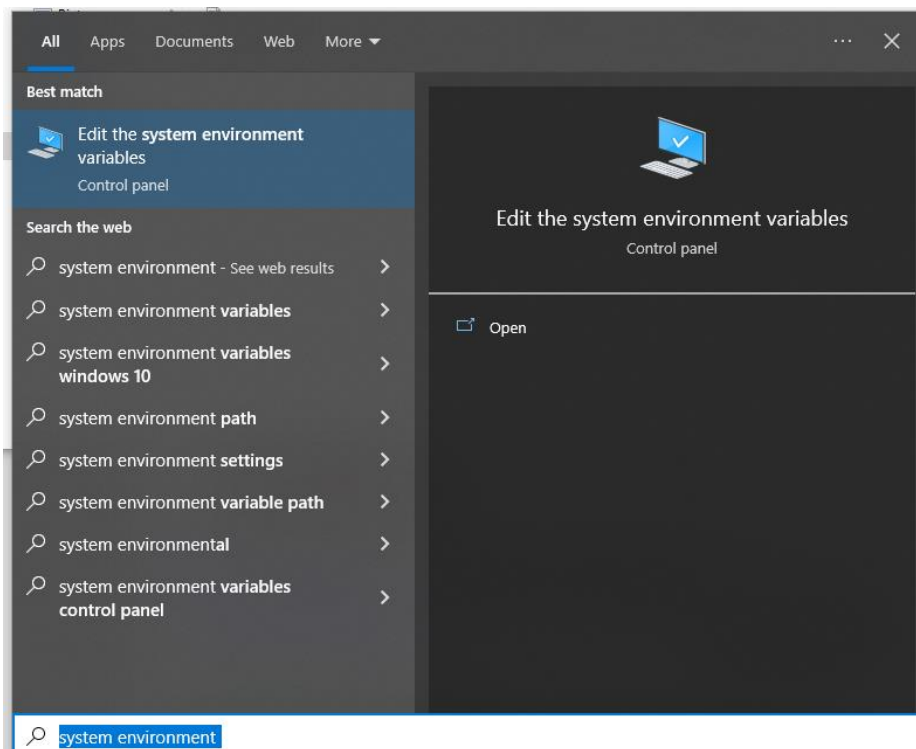
Use 7zip to extract this folder to a desired location, for instance C:\Program Files\ffmpeg\:



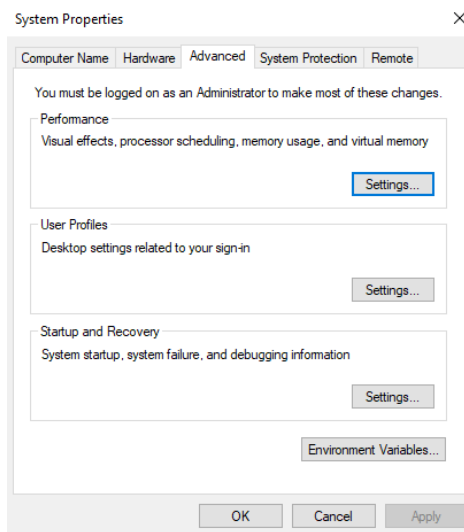
Move this folder to the desired location:



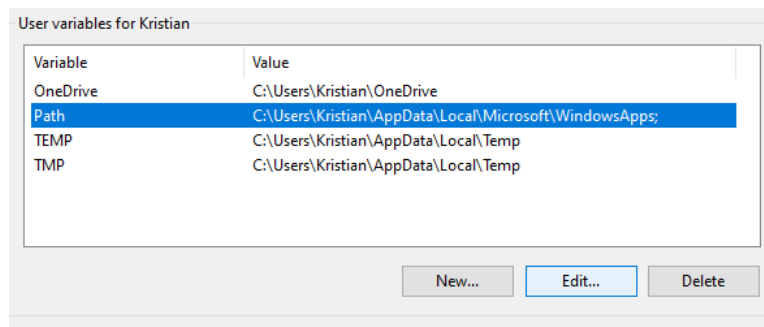
Next, we need to add FFMpeg to the system environment. To do this, type “system environment” into the search bar and click the “Edit the system environment variables” option.



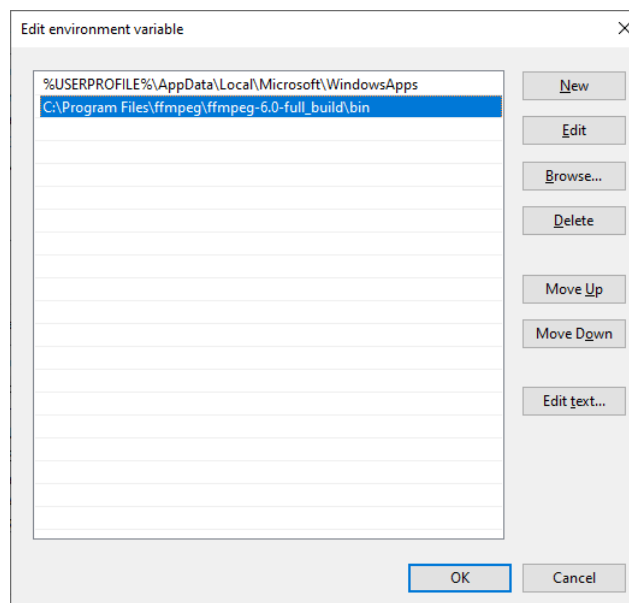
Click “Environment Variables...”



Then edit “Path” in “User variables” (or “System variables”):

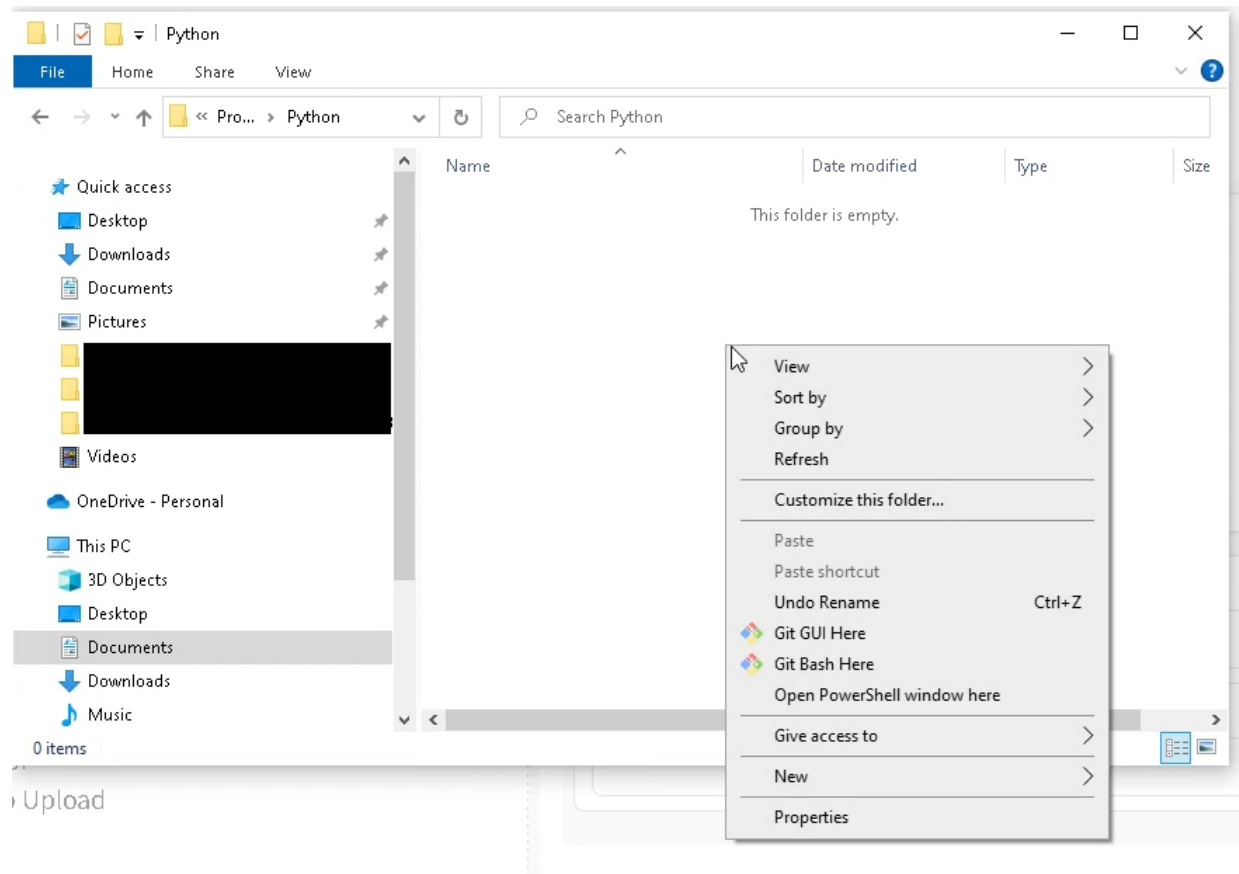


Click “New”, add the path to the “bin”-folder in the extracted ffmpeg-6.0-full\_build-folder, and click “OK”, then “OK”.



## Checkout Whisper WebUI

Next, we will checkout the Whisper WebUI project to a folder. Navigate to a location where you'd like to keep the Python project (for instance the Documents folder), and right click while holding the SHIFT key:



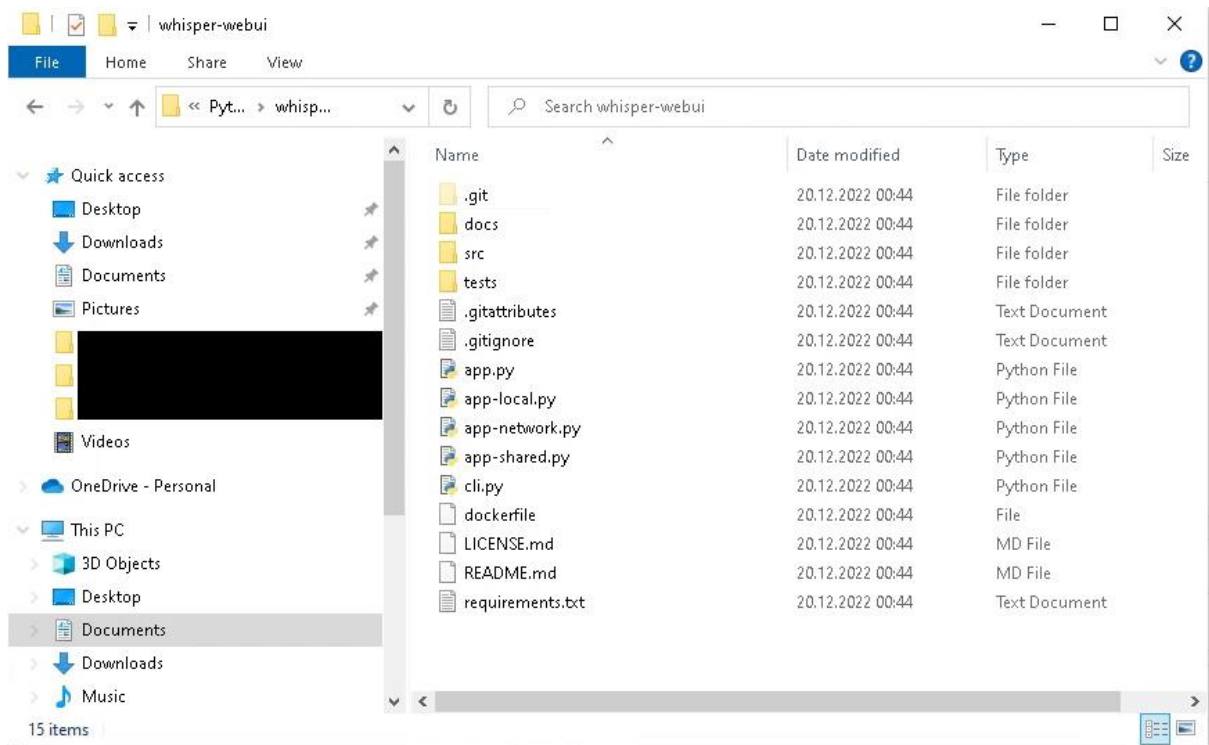
Click "Open Powershell window here". Then type the following command in the Powershell Window:

```
git clone "https://huggingface.co/spaces/aadnk/whisper-webui"
```

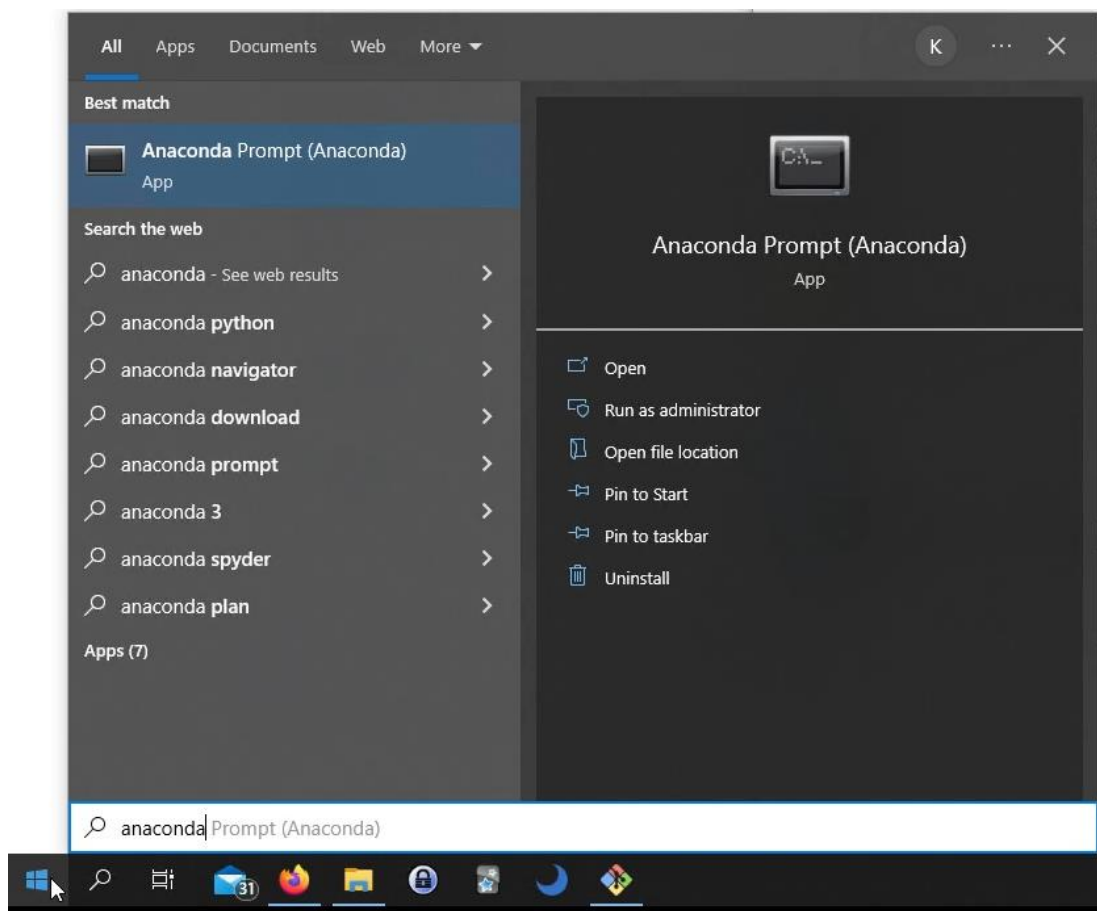
```
Windows PowerShell
PS C:\Users\Kristian\Documents\Programming\Python> git clone "https://huggingface.co/spaces/aadnk/whisper-webui"
Cloning into 'whisper-webui'...
```

Close the window and navigate into the cloned folder:



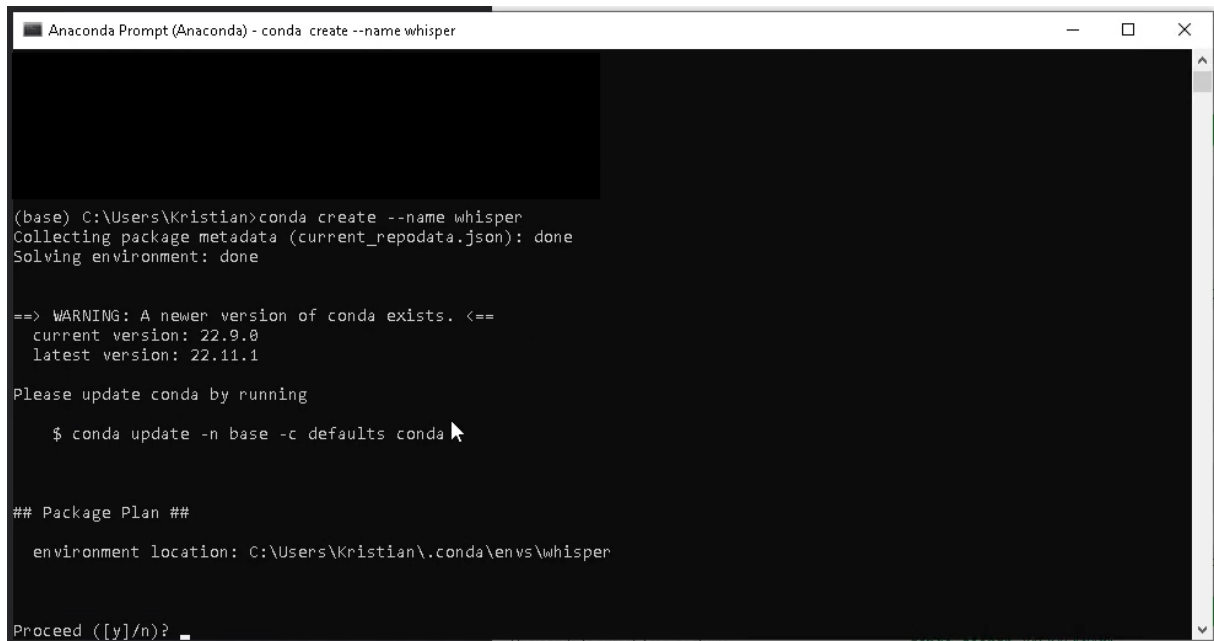


Next, click the Windows button, and type “Anaconda Prompt” into the search window:



Click “Anaconda Prompt”. Then type the following command to create a new “whisper” environment:

```
conda create --name whisper
```



```
Anaconda Prompt (Anaconda) - conda create --name whisper

(base) C:\Users\Kristian>conda create --name whisper
Collecting package metadata (current_repodata.json): done
Solving environment: done

==> WARNING: A newer version of conda exists. <==
  current version: 22.9.0
  latest version: 22.11.1

Please update conda by running

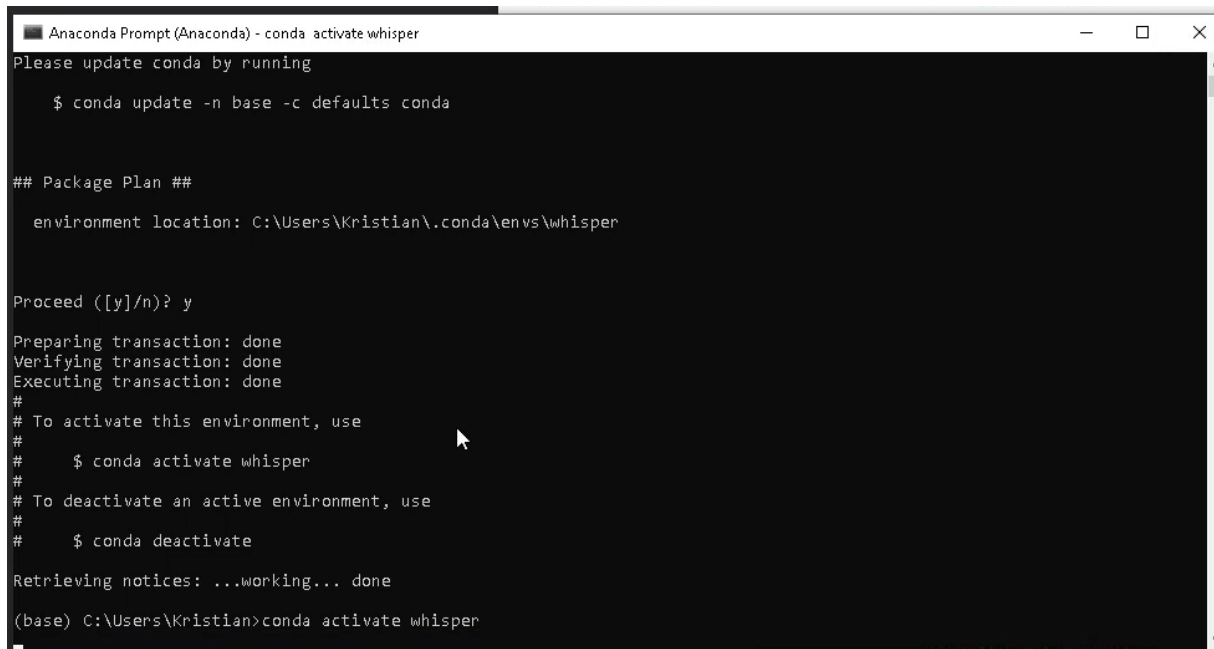
  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\Kristian\.conda\envs\whisper

Proceed ([y]/n)?
```

Type “y” to create this environment:



```
Anaconda Prompt (Anaconda) - conda activate whisper

Please update conda by running

  $ conda update -n base -c defaults conda

## Package Plan ##

  environment location: C:\Users\Kristian\.conda\envs\whisper

Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate whisper
#
# To deactivate an active environment, use
#
#   $ conda deactivate

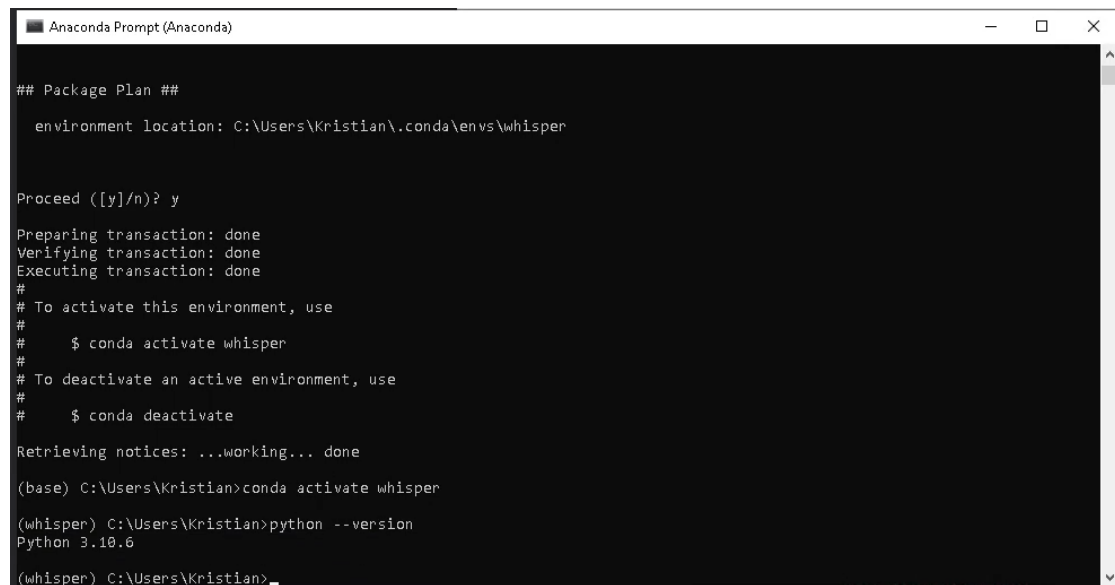
Retrieving notices: ...working... done

(base) C:\Users\Kristian>conda activate whisper
```



Next, type “conda activate whisper” to actually use this environment. You will have to type this every time you want to start the WebUI.

You should then check the Python version with the command “python --version”:



```
## Package Plan ##
environment location: C:\Users\Kristian\.conda\envs\whisper

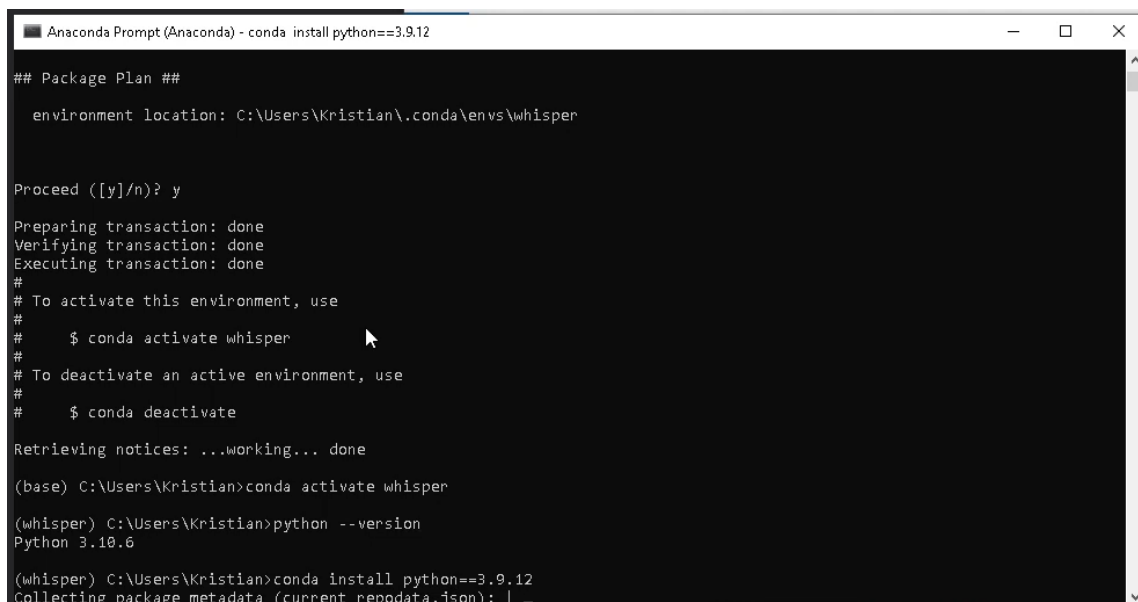
Proceed ([y]/n)? y
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate whisper
#
# To deactivate an active environment, use
#
#   $ conda deactivate

Retrieving notices: ..working... done
(base) C:\Users\Kristian>conda activate whisper

(whisper) C:\Users\Kristian>python --version
Python 3.10.6

(whisper) C:\Users\Kristian>
```

If this is different than Python 3.9.12, you may switch to Python 3.9.12 using the “conda install python==3.9.12” command:



```
## Package Plan ##
environment location: C:\Users\Kristian\.conda\envs\whisper

Proceed ([y]/n)? y
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate whisper
#
# To deactivate an active environment, use
#
#   $ conda deactivate

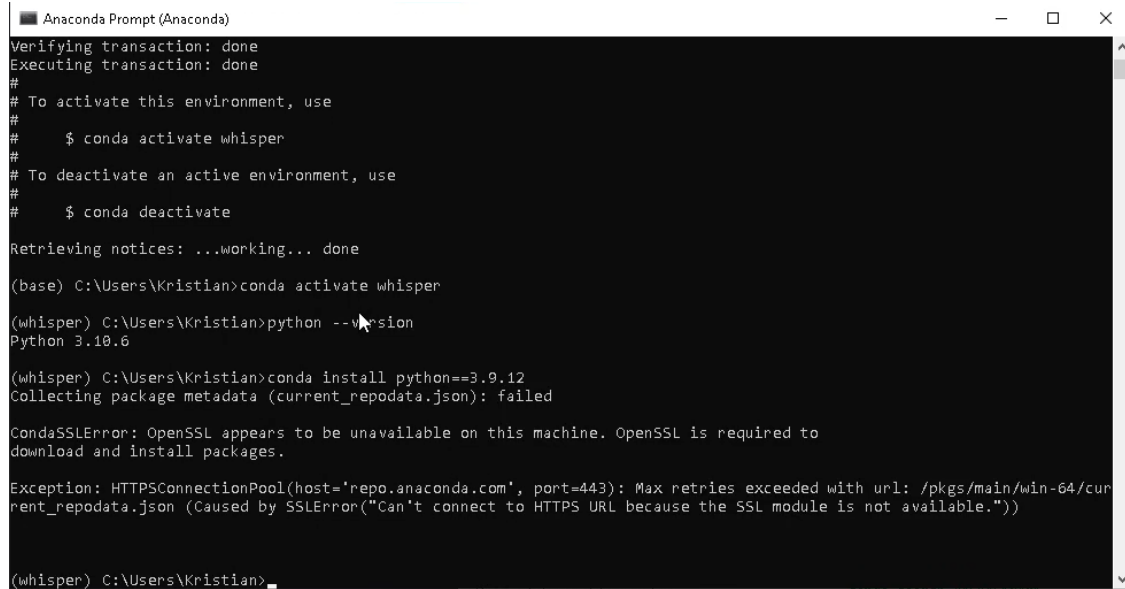
Retrieving notices: ..working... done
(base) C:\Users\Kristian>conda activate whisper

(whisper) C:\Users\Kristian>python --version
Python 3.10.6

(whisper) C:\Users\Kristian>conda install python==3.9.12
Collecting package metadata (current_repodata.json): |
```

## CondaSSLError

Now, for some reason, I encountered an error here – CondaSSLError: OpenSSL appears to be unavailable on this machine.



```

Anaconda Prompt (Anaconda)
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate whisper
#
# To deactivate an active environment, use
#
#   $ conda deactivate

Retrieving notices: ...working... done

(base) C:\Users\Kristian>conda activate whisper

(whisper) C:\Users\Kristian>python --version
Python 3.10.6

(whisper) C:\Users\Kristian>conda install python==3.9.12
Collecting package metadata (current_repodata.json): failed

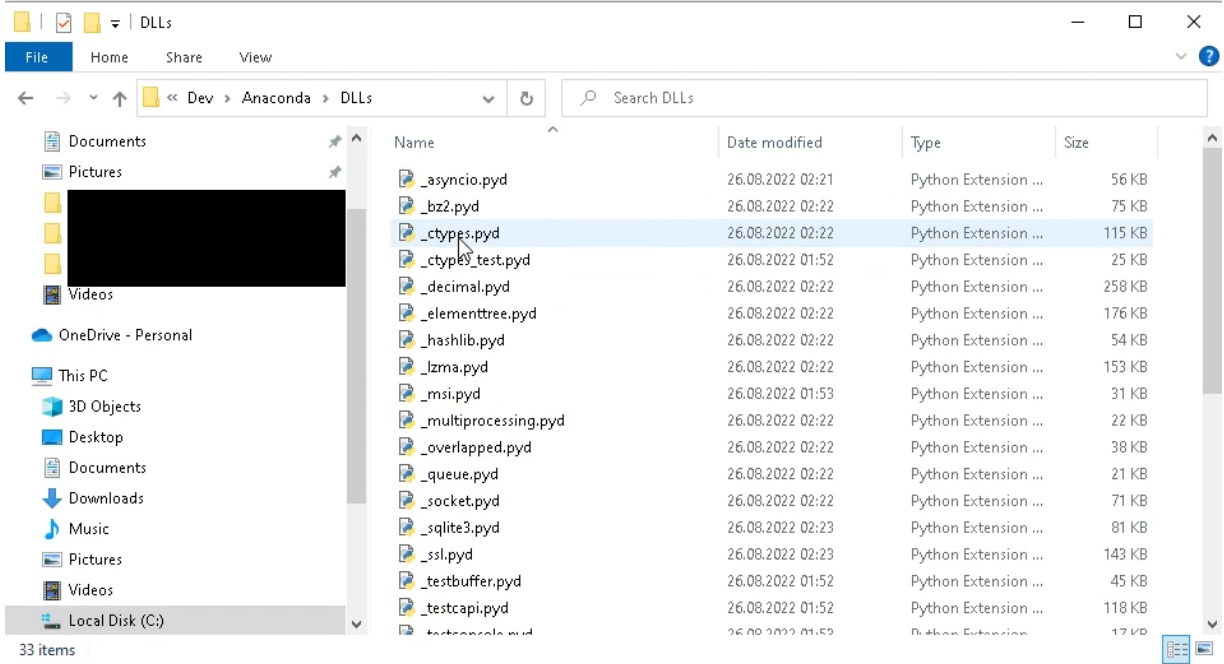
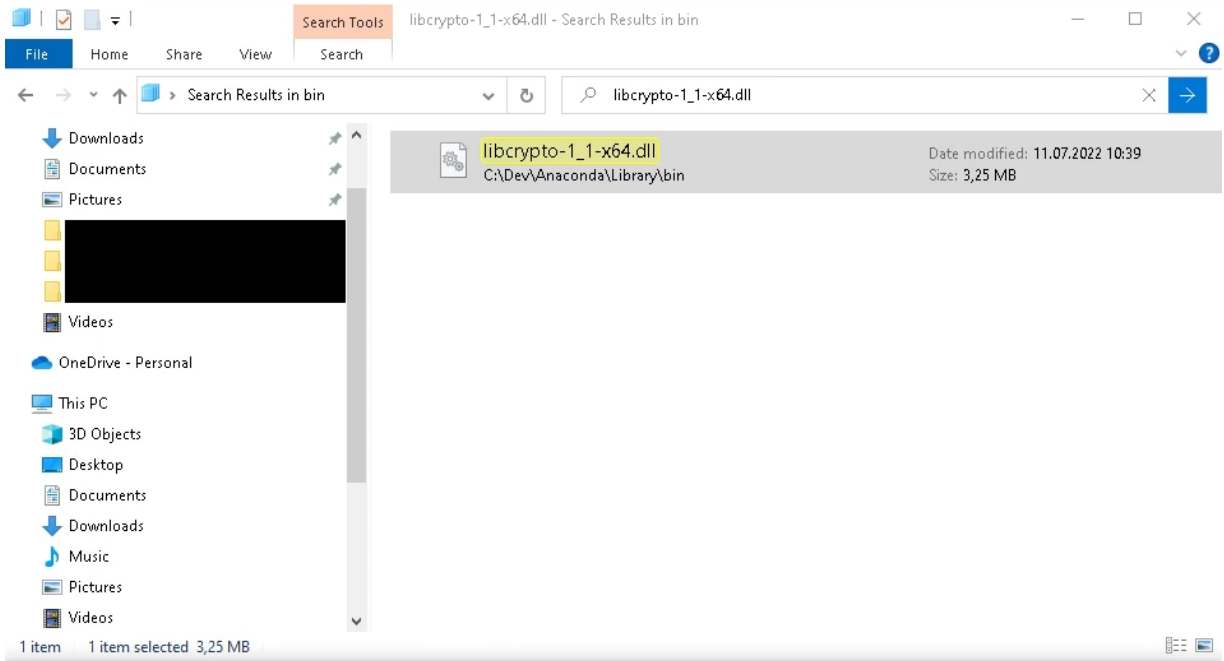
CondaSSLError: OpenSSL appears to be unavailable on this machine. OpenSSL is required to
download and install packages.

Exception: HTTPSConnectionPool(host='repo.anaconda.com', port=443): Max retries exceeded with url: /pkgs/main/win-64/cu
rent_repodata.json (Caused by SSLError("Can't connect to HTTPS URL because the SSL module is not available."))

(whisper) C:\Users\Kristian>
```

This seems to be an issue with the Python version in the base package. A simple workaround is to copy some DLL files as explained here:

- <https://github.com/conda/conda/issues/11982#issuecomment-1285929629>



You should then be able to install Python 3.9.12:

```
Anaconda Prompt (Anaconda) - conda install python==3.9.12
----- build
package -----
ca-certificates-2022.10.11 | haa95532_0 | 125 KB
certifi-2022.9.24 | py39haa95532_0 | 154 KB
openssl-1.1.1s | h2bbff1b_0 | 5.5 MB
pip-22.3.1 | py39haa95532_0 | 2.7 MB
python-3.9.12 | h6244533_0 | 17.1 MB
setuptools-65.5.0 | py39haa95532_0 | 1.1 MB
sqlite-3.40.0 | h2bbff1b_0 | 891 KB
tzdata-2022g | h04d1e81_0 | 114 KB
-----
Total: 27.7 MB

The following NEW packages will be INSTALLED:

ca-certificates pkgs/main/win-64::ca-certificates-2022.10.11-haa95532_0 None
certifi pkgs/main/win-64::certifi-2022.9.24-py39haa95532_0 None
openssl pkgs/main/win-64::openssl-1.1.1s-h2bbff1b_0 None
pip pkgs/main/win-64::pip-22.3.1-py39haa95532_0 None
python pkgs/main/win-64::python-3.9.12-h6244533_0 None
setuptools pkgs/main/win-64::setuptools-65.5.0-py39haa95532_0 None
sqlite pkgs/main/win-64::sqlite-3.40.0-h2bbff1b_0 None
tzdata pkgs/main/noarch::tzdata-2022g-h04d1e81_0 None
vc pkgs/main/win-64::vc-14.2-h21ff451_1 None
vs2015_runtime pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2 None
wheel pkgs/main/noarch::wheel-0.37.1-pyhd3eb1b0_0 None
wincertstore pkgs/main/win-64::wincertstore-0.2-py39haa95532_2 None

Proceed ([y]/n)?
```

Type "y".

```
Anaconda Prompt (Anaconda) - conda install python==3.9.12
openssl pkgs/main/win-64::openssl-1.1.1s-h2bbff1b_0 None
pip pkgs/main/win-64::pip-22.3.1-py39haa95532_0 None
python pkgs/main/win-64::python-3.9.12-h6244533_0 None
setuptools pkgs/main/win-64::setuptools-65.5.0-py39haa95532_0 None
sqlite pkgs/main/win-64::sqlite-3.40.0-h2bbff1b_0 None
tzdata pkgs/main/noarch::tzdata-2022g-h04d1e81_0 None
vc pkgs/main/win-64::vc-14.2-h21ff451_1 None
vs2015_runtime pkgs/main/win-64::vs2015_runtime-14.27.29016-h5e58377_2 None
wheel pkgs/main/noarch::wheel-0.37.1-pyhd3eb1b0_0 None
wincertstore pkgs/main/win-64::wincertstore-0.2-py39haa95532_2 None

Proceed ([y]/n)? y

Downloading and Extracting Packages
ca-certificates-2022 | 125 KB | ##### 100%
tzdata-2022g | 114 KB | ##### 100%
certifi-2022.9.24 | 154 KB | ##### 100%
sqlite-3.40.0 | 891 KB | ##### 100%
python-3.9.12 | 17.1 MB | ##### 100%
setuptools-65.5.0 | 1.1 MB | ##### 100%
openssl-1.1.1s | 5.5 MB | ##### 100%
pip-22.3.1 | 2.7 MB | ##### 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Retrieving notices: ...working... done

(whisper) C:\Users\Kristian>
```

## Installing Torch

If you want GPU acceleration, you will also need to install the GPU version of PyTorch. Go to to the following website:

- <https://pytorch.org/get-started/locally/>

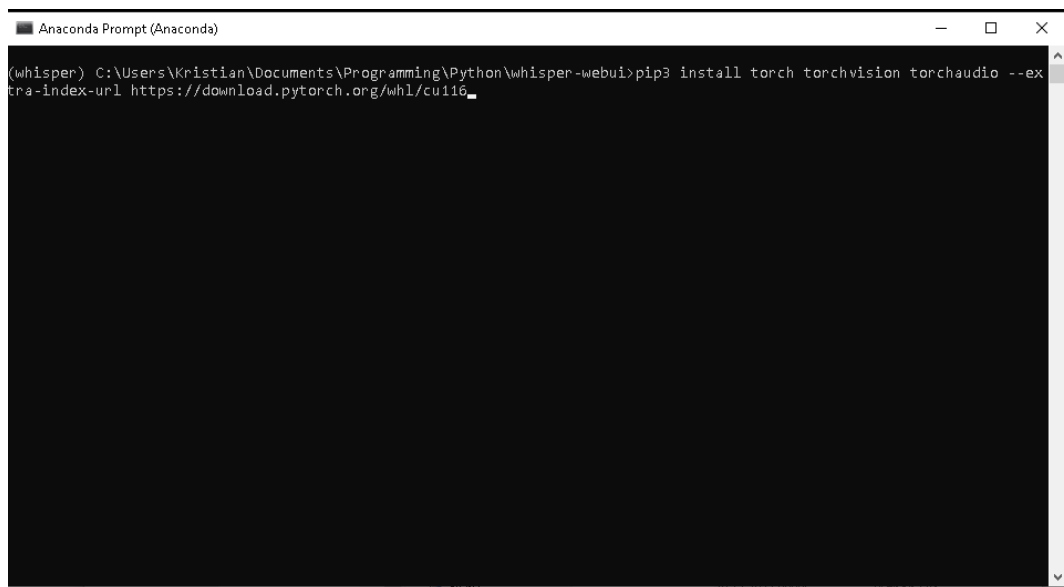
### START LOCALLY

Select your preferences and run the install command. Stable represents the most currently tested and supported version of PyTorch. This should be suitable for many users. Preview is available if you want the latest, not fully tested and supported, builds that are generated nightly. Please ensure that you have **met the prerequisites below (e.g., numpy)**, depending on your package manager. Anaconda is our recommended package manager since it installs all dependencies. You can also **install previous versions of PyTorch**. Note that LibTorch is only available for C++.

PyTorch Build	Stable (1.13.1)	Preview (Nightly)		
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python	C++ / Java		
Compute Platform	CUDA 11.6	CUDA 11.7	ROCm 5.2	CPU
Run this Command:	<pre>pip3 install torch torchvision torchaudio --extra-index-url https://download.pytorch.org/whl/cu116</pre>			

**NOTE:** PyTorch LTS has been deprecated. For more information, see [this blog](#).

Copy the command in “Run this command” and run it in the Anaconda Prompt window:



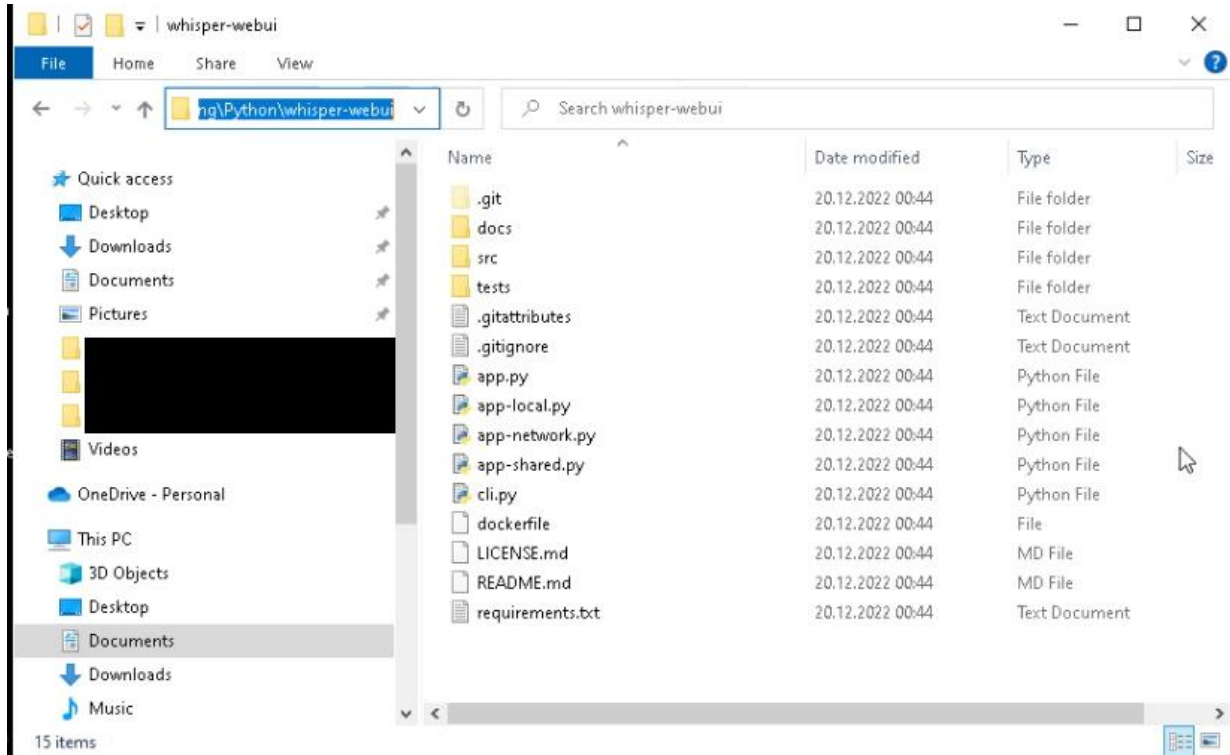
```
Anaconda Prompt (Anaconda)
(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>pip3 install torch torchvision torchaudio --extra-index-url https://download.pytorch.org/whl/cu116
```

This may take some to execute:

```
Anaconda Prompt (Anaconda) - conda install python==3.9.12 - pip install --force-reinstall torch torchvision torchaudio --extra-index-url https://d...
(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>pip install torch torchvision torchaudio --extra-index-url https://download.pytorch.org/whl/cu116
Looking in indexes: https://pypi.org/simple, https://download.pytorch.org/whl/cu116
Collecting torch
  Downloading https://download.pytorch.org/whl/cu116/torch-1.13.1%2Bcu116-cp39-cp39-win_amd64.whl (2434.0 MB)
----- 2.4/2.4 GB 1.9 MB/s eta 0:00:00
Collecting torchvision
  Using cached https://download.pytorch.org/whl/cu116/torchvision-0.14.1%2Bcu116-cp39-cp39-win_amd64.whl (4.8 MB)
Collecting torchaudio
  Downloading https://download.pytorch.org/whl/cu116/torchaudio-0.13.1%2Bcu116-cp39-cp39-win_amd64.whl (2.3 MB)
----- 2.3/2.3 MB 48.5 MB/s eta 0:00:00
Collecting typing-extensions
  Using cached typing_extensions-4.4.0-py3-none-any.whl (26 kB)
Collecting pillow<=8.3.*>=5.3.0
  Using cached Pillow-9.3.0-cp39-cp39-win_amd64.whl (2.5 MB)
Collecting numpy
  Using cached numpy-1.24.0-cp39-cp39-win_amd64.whl (14.9 MB)
```

## Installing requirements

Start by copying the path to the folder “whisper-webui” from Explorer:



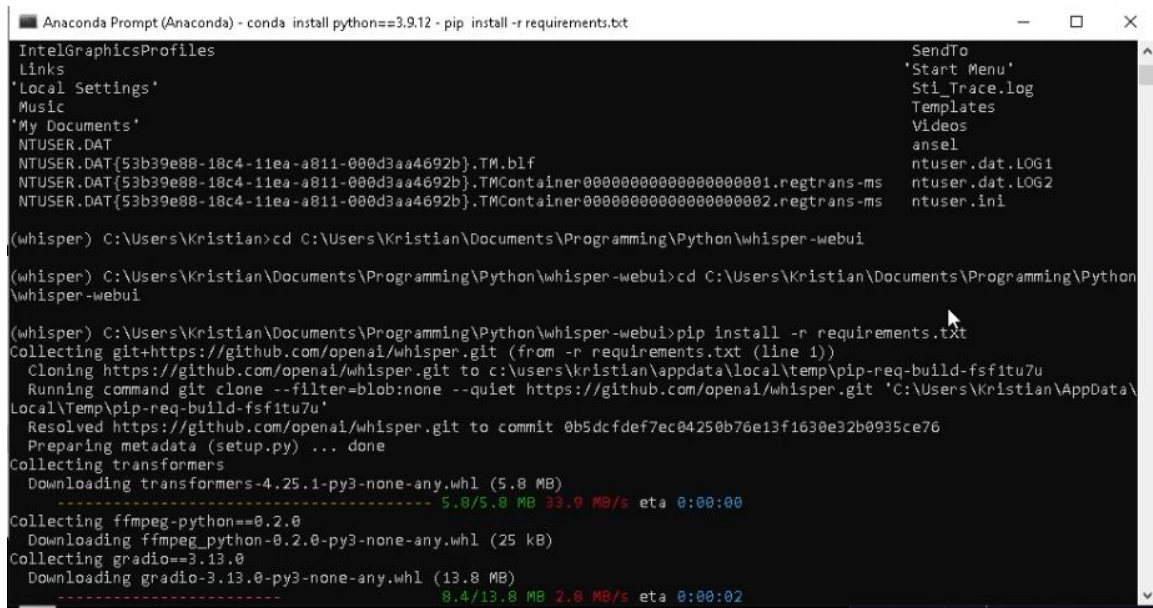
Execute the command “`cd FOLDER-PATH`”, where “FOLDER-PATH” is the path you copied from Explorer:

```
Anaconda Prompt (Anaconda) - conda install python==3.9.12
openssl-1.1.1s | 5.5 MB | ##### | 100%
pip-22.3.1 | 2.7 MB | ##### | 100%
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
Retrieving notices: ...working... done

(whisper) C:\Users\Kristian>ls
'3D Objects' NetHood
AppData OneDrive
'Application Data' OpenVPN
Contacts Pictures
Cookies PrintHood
Desktop Recent
Documents 'Save Games'
Downloads Scans
Favorites Searches
IntelGraphicsProfiles SendTo
Links 'Start Menu'
'Local Settings' Sti_Trace.log
Music Templates
'My Documents' Videos
NTUSER.DAT ansel
NTUSER.DAT{53b39e88-18c4-11ea-a811-000d3aa4692b}.TM.blf ntuser.dat.LOG1
NTUSER.DAT{53b39e88-18c4-11ea-a811-000d3aa4692b}.TM.Container0000000000000001.regtrans-ms ntuser.dat.LOG2
NTUSER.DAT{53b39e88-18c4-11ea-a811-000d3aa4692b}.TM.Container0000000000000002.regtrans-ms ntuser.ini

(whisper) C:\Users\Kristian>cd C:\Users\Kristian\Documents\Programming\Python\whisper-webui
(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>
```

Run the command “`pip install -r requirements.txt`”, which will install all other requirements in addition to PyTorch:



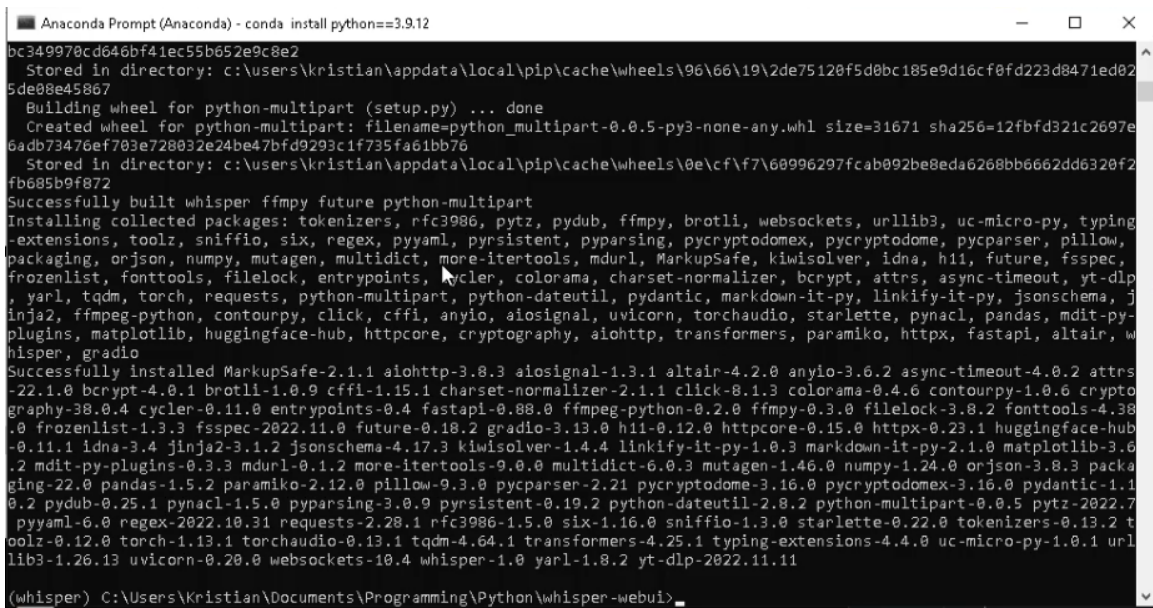
```
Anaconda Prompt (Anaconda) - conda install python==3.9.12 - pip install -r requirements.txt
IntelGraphicsProfiles
Links
Local Settings
Music
My Documents
NTUSER.DAT
NTUSER.DAT[53b39e88-18c4-11ea-a811-000d3aa4692b].TM.blf
NTUSER.DAT[53b39e88-18c4-11ea-a811-000d3aa4692b].TMContainer0000000000000001.regtrans-ms
NTUSER.DAT[53b39e88-18c4-11ea-a811-000d3aa4692b].TMContainer0000000000000002.regtrans-ms
ntuser.dat.LOG1
ntuser.dat.LOG2
ntuser.ini

(whisper) C:\Users\Kristian>cd C:\Users\Kristian\Documents\Programming\Python\whisper-webui

(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>cd C:\Users\Kristian\Documents\Programming\Python\whisper-webui

(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>pip install -r requirements.txt
Collecting git+https://github.com/openai/whisper.git (from -r requirements.txt (line 1))
  Cloning https://github.com/openai/whisper.git to c:\users\kristian\appdata\local\temp\pip-req-build-fsfitu7u
  Running command git clone --filter=blob:none --quiet https://github.com/openai/whisper.git 'C:\Users\Kristian\AppData\Local\Temp\pip-req-build-fsfitu7u'
  Resolved https://github.com/openai/whisper.git to commit 0b5dcfdef7ec04250b76e13f1630e32b0935ce76
  Preparing metadata (setup.py) ... done
Collecting transformers
  Downloading transformers-4.25.1-py3-none-any.whl (5.8 MB)
  ----- 5.8/5.8 MB 33.9 MB/s eta 0:00:00
Collecting ffmpeg-python==0.2.0
  Downloading ffmpeg_python-0.2.0-py3-none-any.whl (25 kB)
Collecting gradio==3.13.0
  Downloading gradio-3.13.0-py3-none-any.whl (13.8 MB)
  ----- 8.4/13.8 MB 2.6 MB/s eta 0:00:02
```

This may take some time ...



```
Anaconda Prompt (Anaconda) - conda install python==3.9.12
bc349970cd646bf41ec55b652e9c8e2
Stored in directory: c:\users\kristian\appdata\local\pip\cache\wheels\96\66\19\2de75128f5d0bc185e9d16cf0fd223d8471ed825de08e45867
Building wheel for python-multipart (setup.py) ... done
Created wheel for python-multipart: filename=python_multipart-0.0.5-py3-none-any.whl size=31671 sha256=12fbfd321c2697e6adb73476ef703e728032e24be47bfd9293c1f735fa61bb76
Stored in directory: c:\users\kristian\appdata\local\pip\cache\wheels\0e\c\cf\7\60996297fcab092be8eda6268bb6662dd6320f2fb685b9f872
Successfully built whisper ffmpeg python-multipart
Installing collected packages: tokenizers, rfc3986, pytz, pydub, ffmpeg, brotli, websockets, urllib3, uc-micro-py, typing-extensions, toolz, sniffio, six, regex, pyyaml, pyrsistent, pyparsing, pycryptodome, pycryptodome, pycparser, pillow, packaging, orjson, numpy, mutagen, multidict, more-itertools, mdurl, MarkupSafe, kiwisolver, idna, h11, future, fsspec, frozenlist, fonttools, filelock, entrypoints, cycler, colorama, charset-normalizer, bcrypt, attrs, async-timeout, yt-dlp, yarl, tqdm, torch, requests, python-multipart, python-dateutil, pydantic, markdown-it-py, linkify-it-py, jsonschema, Jinja2, ffmpeg-python, contourpy, click, cffi, anyio, aiosignal, uvicorn, torchaudio, starlette, pynacl, pandas, mdit-py-plugins, matplotlib, huggingface-hub, httpcore, cryptography, aiohttp, transformers, paramiko, httpx, fastapi, altair, whisper, gradio
Successfully installed MarkupSafe-2.1.1 aiohttp-3.8.3 aiosignal-1.3.1 altair-4.2.0 anyio-3.6.2 async-timeout-4.0.2 attrs-22.1.0 bcrypt-4.0.1 brotli-1.0.9 cffi-1.15.1 charset-normalizer-2.1.1 click-8.1.3 colorama-0.4.6 contourpy-1.0.6 cryptography-38.0.4 cycler-0.11.0 entrypoints-0.4 fastapi-0.88.0 ffmpeg-python-0.2.0 ffmpeg-0.3.0 filelock-3.8.2 fonttools-4.38.0 frozenlist-1.3.3 fsspec-2022.11.0 future-0.18.2 gradio-3.13.0 h11-0.12.0 httpcore-0.15.0 httpx-0.23.1 huggingface-hub-0.11.1 idna-3.4 Jinja2-3.1.2 jsonschema-4.17.3 kiwisolver-1.4.4 linkify-it-py-1.0.3 markdown-it-py-2.1.0 matplotlib-3.6.2 mdit-py-plugins-0.3.3 mdurl-0.1.2 more-itertools-9.0.0 multidict-6.0.3 mutagen-1.46.0 numpy-1.24.0 orjson-3.8.3 packaging-22.0 pandas-1.5.2 paramiko-2.12.0 pillow-9.3.0 pycparser-2.21 pycryptodome-3.16.0 pycryptodome-3.16.0 pydantic-1.10.2 pydub-0.25.1 pynacl-1.5.0 pyparsing-3.0.9 pyrsistent-0.19.2 python-dateutil-2.8.2 python-multipart-0.0.5 pytz-2022.7 PyYAML-6.0 regex-2022.10.31 requests-2.28.1 rfc3986-1.5.0 six-1.16.0 sniffio-1.3.0 starlette-0.22.0 tokenizers-0.13.2 toolz-0.12.0 torch-1.13.1 torchaudio-0.13.1 tqdm-4.64.1 transformers-4.25.1 typing-extensions-4.4.0 uc-micro-py-1.0.1 urllib3-1.26.13 uvicorn-0.20.0 websockets-10.4 whisper-1.0 yarl-1.8.2 yt-dlp-2022.11.11

(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>
```



## Running WebUI

If you have closed “Anaconda Prompt” from the previous steps, or restarted your computer, you may get back to this step by opening “Anaconda Prompt” by searching for it in Windows.

Then type “conda activate whisper” to load the Whisper environment. Finally type “cd FOLDER-PATH” (where “FOLDER-PATH” is path to the whisper-webui directory you cloned earlier).

To run the UI, simply type the following command:

```
python app.py --input_audio_max_duration -1
```



```
Anaconda Prompt (Anaconda) - conda install python==3.9.12
Created wheel for python-multipart: filename=python_multipart-0.0.5-py3-none-any.whl size=31671 sha256=12fbfd321c2697e6ad
b73476ef703e728032e24be47bfd9293c1f735fa61bb76
Stored in directory: c:\users\kristian\appdata\local\pip\cache\wheels\0e\cf\7\60996297fcab092be8eda6268bb6662dd6320f2fb685b9f872
Successfully built whisper ffmpeg future python-multipart
Installing collected packages: tokenizers, rfc3986, pytz, pydub, ffmpeg, brotli, websockets, urllib3, uc-micro-py, typing-extensions, toolz, sniffio, six, regex, pyyaml, pyrsistent, pyparsing, pycryptodomex, pycryptodome, pycparser, pillow, packaging, orjson, numpy, mutagen, multidict, more-itertools, mdurl, MarkupSafe, kiwisolver, idna, h11, future, fsspec, frozenlist, fonttools, filelock, entrypoints, cycler, colorama, charset-normalizer, bcrypt, attrs, async-timeout, yt-dlp, yarl, tqdm, torch, requests, python-multipart, python-dateutil, pydantic, markdown-it-py, linkify-it-py, jsonschema, Jinja2, ffmpeg-python, contourpy, click, cffi, anyio, aiosignal, uvicorn, torchaudio, starlette, pynacl, pandas, mdit-py-plugins, matplotlib, huggingface-hub, httpcore, cryptography, aiohttp, transformers, paramiko, httpx, fastapi, altair, whisper, gradio
Successfully installed MarkupSafe-2.1.1 aiohttp-3.8.3 aiosignal-1.3.1 altair-4.2.0 anyio-3.6.2 async-timeout-4.0.2 attrs-22.1.0 bcrypt-4.0.1 brotli-1.0.9 cffi-1.15.1 charset-normalizer-2.1.1 click-8.1.3 colorama-0.4.6 contourpy-1.0.6 cryptography-38.0.4 cycler-0.11.0 entrypoints-0.4 fastapi-0.88.0 ffmpeg-python-0.2.0 ffmpeg-0.3.0 filelock-3.8.2 fonttools-4.38.0 frozenlist-1.3.3 fsspec-2022.11.0 future-0.18.2 gradio-3.13.0 h11-0.12.0 httpcore-0.15.0 httpx-0.23.1 huggingface-hub-0.11.1 idna-3.4 Jinja2-3.1.2 jsonschema-4.17.3 kiwisolver-1.4.4 linkify-it-py-1.0.3 markdown-it-py-2.1.0 matplotlib-3.6.2 mdit-py-plugins-0.3.3 mdurl-0.1.2 more-itertools-9.0.0 multidict-6.0.3 mutagen-1.46.0 numpy-1.24.0 orjson-3.8.3 packaging-22.0 pandas-1.5.2 paramiko-2.12.0 pillow-9.3.0 pycparser-2.21 pycryptodome-3.16.0 pycryptodomex-3.16.0 pydantic-1.10.2 pydub-0.25.1 pynacl-1.5.0 pyparsing-3.0.9 pyrsistent-0.19.2 python-dateutil-2.8.2 python-multipart-0.0.5 pytz-2022.7 pyyaml-6.0 regex-2022.10.31 requests-2.28.1 rfc3986-1.5.0 six-1.16.0 sniffio-1.3.0 starlette-0.22.0 tokenizers-0.13.2 toolz-0.12.0 torch-1.13.1 torchaudio-0.13.1 tqdm-4.64.1 transformers-4.25.1 typing-extensions-4.4.0 uc-micro-py-1.0.1 urllib3-1.26.13 uvicorn-0.20.0 websockets-10.4 whisper-1.0 yarl-1.8.2 yt-dlp-2022.11.11

(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>ls
LICENSE.md  app-local.py  app-shared.py  cli.py  docs  src
README.md  app-network.py  app.py  dockerfile  requirements.txt  tests

(whisper) C:\Users\Kristian\Documents\Programming\Python\whisper-webui>python app.py --input_audio_max_duration -1
```

Note that depending on the amount of VRAM on your GPU, you may only be able to run the “base” or “small” model. You will get a “Out of memory” error if there’s not enough VRAM.

The image shows a terminal window with the following output:

```
does not have profile information (Triggered internally at C:\actions-runner\_work\pytorch\pytorch\builder\windows\p
pytorch\torch\torch\src\jit\codegen\cuda\graph_fuser.cpp:109.)
return forward_call(*input, **kwargs)
VAD processing took 5.994313899999998 seconds
Transcribing non-speech:
[{'end': 3.993999999999998, 'gap': True, 'start': 0},
 {'end': 32.0, 'expand_amount': 0.0, 'start': 3.993999999999998},
 {'end': 61.648, 'expand_amount': 0.0, 'start': 32.0},
 {'end': 89.488, 'expand_amount': 0.0, 'start': 61.648},
 {'end': 119.44, 'expand_amount': 0.0, 'start': 89.488},
 {'end': 120.9935, 'start': 119.44}]
Processing timestamps:
[{'end': 3.993999999999998, 'gap': True, 'start': 0},
 {'end': 32.0, 'expand_amount': 0.0, 'start': 3.993999999999998},
 {'end': 61.648, 'expand_amount': 0.0, 'start': 32.0},
 {'end': 89.488, 'expand_amount': 0.0, 'start': 61.648},
 {'end': 119.44, 'expand_amount': 0.0, 'start': 89.488},
 {'end': 120.9935, 'start': 119.44}]
Running whisper from 00:00.000 to 00:03.994 , duration: 3.993999999999998 expanded: 0 prompt: None language:
None
Loading whisper model small
100% | 461M/461M [00:23<00:00, 20.8MiB/s]
Running whisper from 00:03.994 to 00:32.000 , duration: 28.006 expanded: 0.0 prompt: None language: None
Running whisper from 00:32.000 to 01:01.648 , duration: 29.648000000000003 expanded: 0.0 prompt: 日本 日本 日本
日本 日本 日本 日本 日本 日本 日本 日本 日本 日本 language: japanese
Running whisper from 01:01.648 to 01:29.488 , duration: 27.839999999999996 expanded: 0.0 prompt: 日本 日本 日本
日本 日本 日本 日本 日本 日本 language: japanese
Running whisper from 01:29.488 to 01:59.440 , duration: 29.951999999999998 expanded: 0.0 prompt: 日本 日本 日本
日本 SNS 日本 日本 日本 日本 日本 日本 日本 日本 日本 日本 Nori JP Teacher 日本 日本 日本 日本 language: japanese
```

The Windows Task Manager Performance tab shows the following system metrics:

- CPU:** 18% usage, 3.92 GHz
- Memory:** 9,3/15,7 GB (59%) usage
- Disk 0 (C: D:):** 1% usage, SSD
- Ethernet:** Ethernet 2, S: 40,0 R: 8,0 Kbps
- GPU 0:** Intel(R) UHD Grap..., 8%
- GPU 1:** NVIDIA GeForce G..., 1% (72 °C)

The Memory section provides a detailed view of memory usage over 60 seconds and a breakdown of memory composition:

Category	In use (Compressed)	Available	Speed
System Memory	9,1 GB (383 MB)	6,4 GB	2667 MHz
GPU 1	17,6/20,1 GB	6,6 GB	Slots used: 2 of 2

Additional GPU 1 details: Form factor: SODIMM, Hardware reserved: 281 MB. Paged pool and Non-paged pool are also visible at the bottom.

Browser tabs: CUDA Tools, Anaconda, Git for Windows, Whisper W, conda-che, Gradio

Address bar: localhost:7860

Model: small

Language: Japanese

URL (YouTube, etc.):

Upload Files

out.mka	1.5 MB	Download
---------	--------	----------

Microphone Input

Record from microphone

Task:

VAD: silero-vad

VAD - Merge Window (s): 5

VAD - Max Merge Size (s): 30

VAD - Padding (s): 1

55.7

55.7

55.7

Flag