GenAl - Containerized Video Transcription And Chat App

Name: Priyanka

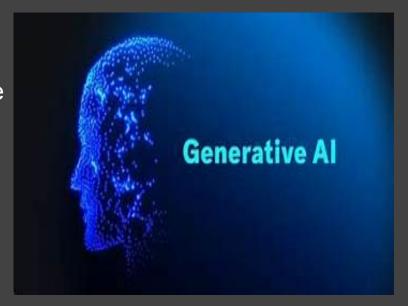
Student-id: 20179

Overview

- ____
- Introduction
- Design
- Implementation
- ❖ Test
- Conclusion
- References
- Appendix

Introduction

This project showcases the development of a containerized application leveraging Generative AI (GenAI) to transcribe videos and provide an integrated chat experience. The app utilizes cutting-edge AI models for speech-to-text conversion and real-time conversation capabilities, deployed within a scalable, containerized environment.



This includes:

- Adding a local database
- Adding a local or remote LLM service

Design

Add a Local Database

We can use containers to set up local services, like a database.

- In this section, you'll update the compose.yaml file to define a database service.
- In addition, you'll specify an environment variables file to load the database connection information rather than manually entering the information every time.

Add a Local or Remote LLM Service

The sample application supports both Ollama and OpenAl.

This guide provides instructions for the following scenarios:

Run Ollama in a Container

To run Ollama in a container and provide GPU access:

Run Your GenAl Application

At this point, you have the following services in your Compose file:

Open terminal and clone the following repository:

\$ git clone https://github.com/Davidnet/docker-genai.git

```
Welcome to Cloud Shell! Type "help" to get started.
Your Cloud Platform project in this session is set to my-project-last-444223.
Use "gcloud config set project [PROJECT ID]" to change to a different project.
priyanka599@cloudshell:~ (my-project-last-444223) $ 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 (from 0)
Receiving objects: 100% (66/66), 114.38 KiB | 984.00 KiB/s, done.
Resolving deltas: 100% (24/24), done.
priyanka599@cloudshell:~ (my-project-last-444223)$
```

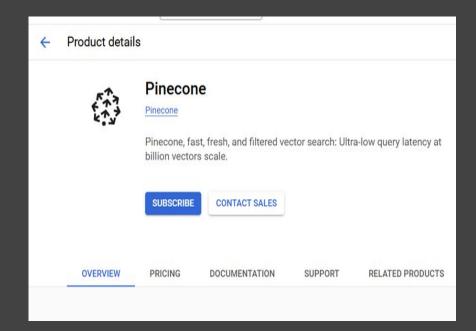
Go to openai.com > API login> API > Dashboard > APIKeys.

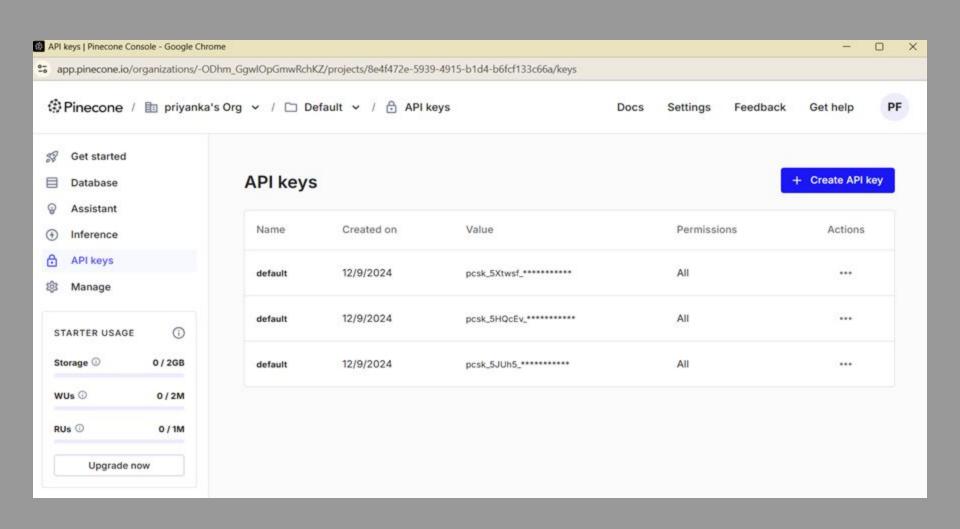
☐ Then click on 'Create new secret key' and give a name to the key.

Create new secret key Owned by Service account You This API key is tied to your user and can make requests against the selected project. If you are removed from the organization or project, this key will be disabled. Name Optional My Test Key Project containerApp No options Restricted Read Only Cancel Create secret key

Then get a pinecone API key.

For this, search pinecone in your google cloud and click on subscribe.





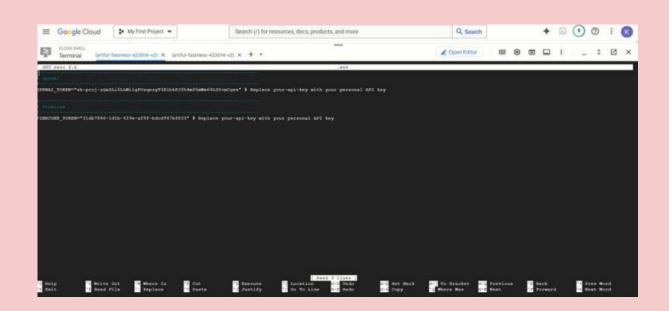
Specify your API keys

Go to docker-genai directory:

\$ cd docker-genai

Create a text file ".env":

\$ vim .env



Build and run the application after getting inside the folder.

\$ docker compose up -build

```
√ Network docker-genai default

√ Container docker-genai-yt-whisper-1

√ Container docker-genai-bot-1

Attaching to bot-1, yt-whisper-1
                Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.
                Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.
                  You can now view your Streamlit app in your browser.
                  URL: http://0.0.0.0:8504
                  You can now view your Streamlit app in your browser.
                  URL: http://0.0.0.0:8503
```

Test

Use the yt-whisper service

Open a browser and access the yt-whisper service at

http://localhost:8503

Chat with your youtube videos

This app uses OpenAl's <u>Whisper</u> model to generate a transcription of your videos and upload it to Pinecone.

Youtube URL

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



Processed videos:

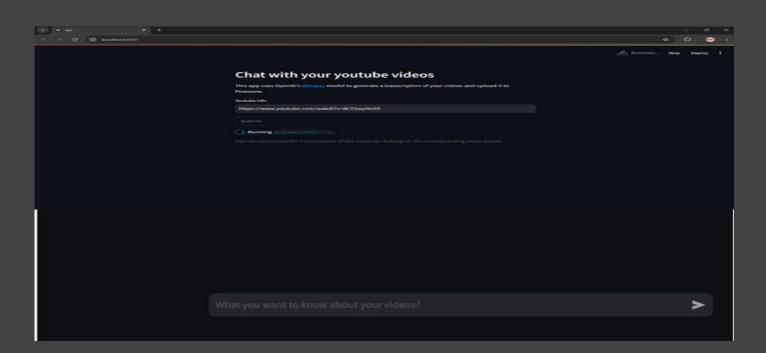
Here are the videos you have processed so far:

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

Test

Running on port 8503, this service processes YouTube videos, transcribes them using Whisper, and stores the data in Pinecone.

http://localhost:8503



Test

In the terminal, press Ctrl + C to stop the application:

```
[+] Running 2/0
 ✓ Container docker-genai-yt-whisper-1 Created
 ✓ Container docker-genai-bot-1
Attaching to bot-1, yt-whisper-1
                Collecting usage statistics. To deactivate, set browser.g
               Collecting usage statistics. To deactivate, set browser.q
                  You can now view your Streamlit app in your browser.
                  URL: http://0.0.0.0:8504
                  You can now view your Streamlit app in your browser.
yt-whisper-1
                 URL: http://0.0.0.0:8503
^CGracefully stopping... (press Ctrl+C again to force)
 / Container docker-genai-bot-1
  Container docker-genai-yt-whisper-1 Stopped
```

Conclusion

This GenAI project demonstrates the power of containerized applications in delivering scalable and efficient solutions for video transcription and AI-driven chat interactions. By combining advanced AI models with modern deployment techniques, this app addresses real-world challenges in video data processing and accessibility.



Reference

GenAl containerize your app

Develop your app

docker/genai-stack







Appendix

Google Slides

Github-Repo