# Migrating the DogAPI to Docker

# Running in Docker before we will split it in two and migrate it to postgresSQL

A screenshot of a computer

Description automatically generated

A screen shot of a computer

Description automatically generated

Postgres database local, dogserver running as a docker image…  
  
Postgres is absolute garbage in the current release for Windows and nearly impossible to get working!!!! So I tried it on my Mac, it is not just garbage, but rotting garbage, so I went back to what was working…..  
  
A screenshot of a computer

Description automatically generated

# So this built nicely of I guess as nice as docker can do things.

*PS C:\Users\barte\OneDrive\Documents\UNO\CYBR84700 -Secure Web App\DogBreedExercise\dogserver> docker build -t dogserver .*

*[+] Building 1.0s (11/11) FINISHED docker:desktop-linux*

*=> [internal] load build definition from Dockerfile 0.0s*

*=> => transferring dockerfile: 538B 0.0s*

*=> [internal] load metadata for docker.io/library/python:3.11-slim 0.7s*

*=> [auth] library/python:pull token for registry-1.docker.io 0.0s*

*=> [internal] load .dockerignore 0.0s*

*=> => transferring context: 102B 0.0s*

*=> [1/5] FROM docker.io/library/python:3.11-slim@sha256:5501a4fe605abe24de87c2f3d6cf9fd760354416a0cad0296cf284fddcdca9e2 0.0s*

*=> => resolve docker.io/library/python:3.11-slim@sha256:5501a4fe605abe24de87c2f3d6cf9fd760354416a0cad0296cf284fddcdca9e2 0.0s*

*=> [internal] load build context 0.0s*

*=> => transferring context: 1.89kB 0.0s*

*=> CACHED [2/5] WORKDIR /app 0.0s*

*=> CACHED [3/5] COPY requirements.txt /app/ 0.0s*

*=> CACHED [4/5] RUN pip install --no-cache-dir -r requirements.txt 0.0s*

*=> CACHED [5/5] COPY . /app/ 0.0s*

*=> exporting to image 0.1s*

*=> => exporting layers 0.0s*

*=> => exporting manifest sha256:95ea4d2b9d0e98d1857e57d6984830e9c9e97958fe1de54e96d82e6a91929c44 0.0s*

*=> => exporting config sha256:88542f1ebcd2b7268e89e3e2ef17b7b67a9b6e798848ac3306c4721b0b624acb 0.0s*

*=> => exporting attestation manifest sha256:29e91506ffe2477b6eed35ca752323015219beeca2cc94c649397a468f383427 0.0s*

*=> => exporting manifest list sha256:22bf41aeeefd2c818072922790cd623cc8ac8991d2deb952e6766489950737e7 0.0s*

*=> => naming to docker.io/library/dogserver:latest 0.0s*

*=> => unpacking to docker.io/library/dogserver:latest 0.0s*

*View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/w6sswpzli5dlik5o4bojqbff4*

*2 warnings found (use docker --debug to expand):*

*- LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 5)*

*- LegacyKeyValueFormat: "ENV key=value" should be used instead of legacy "ENV key value" format (line 6)*

*What's next:*

*View a summary of image vulnerabilities and recommendations → docker scout quickview*

*PS C:\Users\barte\OneDrive\Documents\UNO\CYBR84700 -Secure Web App\DogBreedExercise\dogserver>*

So lets run it and watch Docker vomit all over the place.  
  
*PS C:\Users\barte\OneDrive\Documents\UNO\CYBR84700 -Secure Web App\DogBreedExercise\dogserver> docker run -p 8000:8000 dogserver*

*Watching for file changes with StatReloader*

*Performing system checks...*

*System check identified no issues (0 silenced).*

*Exception in thread django-main-thread:*

*Traceback (most recent call last):*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 279, in ensure\_connection*

*self.connect()*

*File "/usr/local/lib/python3.11/site-packages/django/utils/asyncio.py", line 26, in inner*

*return func(\*args, \*\*kwargs)*

*^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 256, in connect*

*self.connection = self.get\_new\_connection(conn\_params)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/utils/asyncio.py", line 26, in inner*

*return func(\*args, \*\*kwargs)*

*^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/postgresql/base.py", line 332, in get\_new\_connection*

*connection = self.Database.connect(\*\*conn\_params)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/psycopg2/\_\_init\_\_.py", line 122, in connect*

*conn = \_connect(dsn, connection\_factory=connection\_factory, \*\*kwasync)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*psycopg2.OperationalError: connection to server at "localhost" (::1), port 5432 failed: Connection refused*

*Is the server running on that host and accepting TCP/IP connections?*

*connection to server at "localhost" (127.0.0.1), port 5432 failed: Connection refused*

*Is the server running on that host and accepting TCP/IP connections?*

*The above exception was the direct cause of the following exception:*

*Traceback (most recent call last):*

*File "/usr/local/lib/python3.11/threading.py", line 1045, in \_bootstrap\_inner*

*self.run()*

*File "/usr/local/lib/python3.11/threading.py", line 982, in run*

*self.\_target(\*self.\_args, \*\*self.\_kwargs)*

*File "/usr/local/lib/python3.11/site-packages/django/utils/autoreload.py", line 64, in wrapper*

*fn(\*args, \*\*kwargs)*

*File "/usr/local/lib/python3.11/site-packages/django/core/management/commands/runserver.py", line 137, in inner\_run*

*self.check\_migrations()*

*File "/usr/local/lib/python3.11/site-packages/django/core/management/base.py", line 581, in check\_migrations*

*executor = MigrationExecutor(connections[DEFAULT\_DB\_ALIAS])*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/migrations/executor.py", line 18, in \_\_init\_\_*

*self.loader = MigrationLoader(self.connection)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/migrations/loader.py", line 58, in \_\_init\_\_*

*self.build\_graph()*

*File "/usr/local/lib/python3.11/site-packages/django/db/migrations/loader.py", line 235, in build\_graph*

*self.applied\_migrations = recorder.applied\_migrations()*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/migrations/recorder.py", line 89, in applied\_migrations*

*if self.has\_table():*

*^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/migrations/recorder.py", line 63, in has\_table*

*with self.connection.cursor() as cursor:*

*^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/utils/asyncio.py", line 26, in inner*

*return func(\*args, \*\*kwargs)*

*^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 320, in cursor*

*return self.\_cursor()*

*^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 296, in \_cursor*

*self.ensure\_connection()*

*File "/usr/local/lib/python3.11/site-packages/django/utils/asyncio.py", line 26, in inner*

*return func(\*args, \*\*kwargs)*

*^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 278, in ensure\_connection*

*with self.wrap\_database\_errors:*

*File "/usr/local/lib/python3.11/site-packages/django/db/utils.py", line 91, in \_\_exit\_\_*

*raise dj\_exc\_value.with\_traceback(traceback) from exc\_value*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 279, in ensure\_connection*

*self.connect()*

*File "/usr/local/lib/python3.11/site-packages/django/utils/asyncio.py", line 26, in inner*

*return func(\*args, \*\*kwargs)*

*^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/base/base.py", line 256, in connect*

*self.connection = self.get\_new\_connection(conn\_params)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/utils/asyncio.py", line 26, in inner*

*return func(\*args, \*\*kwargs)*

*^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/django/db/backends/postgresql/base.py", line 332, in get\_new\_connection*

*connection = self.Database.connect(\*\*conn\_params)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*File "/usr/local/lib/python3.11/site-packages/psycopg2/\_\_init\_\_.py", line 122, in connect*

*conn = \_connect(dsn, connection\_factory=connection\_factory, \*\*kwasync)*

*^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^*

*django.db.utils.OperationalError: connection to server at "localhost" (::1), port 5432 failed: Connection refused*

*Is the server running on that host and accepting TCP/IP connections?*

*connection to server at "localhost" (127.0.0.1), port 5432 failed: Connection refused*

*Is the server running on that host and accepting TCP/IP connections?*

# What’s not running on port 5432….??? It is but docker compose does not create the docker network that lets two docker images talk to each other.