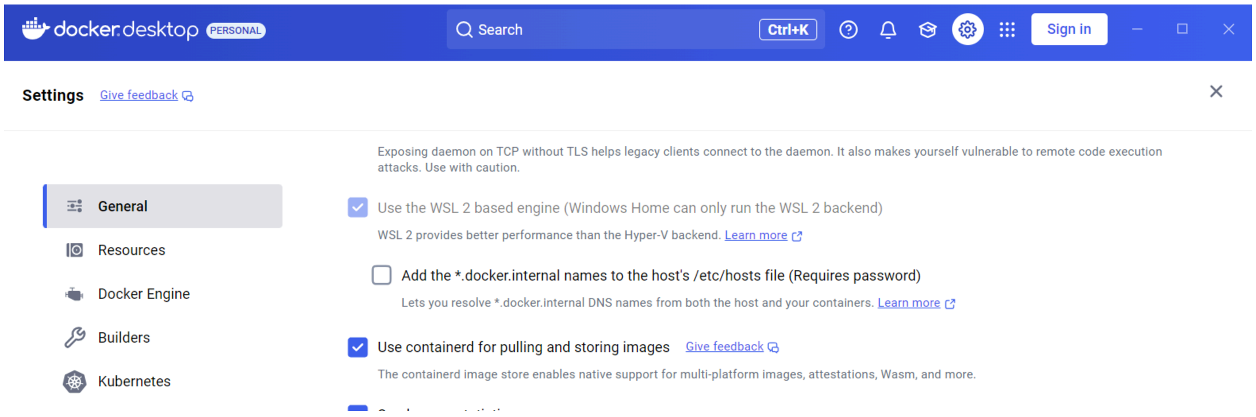
Project: GenAI - Containerize your app - optional

1. Prerequisites

- DockerDesktop supports GPU acceleration only on Windows with the WSL2 backend
- The examples in this section use a command-line-based git client, but you can use any client.



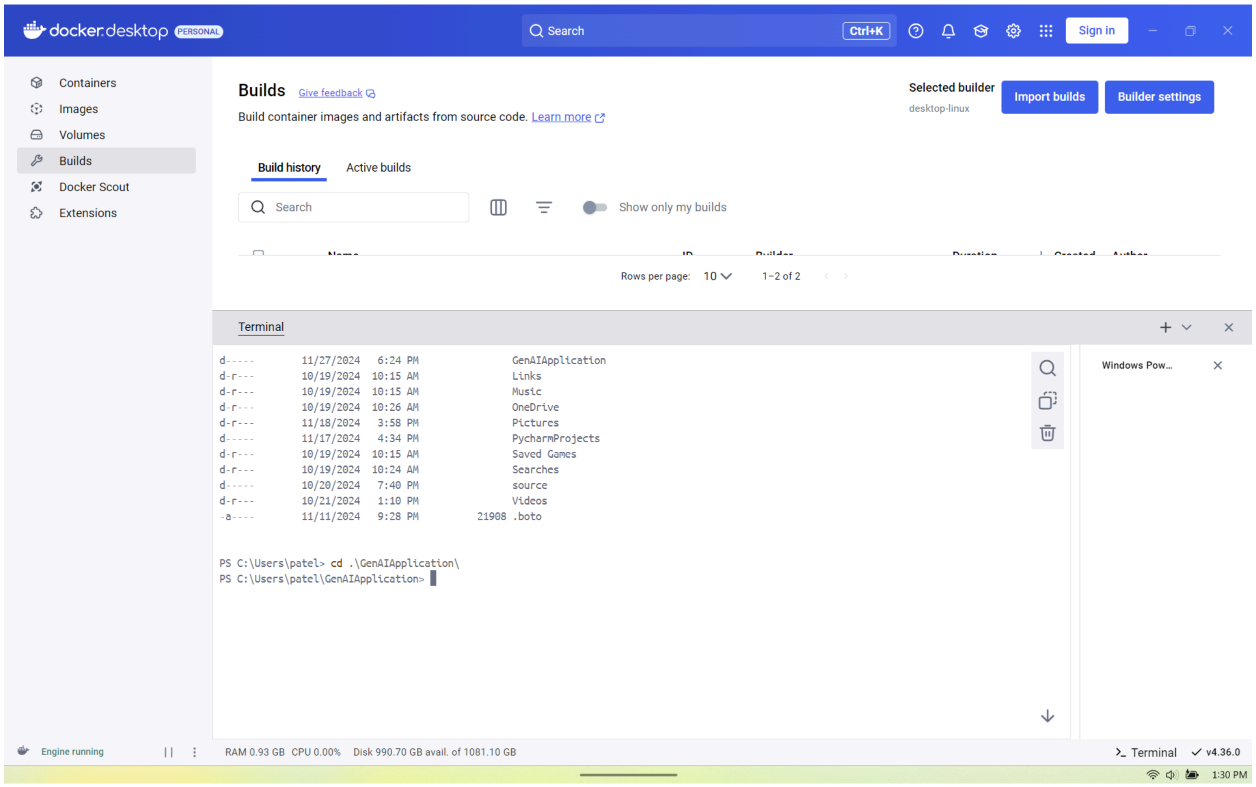
Exposing daemon on TCP without TLS helps legacy clients connect to the daemon. It also makes yourself vulnerable to remote code execution attacks. Use with caution.

[✓] Use the WSL 2 based engine (Windows Home can only run the WSL 2 backend)
WSL 2 provides better performance than the Hyper-V backend. Learn more

[ ] Add the *.docker.internal names to the host's /etc/hosts file (Requires password)
Lets you resolve *.docker.internal DNS names from both the host and your containers. Learn more

[✓] Use containerd for pulling and storing images Give feedback
The containerd image store enables native support for multi-platform images, attestations, Wasm, and more.

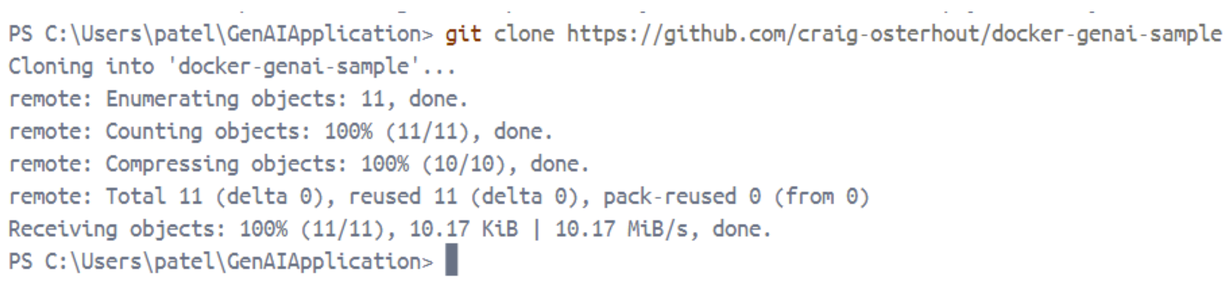
2. Create a working directory and navigate inside it(optional).



Build container images and artifacts from source code. Learn more

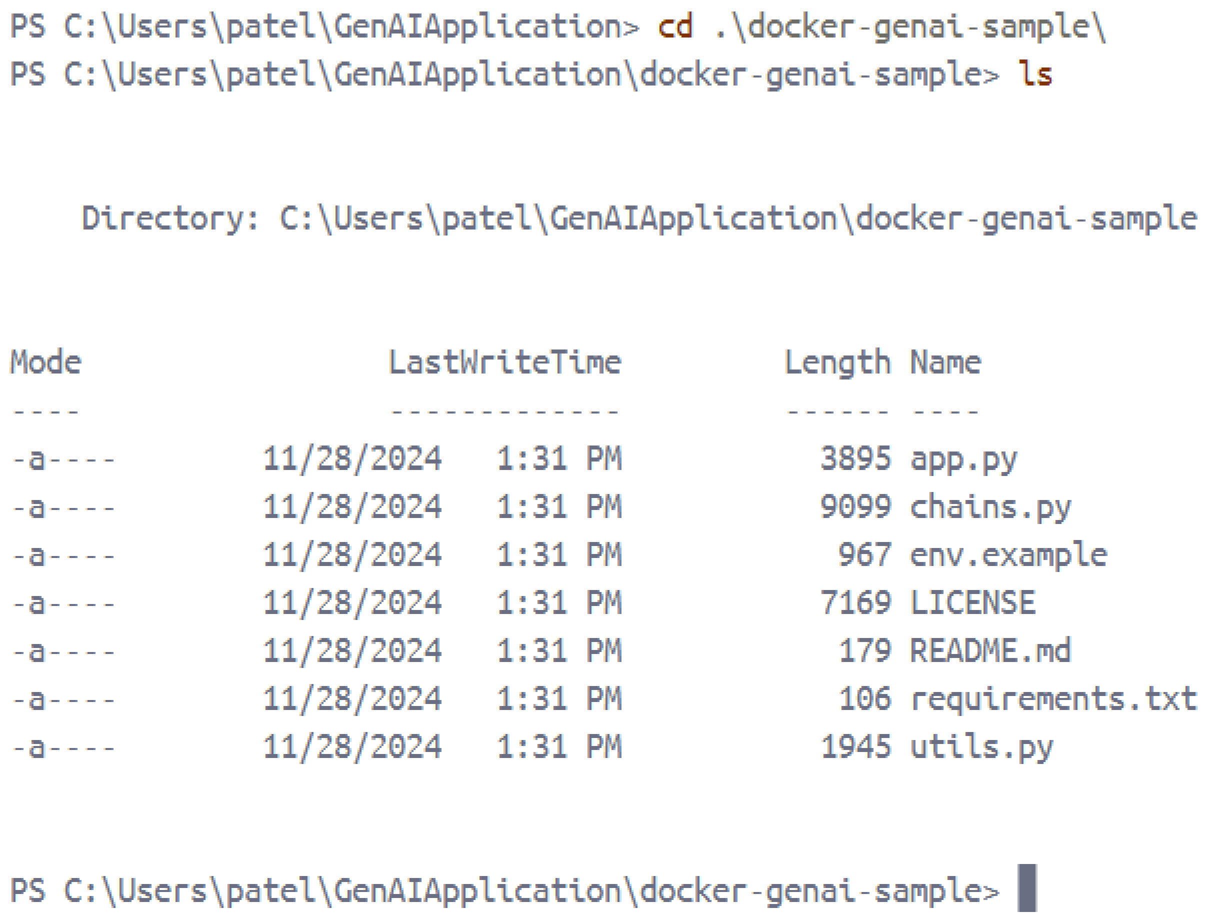
```
d-----        11/27/2024   6:24 PM                GenAIApplication
d-r---        10/19/2024  10:15 AM                Links
d-r---        10/19/2024  10:15 AM                Music
d-r---        10/19/2024  10:26 AM                OneDrive
d-r---        11/18/2024   3:58 PM                Pictures
d-----        11/17/2024   4:34 PM                PycharmProjects
d-r---        10/19/2024  10:15 AM                Saved Games
d-r---        10/19/2024  10:24 AM                Searches
d-----        10/20/2024   7:40 PM                source
d-r---        10/21/2024   1:10 PM                Videos
-a----        11/11/2024   9:28 PM          21908 .boto


PS C:\Users\patel> cd .\GenAIApplication\
PS C:\Users\patel\GenAIApplication>
```

3. Clone the sample application. We run the following command to clone the repository:



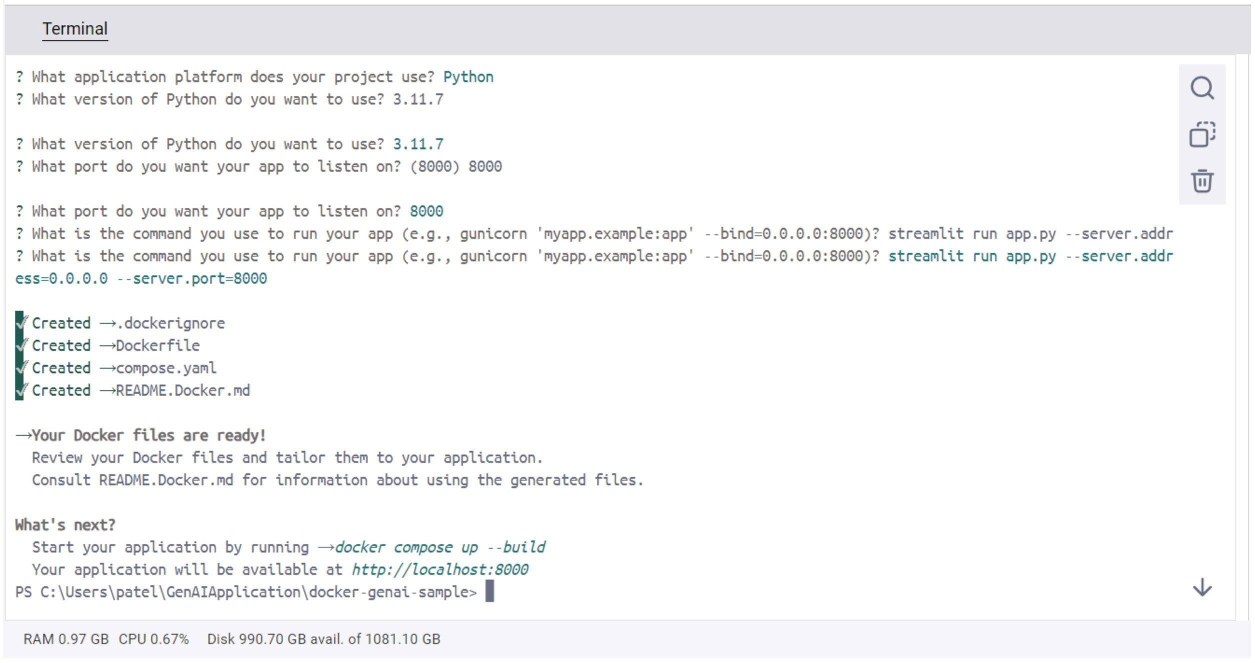
```
PS C:\Users\patel\GenAIApplication> git clone https://github.com/craig-osterhout/docker-genai-sample
Cloning into 'docker-genai-sample'...
remote: Enumerating objects: 11, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (10/10), done.
remote: Total 11 (delta 0), reused 11 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (11/11), 10.17 KiB | 10.17 MiB/s, done.
PS C:\Users\patel\GenAIApplication>
```

- You should now have the following files in your docker-genai-sample directory



```
PS C:\Users\patel\GenAIApplication> cd .\docker-genai-sample\
PS C:\Users\patel\GenAIApplication\docker-genai-sample> ls


    Directory: C:\Users\patel\GenAIApplication\docker-genai-sample


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        11/28/2024   1:31 PM           3895 app.py
-a----        11/28/2024   1:31 PM           9099 chains.py
-a----        11/28/2024   1:31 PM            967 env.example
-a----        11/28/2024   1:31 PM           7169 LICENSE
-a----        11/28/2024   1:31 PM            179 README.md
-a----        11/28/2024   1:31 PM            106 requirements.txt
-a----        11/28/2024   1:31 PM           1945 utils.py


PS C:\Users\patel\GenAIApplication\docker-genai-sample>
```

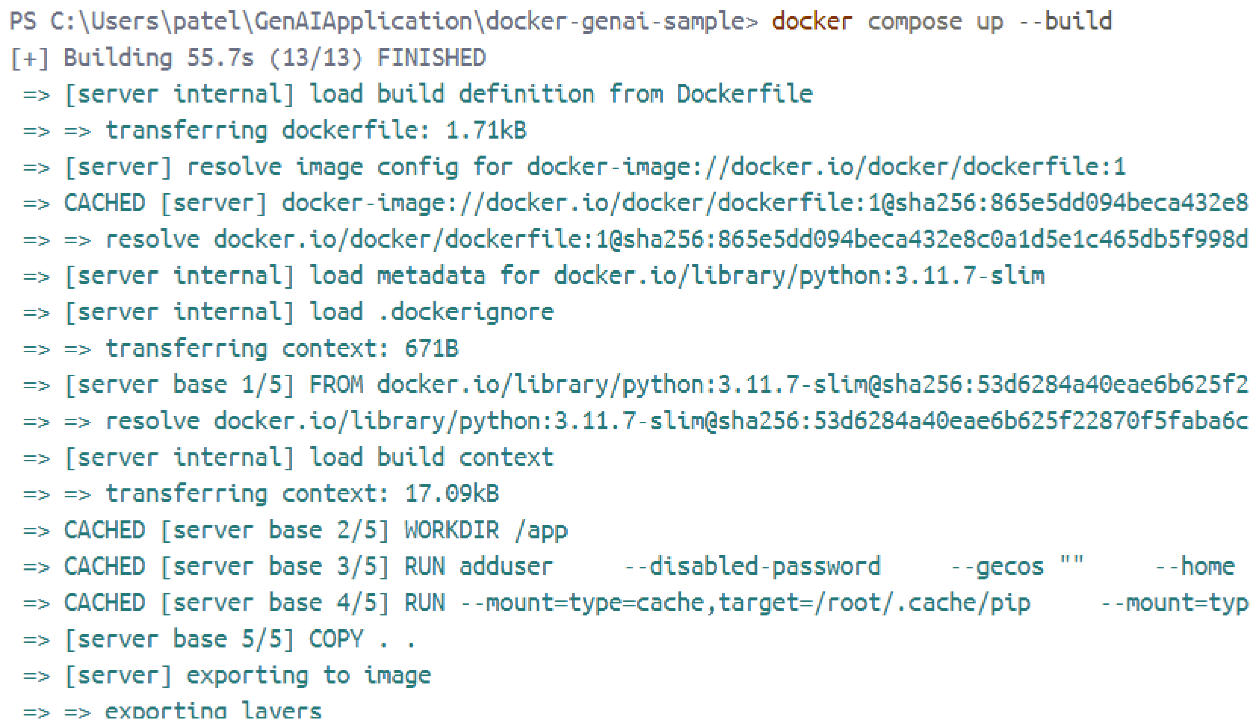
4. Now that we have an application, we can use docker init to create the necessary Docker assets to containerize our application. Inside the docker-genai-sample directory, run the docker init command.
   - docker init



```
Terminal

? What application platform does your project use? Python
? What version of Python do you want to use? 3.11.7

? What version of Python do you want to use? 3.11.7
? What port do you want your app to listen on? (8000) 8000

? What port do you want your app to listen on? 8000
? What is the command you use to run your app (e.g., gunicorn 'myapp.example:app' --bind=0.0.0.0:8000)? streamlit run app.py --server.addr
? What is the command you use to run your app (e.g., gunicorn 'myapp.example:app' --bind=0.0.0.0:8000)? streamlit run app.py --server.addr
ess=0.0.0.0 --server.port=8000

✓ Created →.dockerignore
✓ Created →Dockerfile
✓ Created →compose.yaml
✓ Created →README.Docker.md

→Your Docker files are ready!
  Review your Docker files and tailor them to your application.
  Consult README.Docker.md for information about using the generated files.

What's next?
  Start your application by running →docker compose up --build
  Your application will be available at http://localhost:8000
PS C:\Users\patel\GenAIApplication\docker-genai-sample>

RAM 0.97 GB  CPU 0.67%   Disk 990.70 GB avail. of 1081.10 GB
```

5. Next, for Docker to build and runs your application, run the following command in a terminal:

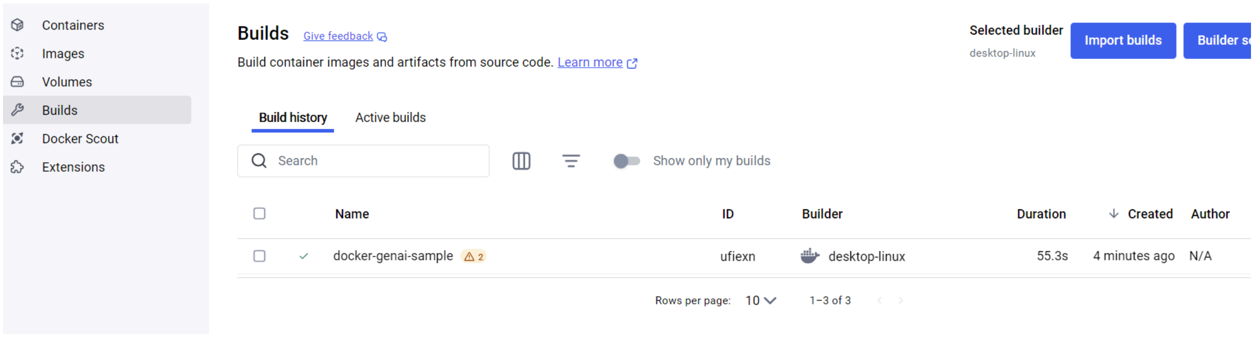
- docker compose up --build



```
PS C:\Users\patel\GenAIApplication\docker-genai-sample> docker compose up --build
[+] Building 55.7s (13/13) FINISHED
 => [server internal] load build definition from Dockerfile
 => => transferring dockerfile: 1.71kB
 => [server] resolve image config for docker-image://docker.io/docker/dockerfile:1
 => CACHED [server] docker-image://docker.io/docker/dockerfile:1@sha256:865e5dd094beca432e8
 => => resolve docker.io/docker/dockerfile:1@sha256:865e5dd094beca432e8c0a1d5e1c465db5f998d
 => [server internal] load metadata for docker.io/library/python:3.11.7-slim
 => [server internal] load .dockerignore
 => => transferring context: 671B
 => [server base 1/5] FROM docker.io/library/python:3.11.7-slim@sha256:53d6284a40eae6b625f2
 => => resolve docker.io/library/python:3.11.7-slim@sha256:53d6284a40eae6b625f22870f5faba6c
 => [server internal] load build context
 => => transferring context: 17.09kB
 => CACHED [server base 2/5] WORKDIR /app
 => CACHED [server base 3/5] RUN adduser     --disabled-password     --gecos ""     --home
 => CACHED [server base 4/5] RUN --mount=type=cache,target=/root/.cache/pip     --mount=typ
 => [server base 5/5] COPY . .
 => [server] exporting to image
 => => exporting layers
```

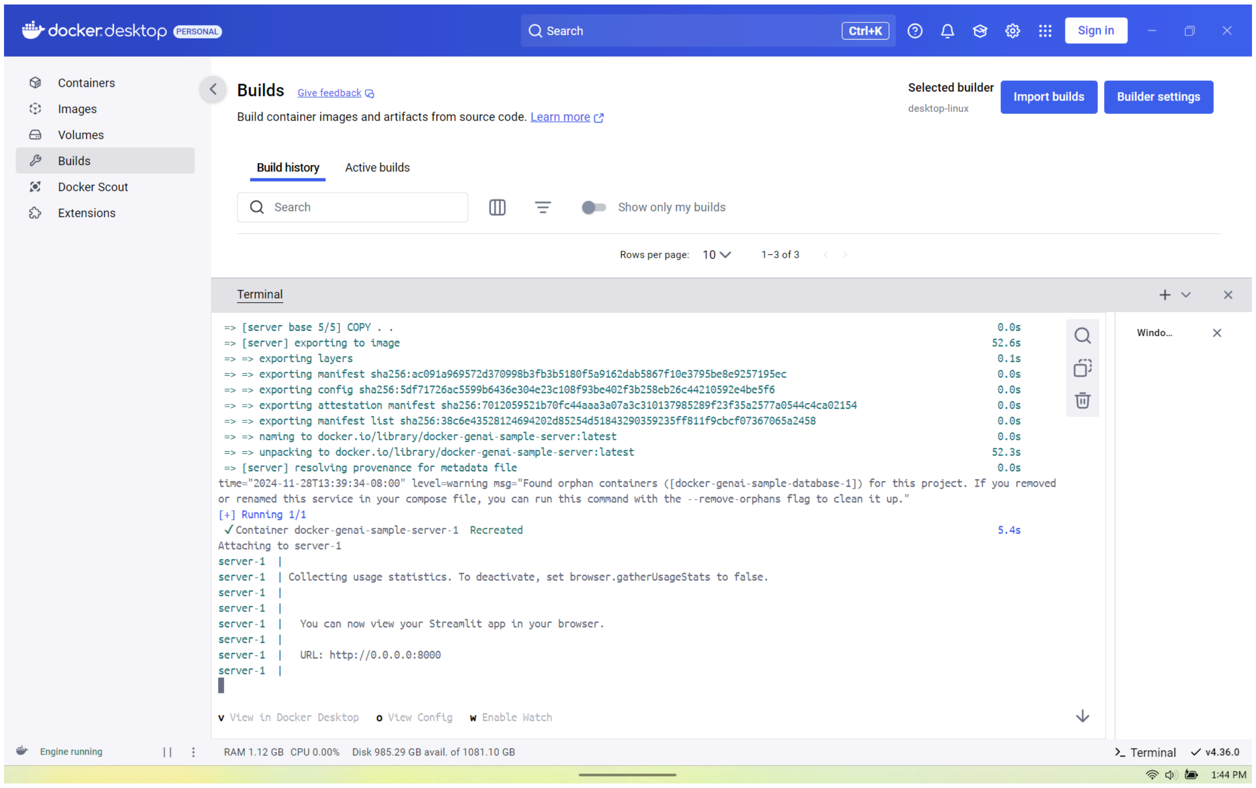
- We can also see the progress from Docker Desktop.

Project: GenAI - Containerize your app - optional



	Name	ID	Builder	Duration	Created	Author
✓	docker-genai-sample ⚠ 2	ufiexn	desktop-linux	55.3s	4 minutes ago	N/A

6. When the application is finally running, you will see a message like the following in the terminal. (Depending on your network connection, it may take several minutes to download all the dependencies.)

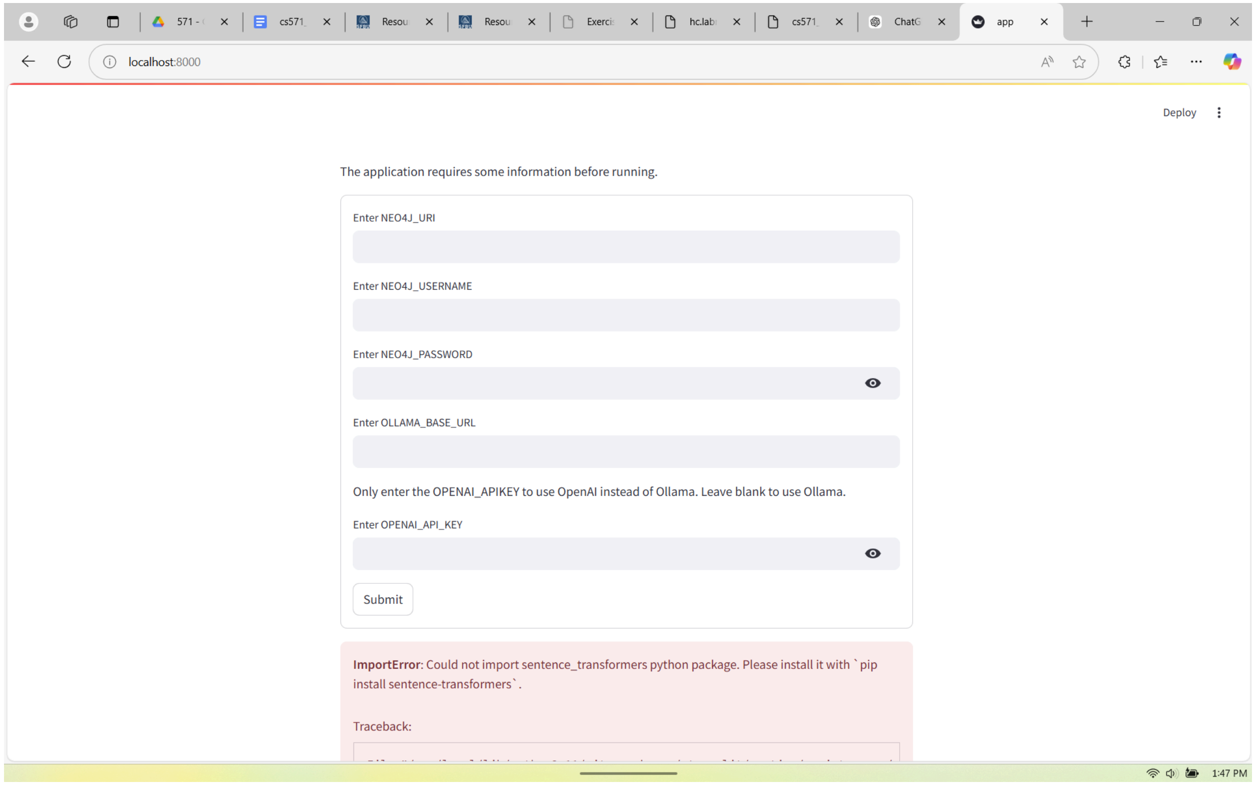


```
 => [server base 5/5] COPY . .                                                                                   0.0s
 => [server] exporting to image                                                                                  52.6s
 => => exporting layers                                                                                          0.1s
 => => exporting manifest sha256:ac091a969572d370998b3fb3b5180f5a9162dab5867f10e3795be8e9257195ec                0.0s
 => => exporting config sha256:5df71726ac5599b6436e304e23c108f93be402f3b258eb26c44210592e4be5f6                  0.0s
 => => exporting attestation manifest sha256:7012059521b70fc44aaa3a07a3c310137985289f23f35a2577a0544c4ca02154    0.0s
 => => exporting manifest list sha256:38c6e43528124694202d85254d51843290359235ff811f9cbcf07367065a2458           0.0s
 => => naming to docker.io/library/docker-genai-sample-server:latest                                             0.0s
 => => unpacking to docker.io/library/docker-genai-sample-server:latest                                          52.3s
 => [server] resolving provenance for metadata file                                                              0.0s
time="2024-11-28T13:39:34-08:00" level=warning msg="Found orphan containers ([docker-genai-sample-database-1]) for this project. If you removed
or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up."
[+] Running 1/1
 ✓Container docker-genai-sample-server-1  Recreated                                                              5.4s
Attaching to server-1
server-1  |
server-1  | Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.
server-1  |
server-1  |
server-1  |   You can now view your Streamlit app in your browser.
server-1  |
server-1  |   URL: http://0.0.0.0:8000
server-1  |


v View in Docker Desktop   o View Config   w Enable Watch
```

Then we open a browser and view the application at:

- http://localhost:8000

Project: GenAI - Containerize your app - optional



localhost:8000

The application requires some information before running.

Enter NEO4J_URI

Enter NEO4J_USERNAME

Enter NEO4J_PASSWORD

Enter OLLAMA_BASE_URL

Only enter the OPENAI_APIKEY to use OpenAI instead of Ollama. Leave blank to use Ollama.

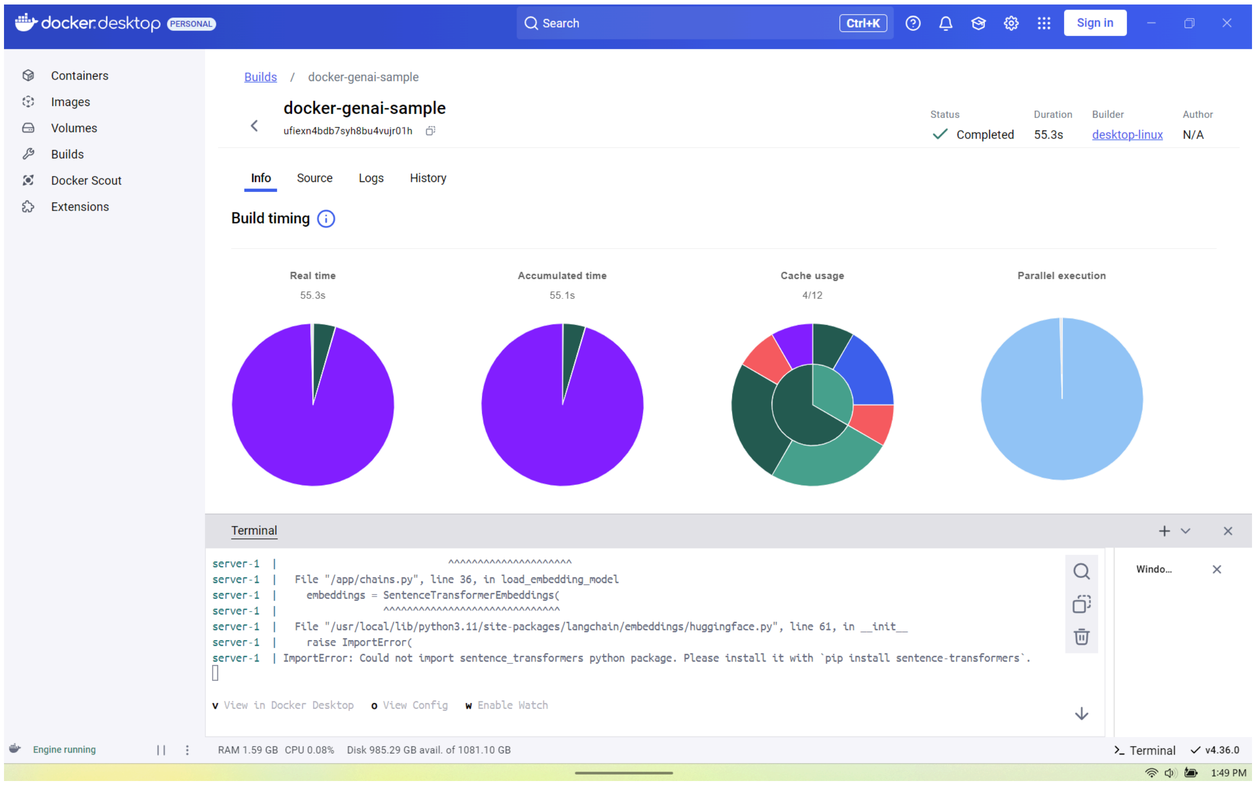
Enter OPENAI_API_KEY

Submit

ImportError: Could not import sentence_transformers python package. Please install it with `pip install sentence-transformers`.

Traceback:

- We can see the final completed build in docker here.



Builds / docker-genai-sample

docker-genai-sample
ufiexn4bdb7syh8bu4vujr01h

Status	Duration	Builder	Author
✓ Completed	55.3s	desktop-linux	N/A

Build timing

Real time 55.3s

Accumulated time 55.1s

Cache usage 4/12

Parallel execution

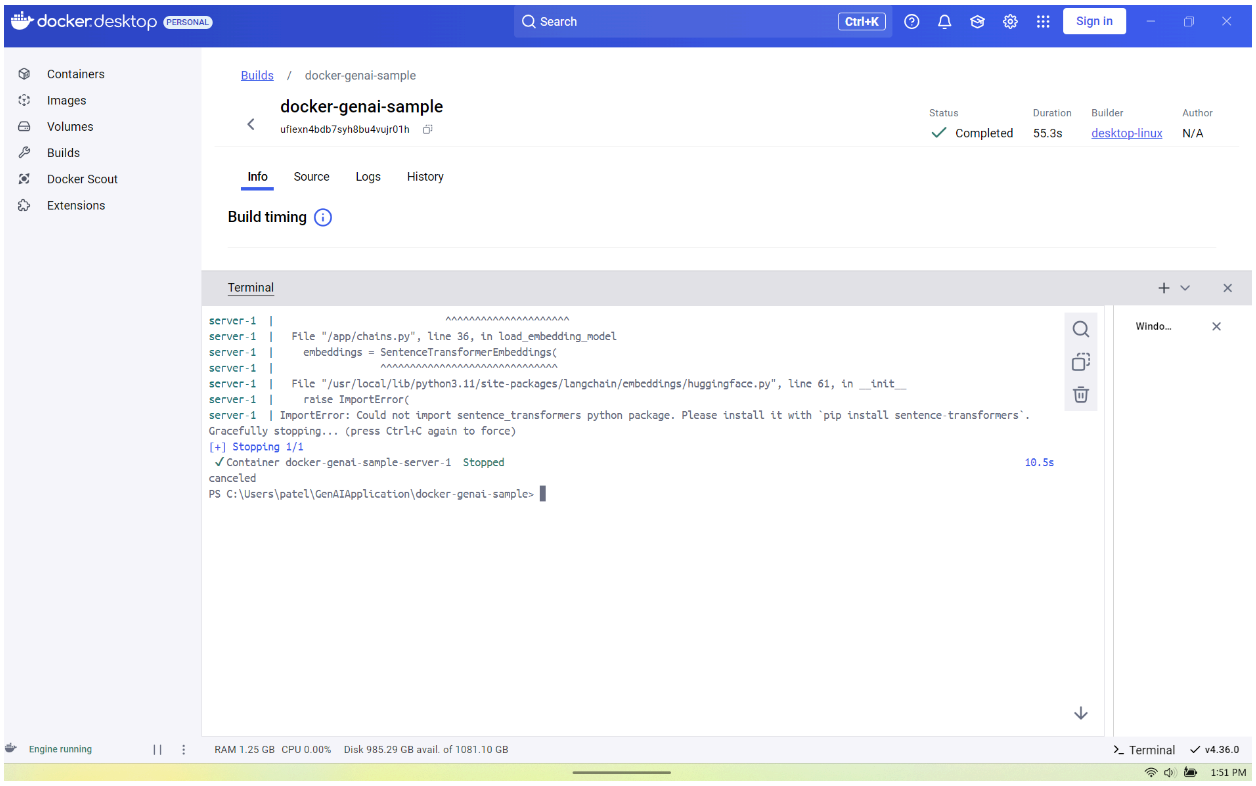
Terminal

```
server-1  |                                   ^^^^^^^^^^^^^^^^^^^^^
server-1  |   File "/app/chains.py", line 36, in load_embedding_model
server-1  |     embeddings = SentenceTransformerEmbeddings(
server-1  |                  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
server-1  |   File "/usr/local/lib/python3.11/site-packages/langchain/embeddings/huggingface.py", line 61, in __init__
server-1  |     raise ImportError(
server-1  | ImportError: Could not import sentence_transformers python package. Please install it with `pip install sentence-transformers`.

v View in Docker Desktop   o View Config   w Enable Watch
```

Engine running   RAM 1.59 GB CPU 0.08%   Disk 985.29 GB avail. of 1081.10 GB   v4.36.0

Project: GenAI - Containerize your app - optional

8. To stop the application, we press ctrl+C in the terminal.
   - ctrl+c



```
server-1  |                           ^^^^^^^^^^^^^^^^^^^^^
server-1  |   File "/app/chains.py", line 36, in load_embedding_model
server-1  |     embeddings = SentenceTransformerEmbeddings(
server-1  |                  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
server-1  |   File "/usr/local/lib/python3.11/site-packages/langchain/embeddings/huggingface.py", line 61, in __init__
server-1  |     raise ImportError(
server-1  | ImportError: Could not import sentence_transformers python package. Please install it with `pip install sentence-transformers`.
Gracefully stopping... (press Ctrl+C again to force)
[+] Stopping 1/1
 ✓Container docker-genai-sample-server-1  Stopped                                                              10.5s
canceled
PS C:\Users\patel\GenAIApplication\docker-genai-sample>
```