GenAI - 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 OpenAI.

- This guide provides instructions for the following scenarios:

Run Ollama in a Container

To run Ollama in a container and provide GPU access:

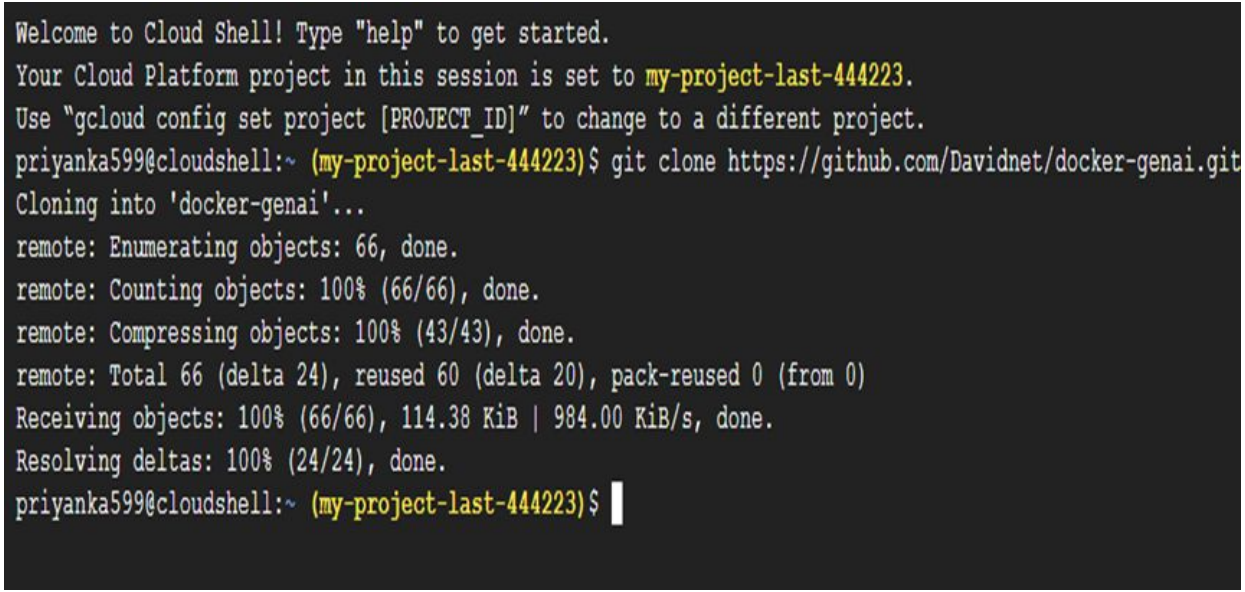
Run Your GenAI Application

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

Implementation

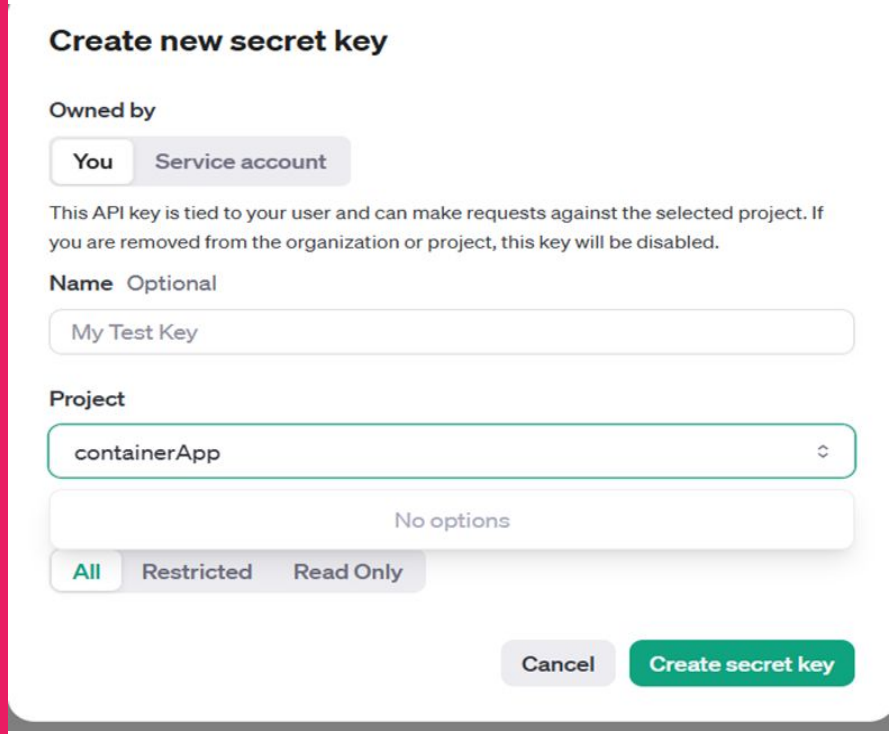
Open terminal and clone the following repository:

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



Implementation

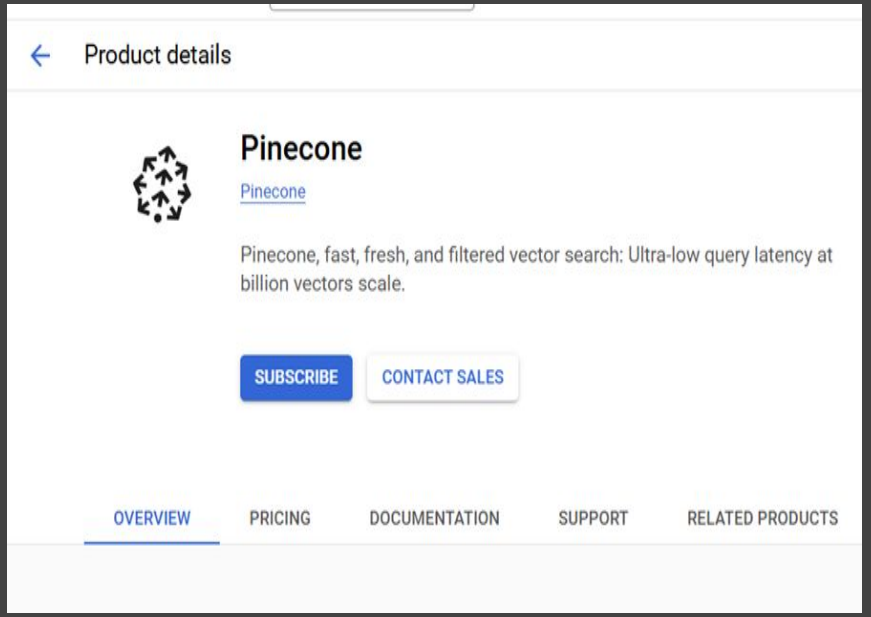
- Go to openai.com > API login > API > Dashboard > API Keys.
- Then click on ‘Create new secret key’ and give a name to the key.

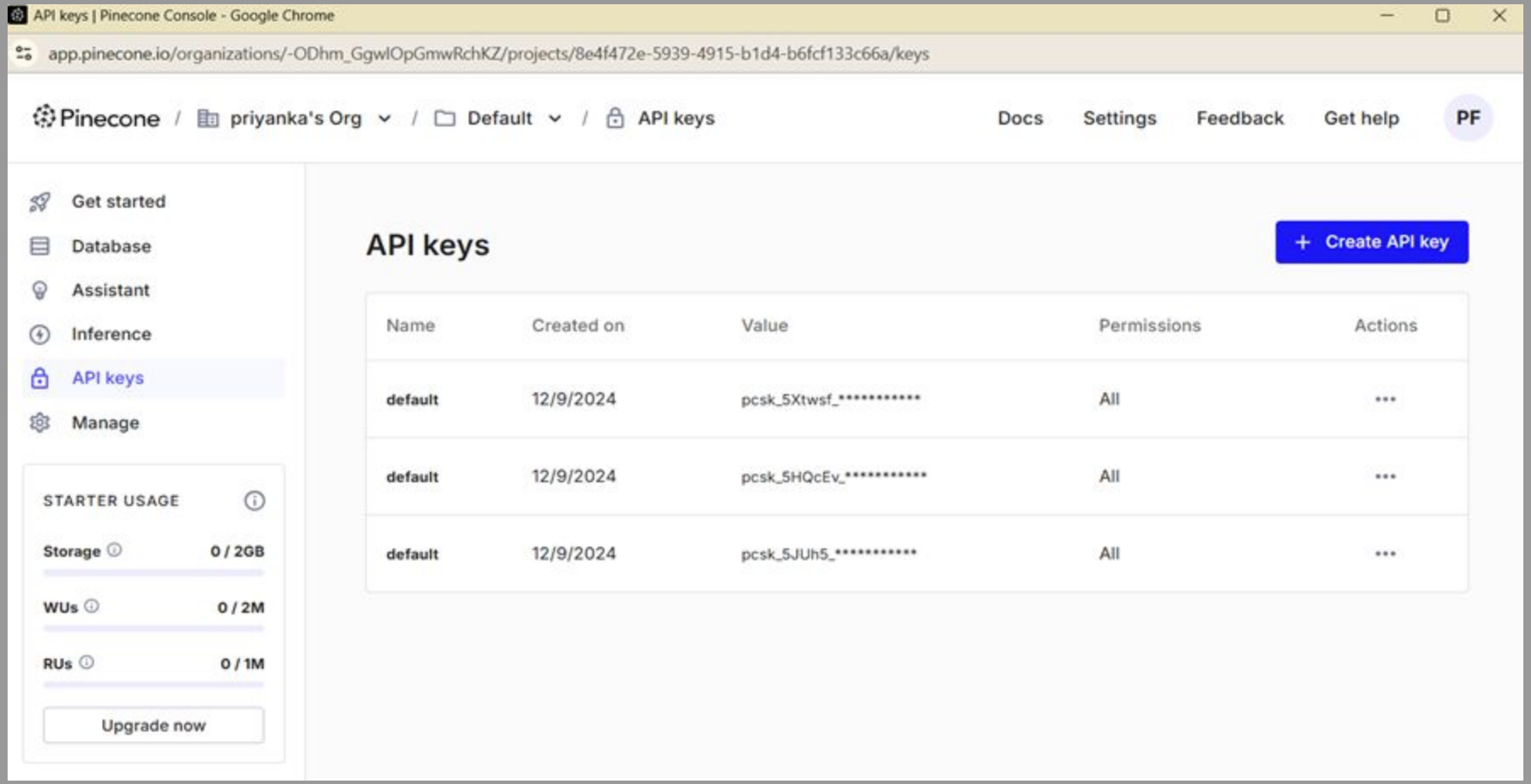


Implementation

Then get a pinecone API key.

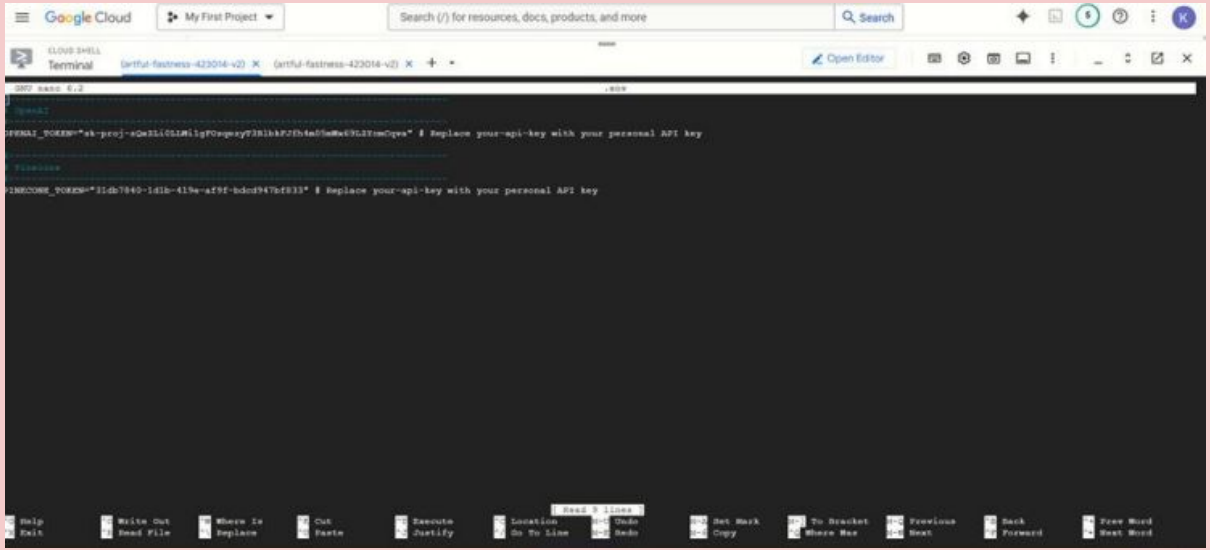
For this, search pinecone in your google cloud and click on subscribe.




API keys | Pinecone Console - Google Chrome
app.pinecone.io/organizations/-ODhm_GgwlOpGmwRchKZ/projects/8e4f472e-5939-4915-b1d4-b6fcf133c66a/keys
Pinecone / priyanka's Org / Default / API keys
Docs
Settings
Feedback
Get help
PF
Get started
Database
Assistant
Inference
API keys
Manage
STARTER USAGE
Storage 0 / 2GB
WUs 0 / 2M
RUs 0 / 1M
Upgrade now
API keys
+ Create API key
Name
Created on
Value
Permissions
Actions
default
12/9/2024
pcsk_5Xtwsf_**********
All
default
12/9/2024
pcsk_5HQcEv_**********
All
default
12/9/2024
pcsk_5JUh5_**********
All

Implementation

Specify your API keys.



Go to docker-genai directory:

$ cd docker-genai

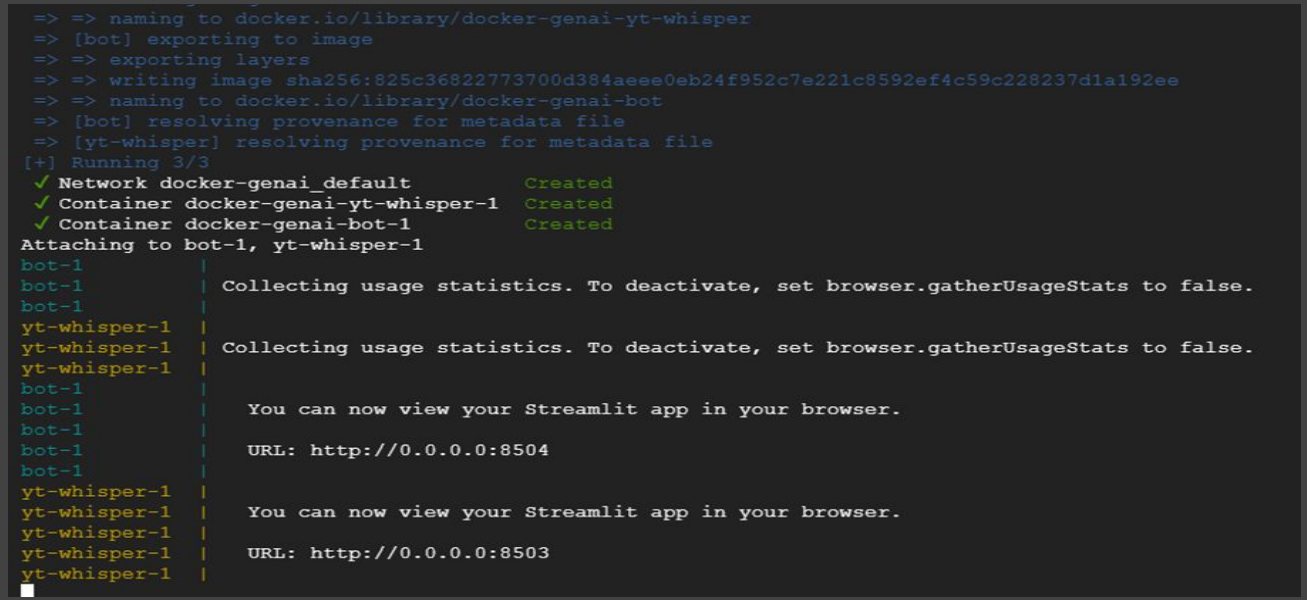
Create a text file “.env”:

$ vim .env

Implementation

Build and run the application after getting inside the folder.

$ docker compose up –build

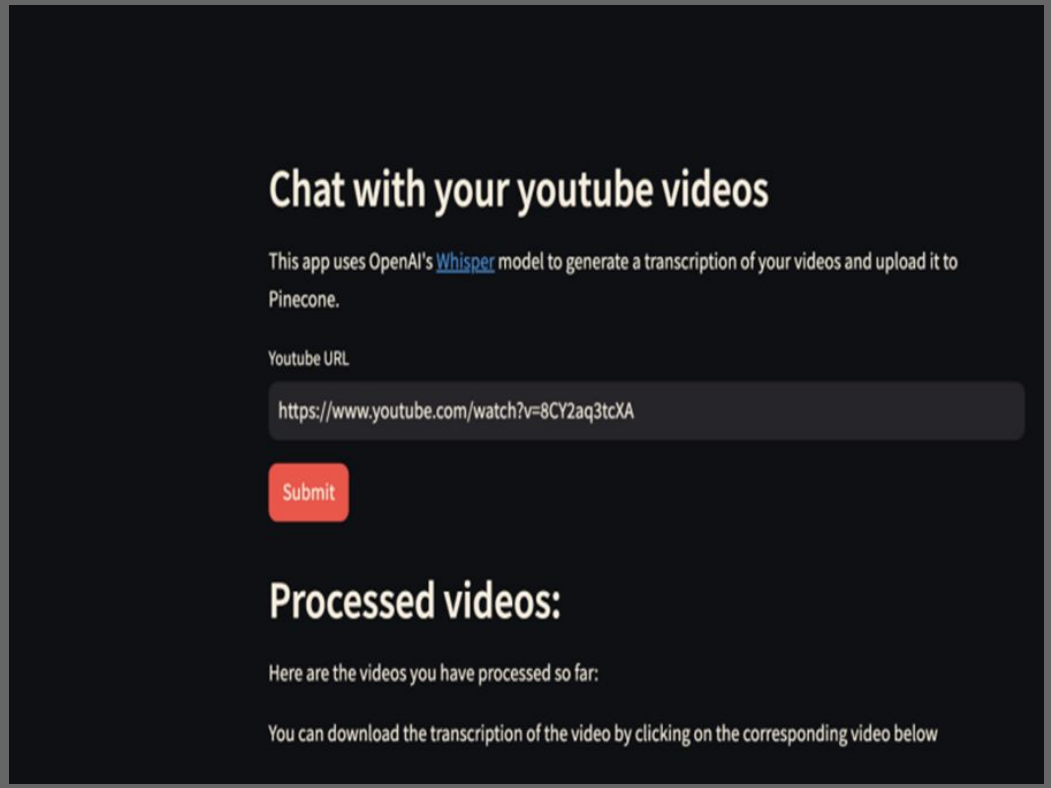


Test

Use the yt-whisper service

Open a browser and access the yt-whisper service at

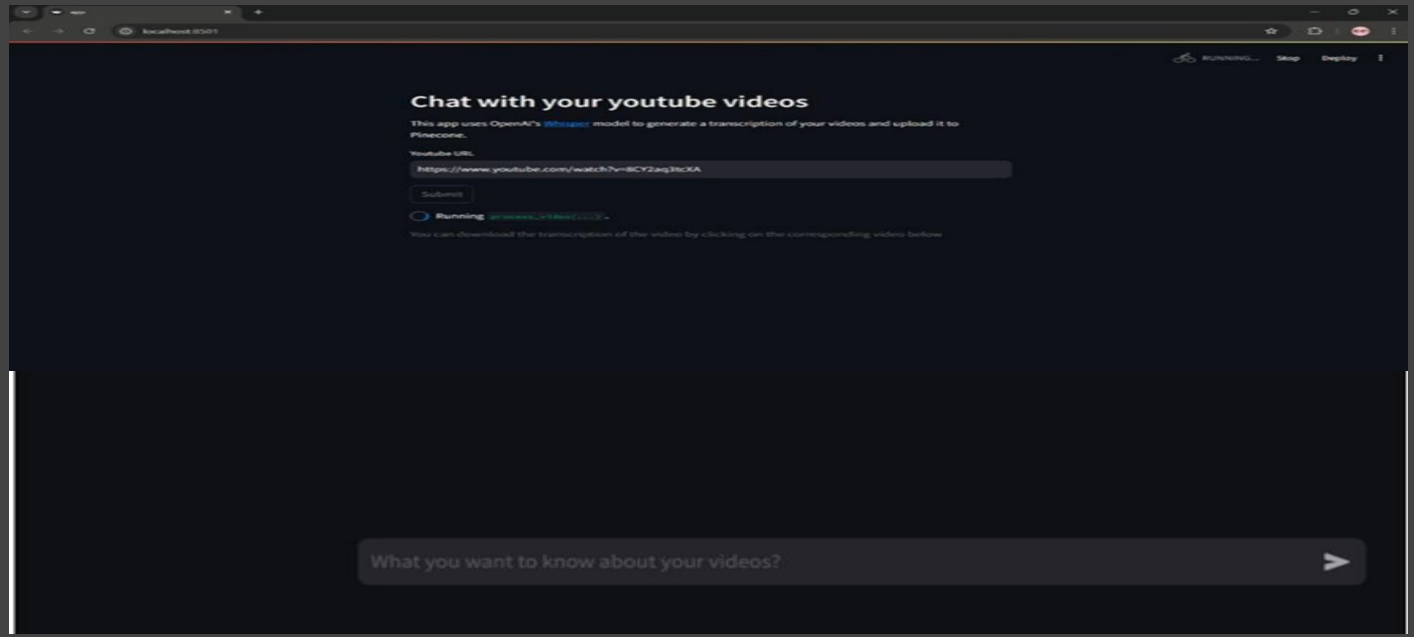
http://localhost:8503



Test

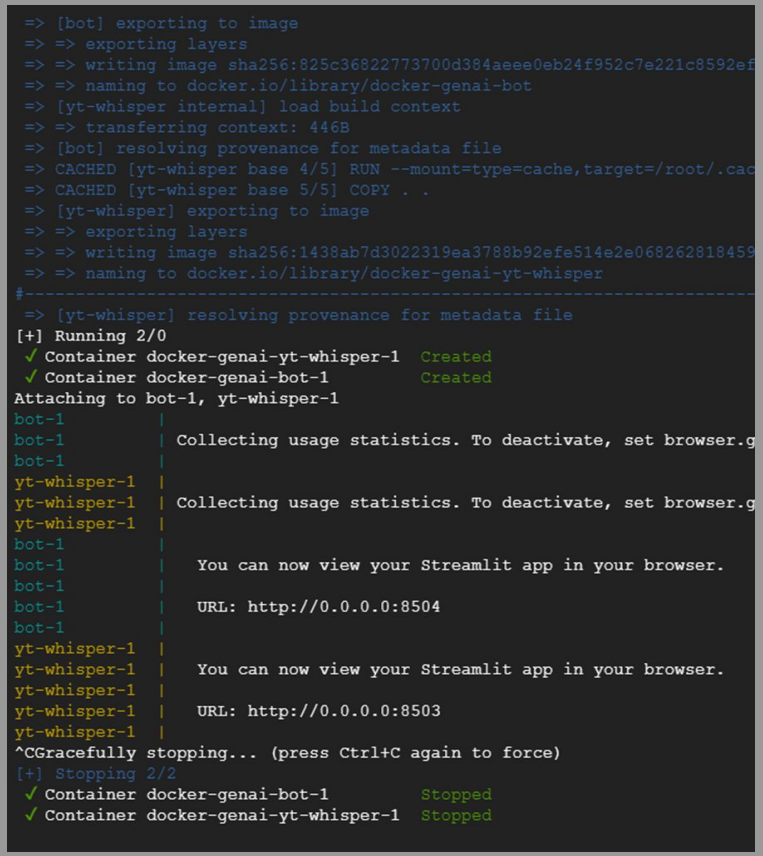
Running on port 8503, this service processes YouTube videos, transcribes them using Whisper, and stores the data in Pinecone.

http://localhost:8503



Test

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



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

GenAI containerize your app

Develop your app

docker/genai-stack

Appendix

Google Slides

Github-Repo