

Q6 GenAI - Containerized video transcription and chat app

1. Use the containerized app that we've created we will use it to video transcription and chat app
2. Make sure you have pinecone key and API key account
3. Clone the sample application's repository.

- a. In a terminal, run the following command
- b. `git clone https://github.com/Davidnet/docker-genai.git`

```
emilyweng@Emilys-MacBook-Pro ~ % git clone https://github.com/Davidnet/docker-genai.git
Cloning into 'docker-genai'...
remote: Enumerating objects: 66, done.
remote: Counting objects: 100% (66/66), done.
remote: Compressing objects: 100% (43/43), done.
remote: Total 66 (delta 24), reused 60 (delta 20), pack-reused 0
Receiving objects: 100% (66/66), 114.38 KiB | 1.13 MiB/s, done.
Resolving deltas: 100% (24/24), done.
emilyweng@Emilys-MacBook-Pro ~ % ll
zsh: command not found: ll
emilyweng@Emilys-MacBook-Pro ~ % ls
Applications      Movies            celery.sqlite     docker-genai-sample
Desktop           Music            celerybeat-schedule.db  intermediate_key.pem
Documents         Pictures         certificate.pem    keystore.jks
Downloads        Public           django-errors.log  my_ca
Library          VirtualBox VMs   docker-genai
```

- c.
4. Specify your API keys. In the docker-genai directory, create a text file called `.env` and specify your API keys inside.

- a. Sample:

```
#-----
# OpenAI
#-----
OPENAI_TOKEN=your-api-key # Replace your-api-key with your personal API key

#-----
# Pinecone
#-----
PINECONE_TOKEN=your-api-key # Replace your-api-key with your personal API key
~
```

5. Build and run the application. In a terminal, change directory to your **docker-genai** directory and run the following command.
- a. `docker compose up --build`

```
[+] Running 3/0
✓ Network docker-genai_default          Created
✓ Container docker-genai-bot-1          Created
✓ Container docker-genai-yt-whisper-1   Created
Attaching to bot-1, yt-whisper-1
bot-1 | Collecting usage statistics. To deactivate, set browser.gatherUsageStats to False.
yt-whisper-1 | Collecting usage statistics. To deactivate, set browser.gatherUsageStats to False.
bot-1 | You can now view your Streamlit app in your browser.
bot-1 | URL: http://0.0.0.0:8504
yt-whisper-1 | You can now view your Streamlit app in your browser.
yt-whisper-1 | URL: http://0.0.0.0:8503
```

- b.
6. Open a **browser** and access the **yt-whisper service** at
- a. <http://0.0.0.0:8503>

Chat with your youtube videos

This app uses OpenAI's [Whisper](#) model to generate a transcription of your videos and upload it to Pinecone.

Youtube URL

Submit

Processed videos:

Here are the videos you have processed so far:

You can download the transcription of the video by clicking on the corresponding video below

b.

Chat with your youtube videos

This app uses OpenAI's [Whisper](#) model to generate a transcription of your videos and upload it to Pinecone.

Youtube URL

<https://www.youtube.com/watch?v=8CY2aq3tcXA>


Submit



Running `process_video(...)`.

You can download the **transcription** of the video by clicking on the corresponding video below

- c.
- d. We will need to pay a fee for this to work (pay around \$5 for this)
7. To access servless index, we can link our GCP with Pinecone
 1. Log in to the [GCP Marketplace](#). Your project must be enabled for purchase by your billing administrator.
 2. Search for the [Pinecone listing](#).

 The **Pricing information** section on the Pinecone [listing page](#) is incorrect. For more information about pricing and tools to estimate the cost of Pinecone, see [How pricing works](#) and [Understanding cost](#).
 3. Click **Subscribe**.
 4. Select a billing account.
 5. Read and agree to the terms and conditions.
 6. Click **Subscribe**.
 7. In the **Your order request has been sent to Pinecone** modal, click **Sign up with Pinecone**.

A new window will open with a GCP-specific Pinecone sign up page. If the window does not open, click **Sign up with Pinecone** on the [Pinecone listing page](#).
 8. Choose an authentication method and click **Sign up**. If you have an existing Pinecone organization, we recommend using the same authentication method.
- a.

- b. Then we'll have to switch the organization in Pinecone to student GCP
- c. After processing the video, a video list appears in the web app that informs you which videos have been indexed in Pinecone. It also provides a button to download the transcript

You can download the transcription of the video by clicking on the corresponding video below

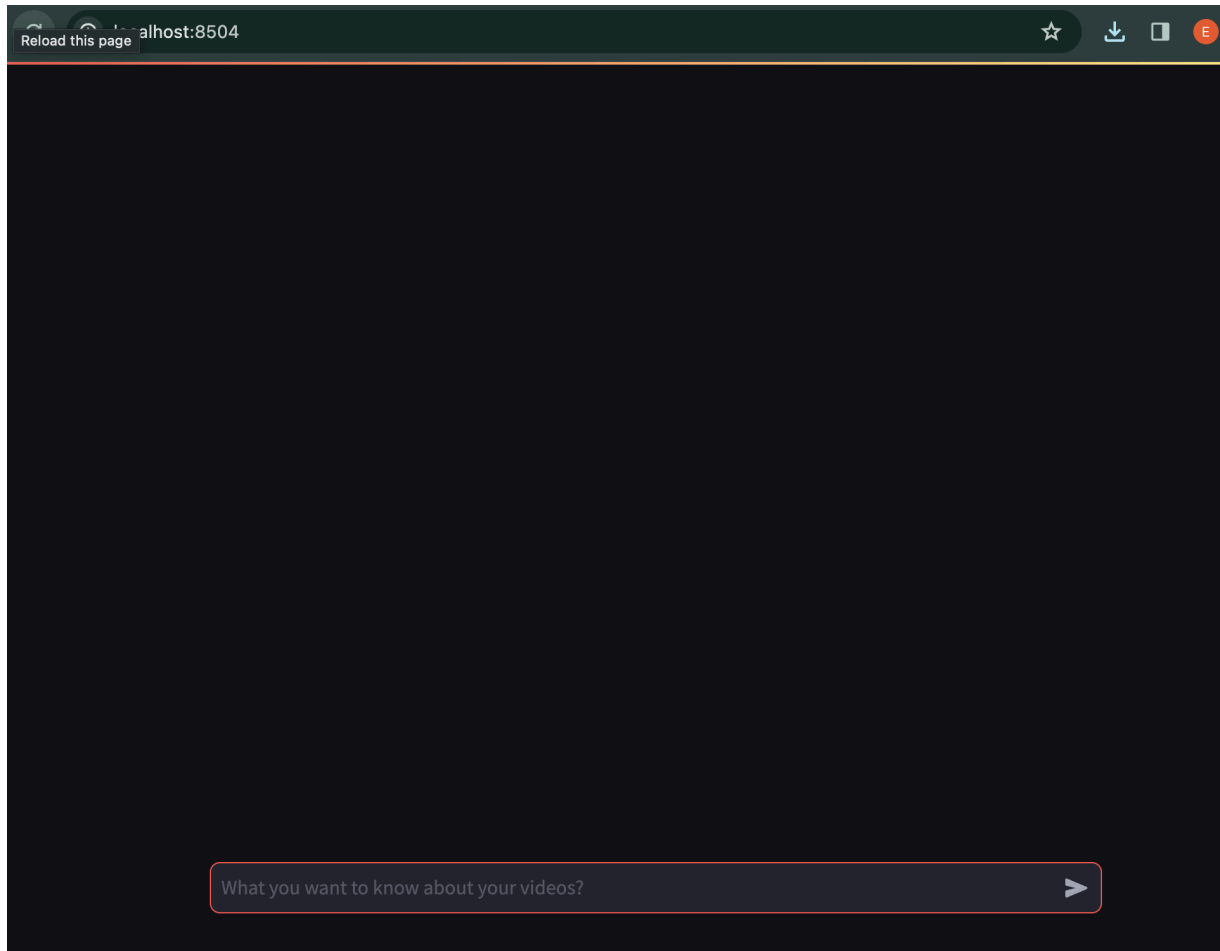
Develop ML interactive gpu-workflows with Visual Studio Code, Docker and Dockerhub



Download transcription

Download transcription 8CY2aq3tcXA

- d. <https://www.youtube.com/watch?v=8CY2aq3tcXA>
 - e. This was the youtube video provided in the example
8. Open a browser and access the service at
- a. <http://localhost:8504>



- b.
9. In the What do you want to know about your videos? text box, ask the Dockerbot a question about a video that was processed by the yt-whisper service.
 - a. I asked a question from the video and got the results:



What is nvidia



Dockerbot

NVIDIA is a technology company known primarily for its Graphics Processing Unit (GPU) products. These are specialized electronic circuits designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display device. NVIDIA's GPUs are widely used in applications ranging from video games to professional visualization, and increasingly for general purpose computing and artificial intelligence (AI) tasks.

In the context of the video transcripts provided:

1. NVIDIA drivers are mentioned as necessary for Linux machines to communicate with NVIDIA GPUs. These drivers are the software components that allow the operating system to interact with the hardware.
2. The importance of CUDA (Compute Unified Device Architecture) is highlighted. CUDA is a parallel computing platform and application programming interface (API) model created by NVIDIA. It allows software developers to use a CUDA-enabled graphics processing unit (GPU) for general purpose processing – an approach termed GPGPU (General-Purpose computing on Graphics Processing Units).

b.

3. CUDNN (CUDA Deep Neural Network library) is another NVIDIA library based on CUDA, providing GPU-accelerated primitives for deep learning algorithms. This aids in the execution of deep learning models more efficiently on NVIDIA GPUs.
4. The transcripts mention the challenges associated with installing NVIDIA software on Linux due to compatibility issues between NVIDIA and the Linux kernel, as well as historical tensions between NVIDIA and the Linux community, including remarks from Linus Torvalds, the creator of Linux.
5. Finally, it's noted that leveraging NVIDIA's technology (via CUDA and CUDNN) for machine learning and developing interactive GPU workflows can be complex. However, Docker and Visual Studio Code are presented as tools that can assist in creating more efficient development workflows for machine learning engineers working with NVIDIA's ecosystem.

In summary, NVIDIA is a key player in the computing industry, especially in the realms of graphics processing, parallel computing, and artificial intelligence, providing critical hardware and software that enable a wide range of computational tasks. Click on the following for more information:

- <https://www.youtube.com/watch?v=8CY2aq3tcXA&t=793s>
- <https://www.youtube.com/watch?v=8CY2aq3tcXA&t=538s>
- <https://www.youtube.com/watch?v=8CY2aq3tcXA&t=594s>

- c.
- d. We can select the first link and see the information provided based on our questions