

Student:

Andre Hardy

Email:

ahardy754@email.porterchester.edu

Time on Task:

28 hours, 35 minutes

Progress:

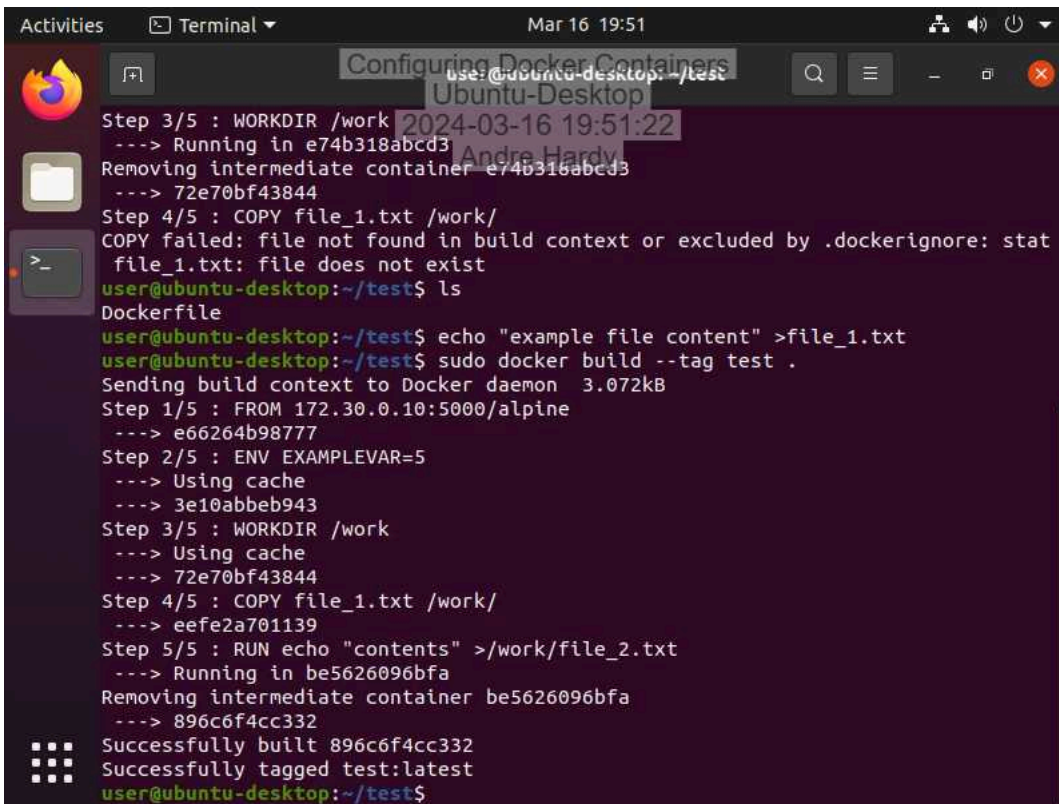
100%

Report Generated: Saturday, March 16, 2024 at 9:16 PM

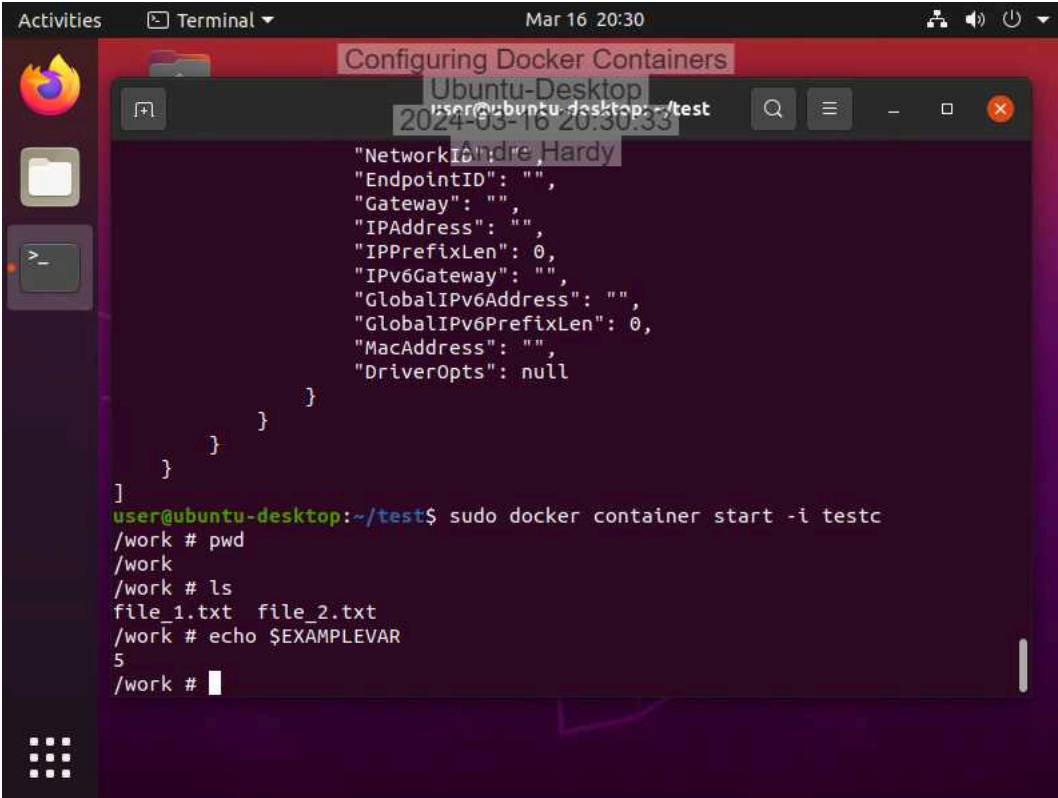
Hands-On Demonstration

Part 1: Creating and Running a Docker Container

9. Make a screen capture showing the **successful build of the test image**.

A terminal window titled 'Terminal' with a date and time of 'Mar 16 19:51'. The terminal shows the output of a Docker build command. It starts with 'Step 3/5 : WORKDIR /work', followed by '---> Running in e74b318abcd3', 'Removing intermediate container e74b318abcd3', and '---> 72e70bf43844'. Then 'Step 4/5 : COPY file_1.txt /work/' is shown, followed by an error: 'COPY failed: file not found in build context or excluded by .dockerignore: stat file_1.txt: file does not exist'. The user then runs 'ls' and 'echo "example file content" >file_1.txt'. Finally, 'sudo docker build --tag test .' is executed, showing 'Sending build context to Docker daemon 3.072kB', 'Step 1/5 : FROM 172.30.0.10:5000/alpine', '---> e66264b98777', 'Step 2/5 : ENV EXAMPLEVAR=5', '---> Using cache', '---> 3e10abbeb943', 'Step 3/5 : WORKDIR /work', '---> Using cache', '---> 72e70bf43844', 'Step 4/5 : COPY file_1.txt /work/', '---> eefe2a701139', 'Step 5/5 : RUN echo "contents" >/work/file_2.txt', '---> Running in be5626096bfa', 'Removing intermediate container be5626096bfa', '---> 896c6f4cc332', 'Successfully built 896c6f4cc332', 'Successfully tagged test:latest', and the prompt 'user@ubuntu-desktop:~/test\$'.

19. **Make a screen capture** showing the **working directory**, **file listing**, and **environment variable value**.

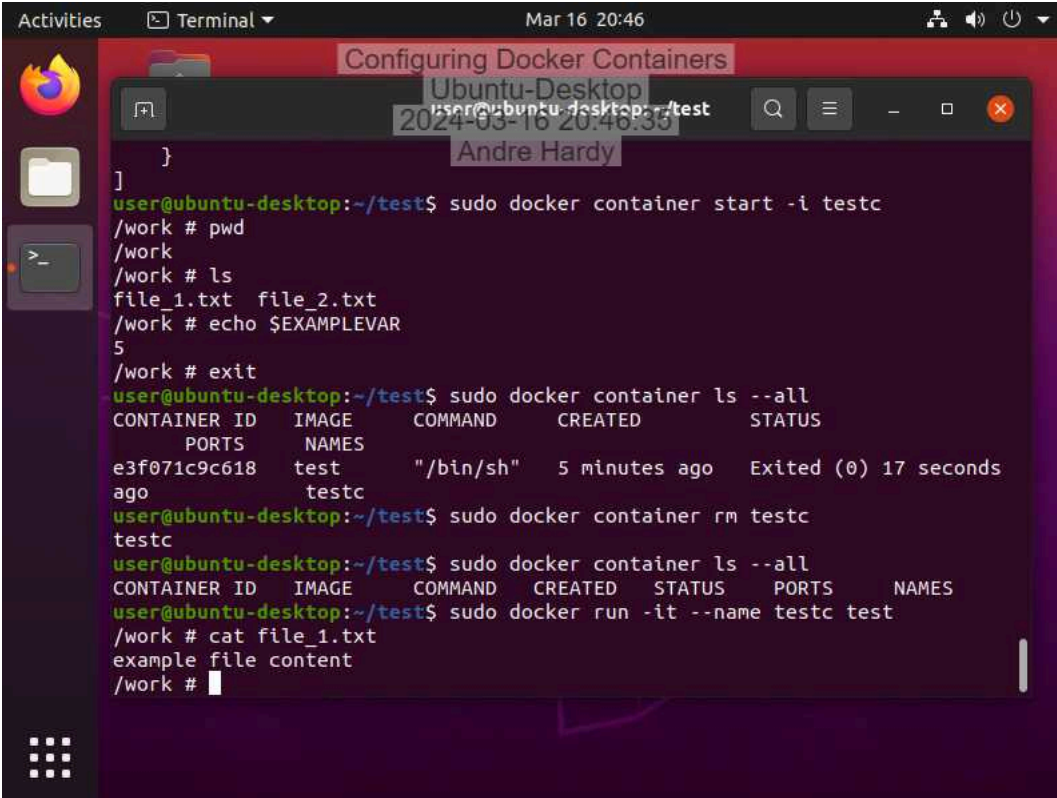


The screenshot shows a terminal window titled "Configuring Docker Containers" with the subtitle "Ubuntu-Desktop". The terminal displays the following commands and output:

```
user@ubuntu-desktop:~/test$ sudo docker container start -i testc
/work # pwd
/work
/work # ls
file_1.txt  file_2.txt
/work # echo $EXAMPLEVAR
5
/work #
```

The terminal window also shows a JSON configuration for a Docker container, including fields like "NetworkID", "EndpointID", "Gateway", "IPAddress", "IPPrefixLen", "IPv6Gateway", "GlobalIPv6Address", "GlobalIPv6PrefixLen", "MacAddress", and "DriverOpts".

26. Make a screen capture showing both your docker run command and the contents of the file_1.txt.

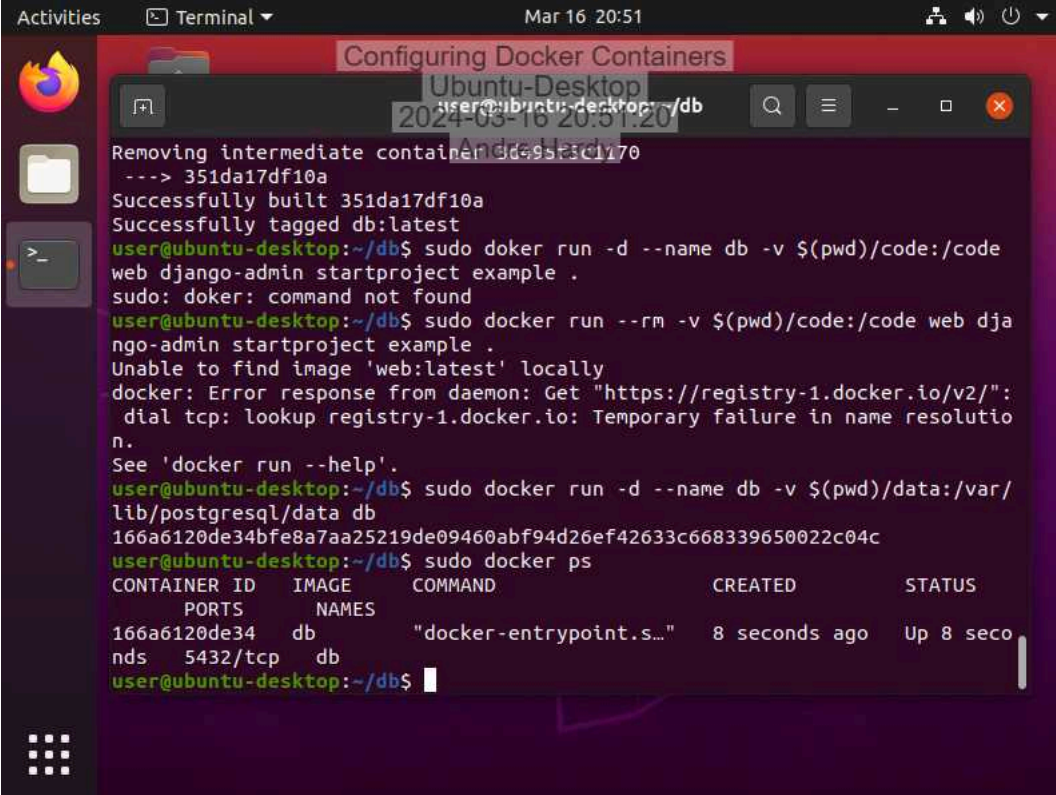


The screenshot shows a terminal window titled "Configuring Docker Containers" on an Ubuntu-Desktop. The user is logged in as "user" and is in the directory "~/test". The terminal output shows the following commands and results:

```
user@ubuntu-desktop:~/test$ sudo docker container start -i testc
/work # pwd
/work
/work # ls
file_1.txt  file_2.txt
/work # echo $EXAMPLEVAR
5
/work # exit
user@ubuntu-desktop:~/test$ sudo docker container ls --all
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
e3f071c9c618   test     "/bin/sh"  5 minutes ago   Exited (0) 17 seconds ago   testc
user@ubuntu-desktop:~/test$ sudo docker container rm testc
testc
user@ubuntu-desktop:~/test$ sudo docker container ls --all
CONTAINER ID   IMAGE     COMMAND   CREATED   STATUS    PORTS   NAMES
user@ubuntu-desktop:~/test$ sudo docker run -it --name testc test
/work # cat file_1.txt
example file content
/work #
```

Part 2: Configuring an Application with Multiple Containers

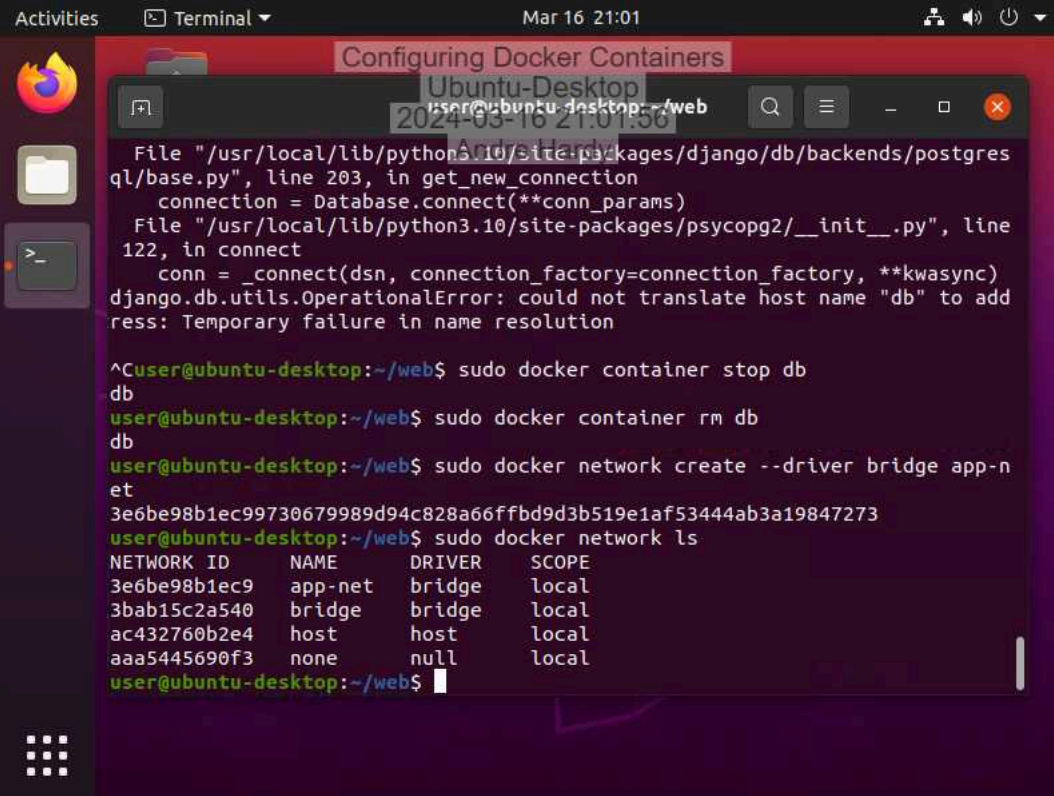
7. Make a screen capture showing the running db container.



The screenshot shows a terminal window titled 'Configuring Docker Containers' on an Ubuntu desktop. The terminal output shows the removal of an intermediate container, building and tagging a new image 'db:latest', and then running the container with the command 'sudo docker run -d --name db -v \$(pwd)/code:/code web django-admin startproject example .'. The container is successfully started and is running. The terminal also shows an attempt to run 'sudo docker ps' which failed with a 'command not found' error, followed by a successful 'docker ps' command that displays the running container's details.

```
Removing intermediate container 351da17df10a
--> 351da17df10a
Successfully built 351da17df10a
Successfully tagged db:latest
user@ubuntu-desktop:~/db$ sudo docker run -d --name db -v $(pwd)/code:/code
web django-admin startproject example .
sudo: docker: command not found
user@ubuntu-desktop:~/db$ sudo docker run --rm -v $(pwd)/code:/code web dja
ngo-admin startproject example .
Unable to find image 'web:latest' locally
docker: Error response from daemon: Get "https://registry-1.docker.io/v2/":
dial tcp: lookup registry-1.docker.io: Temporary failure in name resolutio
n.
See 'docker run --help'.
user@ubuntu-desktop:~/db$ sudo docker run -d --name db -v $(pwd)/data:/var/
lib/postgresql/data db
166a6120de34bfe8a7aa25219de09460abf94d26ef42633c668339650022c04c
user@ubuntu-desktop:~/db$ sudo docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS
PORTS         NAMES
166a6120de34   db       "docker-entrypoint.s..." 8 seconds ago   Up 8 seco
nds          5432/tcp    db
user@ubuntu-desktop:~/db$
```

24. Make a screen capture showing the command creating the network and the resulting id.



The screenshot shows a terminal window titled "Configuring Docker Containers" on an Ubuntu Desktop. The terminal displays the following commands and output:

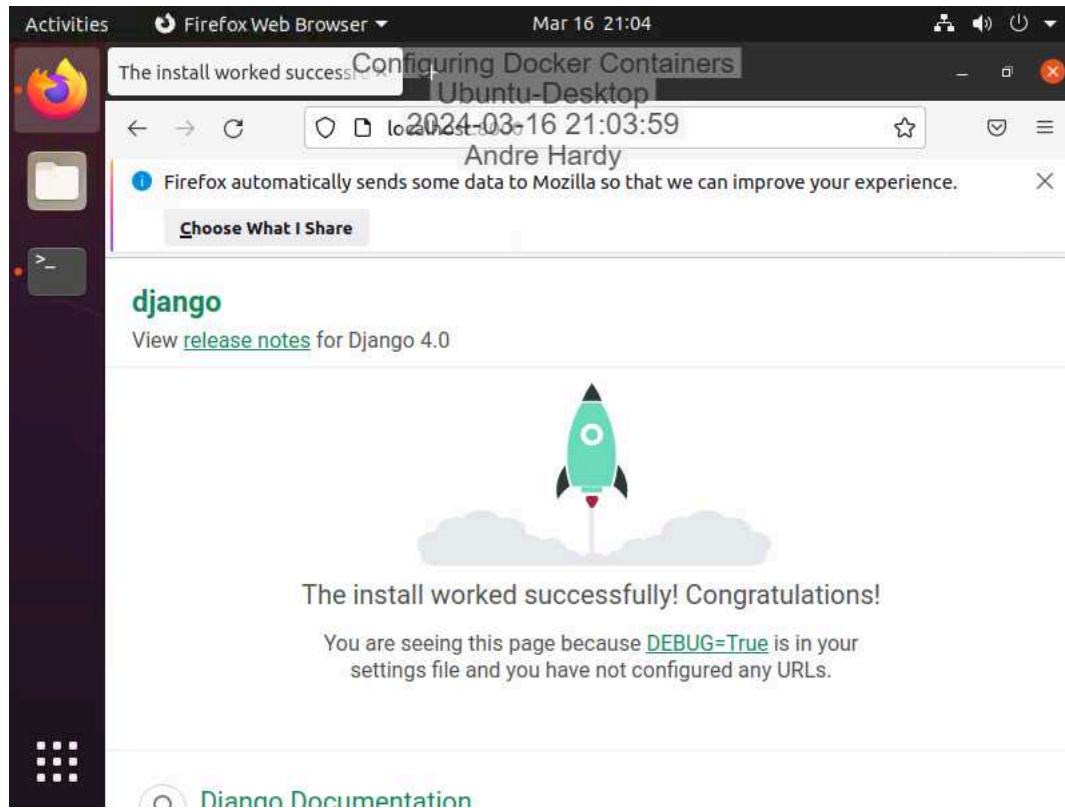
```
File "/usr/local/lib/python3.10/site-packages/django/db/backends/postgresql/base.py", line 203, in get_new_connection
    connection = Database.connect(**conn_params)
File "/usr/local/lib/python3.10/site-packages/psycopg2/__init__.py", line 122, in connect
    conn = _connect(dsn, connection_factory=connection_factory, **kwasync)
django.db.utils.OperationalError: could not translate host name "db" to address: Temporary failure in name resolution

^Cuser@ubuntu-desktop:~/web$ sudo docker container stop db
db
user@ubuntu-desktop:~/web$ sudo docker container rm db
db
user@ubuntu-desktop:~/web$ sudo docker network create --driver bridge app-net
3e6be98b1ec99730679989d94c828a66ffbd9d3b519e1af53444ab3a19847273
user@ubuntu-desktop:~/web$ sudo docker network ls
```

NETWORK ID	NAME	DRIVER	SCOPE
3e6be98b1ec9	app-net	bridge	local
3bab15c2a540	bridge	bridge	local
ac432760b2e4	host	host	local
aaa5445690f3	none	null	local

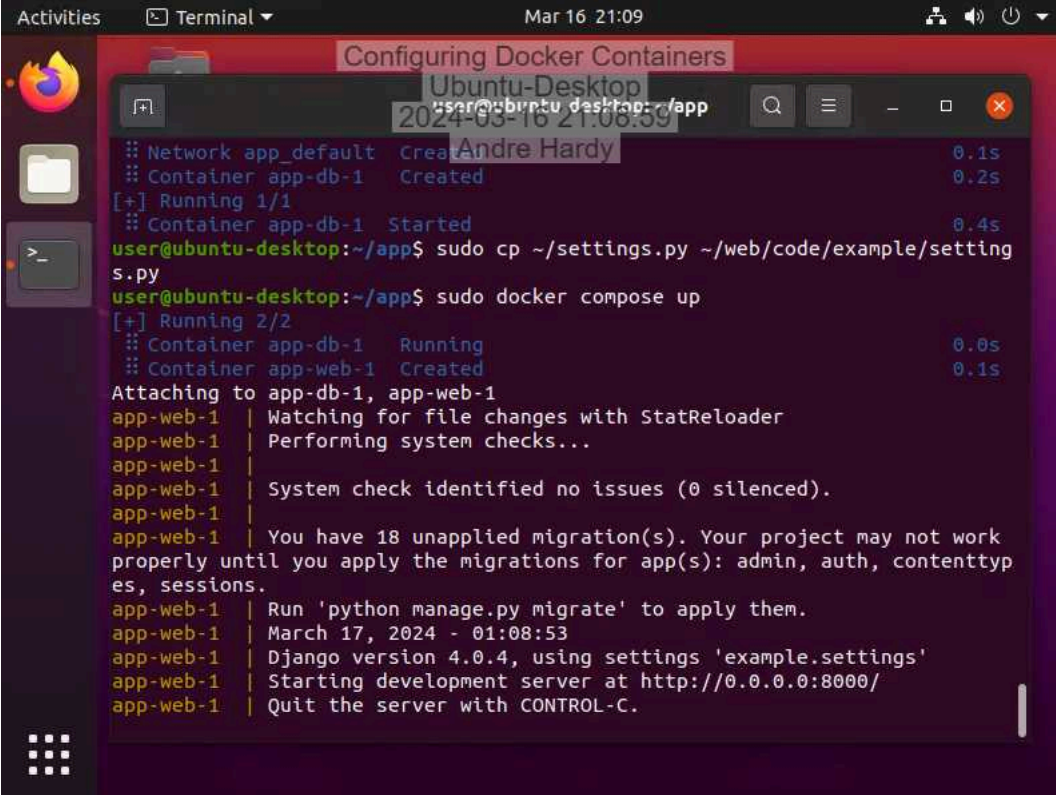
The terminal window also shows a file explorer in the background with the path `~/web` and a timestamp of `2024-03-16 21:01:56`.

30. Make a screen capture showing the Django page.



Part 3: Using Docker Compose

11. Make a screen capture showing output from the compose up command.

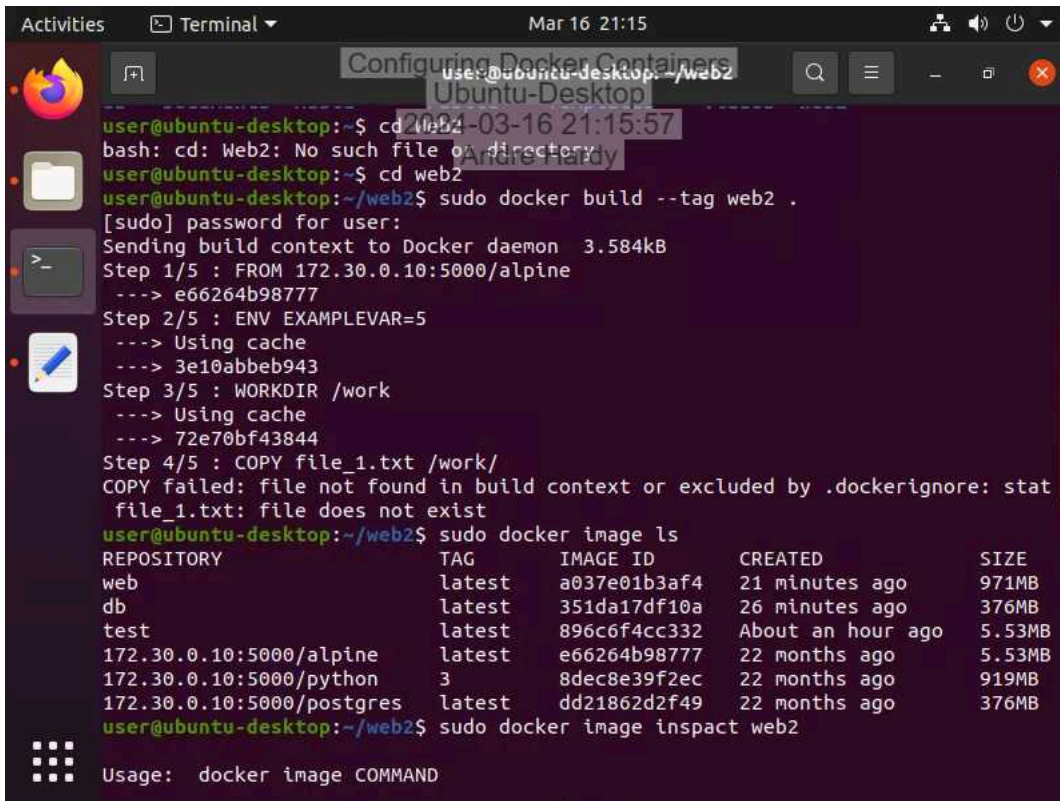
A terminal window titled 'Configuring Docker Containers' on an Ubuntu Desktop. The terminal shows the output of the 'docker compose up' command. It displays the creation and starting of containers 'app-db-1' and 'app-web-1'. The 'app-web-1' container logs show Django startup information, including the version (4.0.4) and the development server URL (http://0.0.0.0:8000/). The terminal also shows the user copying a file from the home directory to the web code directory.

```
user@ubuntu-desktop:~/app$ sudo docker compose up
[+] Running 2/2
  Container app-db-1   Running      0.0s
  Container app-web-1  Created      0.1s
Attaching to app-db-1, app-web-1
app-web-1 | Watching for file changes with StatReloader
app-web-1 | Performing system checks...
app-web-1 |
app-web-1 | System check identified no issues (0 silenced).
app-web-1 |
app-web-1 | You have 18 unapplied migration(s). Your project may not work
app-web-1 | properly until you apply the migrations for app(s): admin, auth, contenttyp
app-web-1 | es, sessions.
app-web-1 | Run 'python manage.py migrate' to apply them.
app-web-1 | March 17, 2024 - 01:08:53
app-web-1 | Django version 4.0.4, using settings 'example.settings'
app-web-1 | Starting development server at http://0.0.0.0:8000/
app-web-1 | Quit the server with CONTROL-C.
```

Challenge and Analysis

Part 1: Create an Image to Serve a Static File

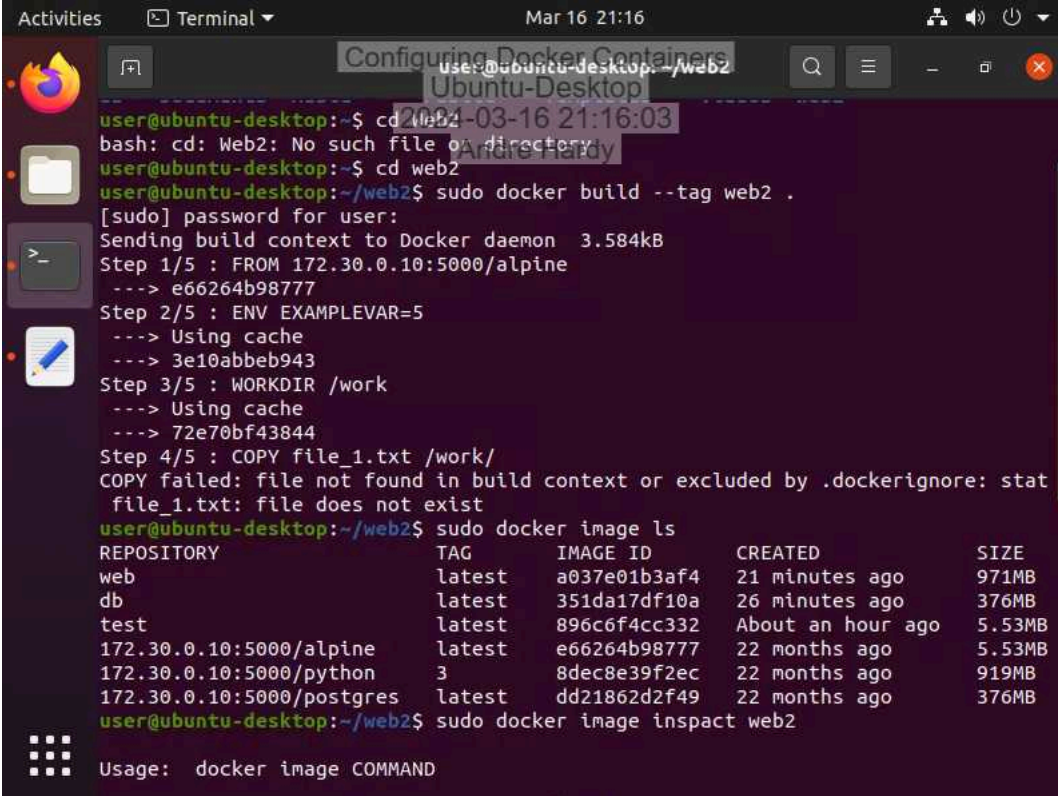
Make a screen capture showing the successful build of the web2 image.

A terminal window titled 'Terminal' with a date and time of 'Mar 16 21:15'. The user is at 'user@ubuntu-desktop: ~/web2'. The terminal shows the following commands and output:

```
user@ubuntu-desktop:~$ cd /web2
bash: cd: Web2: No such file or directory
user@ubuntu-desktop:~$ cd web2
user@ubuntu-desktop:~/web2$ sudo docker build --tag web2 .
[sudo] password for user:
Sending build context to Docker daemon 3.584kB
Step 1/5 : FROM 172.30.0.10:5000/alpine
--> e66264b98777
Step 2/5 : ENV EXAMPLEVAR=5
--> Using cache
--> 3e10abbeb943
Step 3/5 : WORKDIR /work
--> Using cache
--> 72e70bf43844
Step 4/5 : COPY file_1.txt /work/
COPY failed: file not found in build context or excluded by .dockerignore: stat
file_1.txt: file does not exist
user@ubuntu-desktop:~/web2$ sudo docker image ls
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
web                  latest      a037e01b3af4 21 minutes ago 971MB
db                   latest      351da17df10a 26 minutes ago 376MB
test                 latest      896c6f4cc332 About an hour ago 5.53MB
172.30.0.10:5000/alpine latest      e66264b98777 22 months ago 5.53MB
172.30.0.10:5000/python 3           8dec8e39f2ec 22 months ago 919MB
172.30.0.10:5000/postgres latest      dd21862d2f49 22 months ago 376MB
user@ubuntu-desktop:~/web2$ sudo docker image inspect web2
```

Usage: docker image COMMAND

Make a screen capture showing the **directory listing** served on port 8001.

A terminal window titled 'Terminal' with a date and time of 'Mar 16 21:16'. The prompt is 'user@ubuntu-desktop: ~/web2'. The user enters 'cd /web2', which results in a 'bash: cd: Web2: No such file or directory' error. The user then enters 'cd web2'. Next, the user runs 'sudo docker build --tag web2 .' and provides a password. The build process shows steps for FROM, ENV, WORKDIR, and COPY, with a final error: 'COPY failed: file not found in build context or excluded by .dockerignore: stat file_1.txt: file does not exist'. The user then runs 'sudo docker image ls', which displays a table of Docker images. Finally, the user enters 'sudo docker image inspect web2', and the terminal shows the usage for the 'docker image inspect' command.

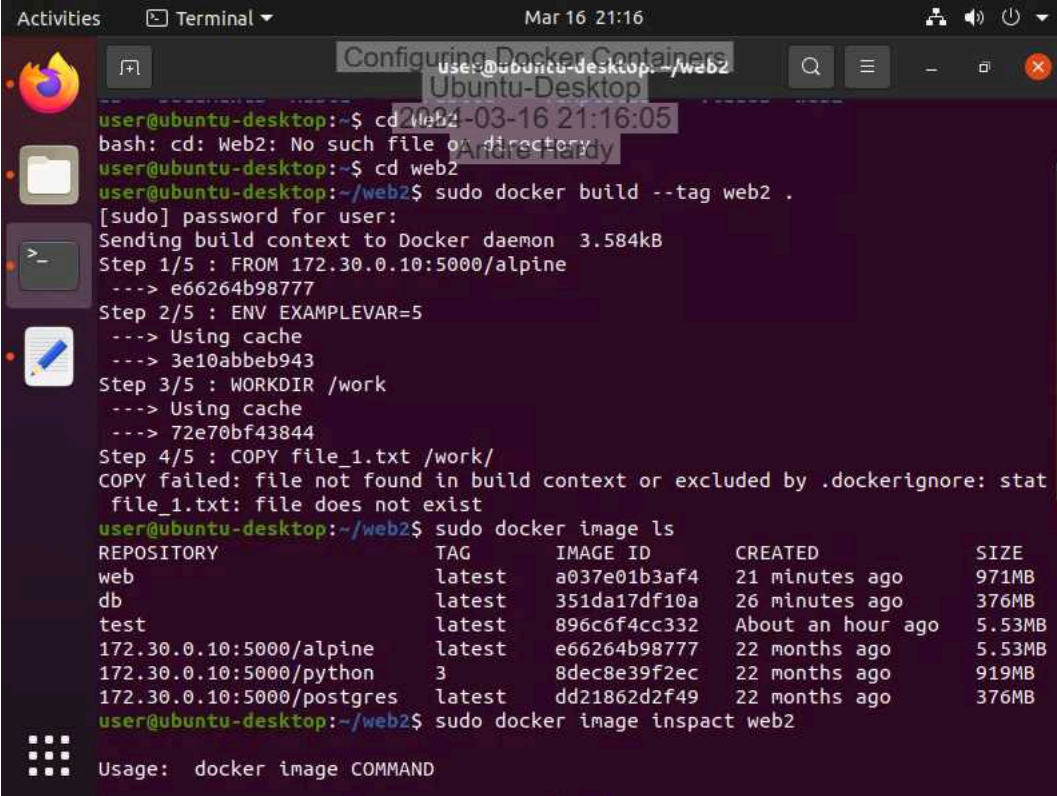
```
user@ubuntu-desktop:~$ cd /web2
bash: cd: Web2: No such file or directory
user@ubuntu-desktop:~$ cd web2
user@ubuntu-desktop:~/web2$ sudo docker build --tag web2 .
[sudo] password for user:
Sending build context to Docker daemon  3.584kB
Step 1/5 : FROM 172.30.0.10:5000/alpine
--> e66264b98777
Step 2/5 : ENV EXAMPLEVAR=5
--> Using cache
--> 3e10abbeb943
Step 3/5 : WORKDIR /work
--> Using cache
--> 72e70bf43844
Step 4/5 : COPY file_1.txt /work/
COPY failed: file not found in build context or excluded by .dockerignore: stat
file_1.txt: file does not exist
user@ubuntu-desktop:~/web2$ sudo docker image ls
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
web                  latest      a037e01b3af4  21 minutes ago 971MB
db                   latest      351da17df10a  26 minutes ago 376MB
test                latest      896c6f4cc332  About an hour ago 5.53MB
172.30.0.10:5000/alpine latest      e66264b98777  22 months ago 5.53MB
172.30.0.10:5000/python 3           8dec8e39f2ec  22 months ago 919MB
172.30.0.10:5000/postgres latest      dd21862d2f49  22 months ago 376MB
user@ubuntu-desktop:~/web2$ sudo docker image inspect web2
Usage:  docker image COMMAND
```

Part 2: Add the Static Service to the Docker Compose File

Configuring Docker Containers

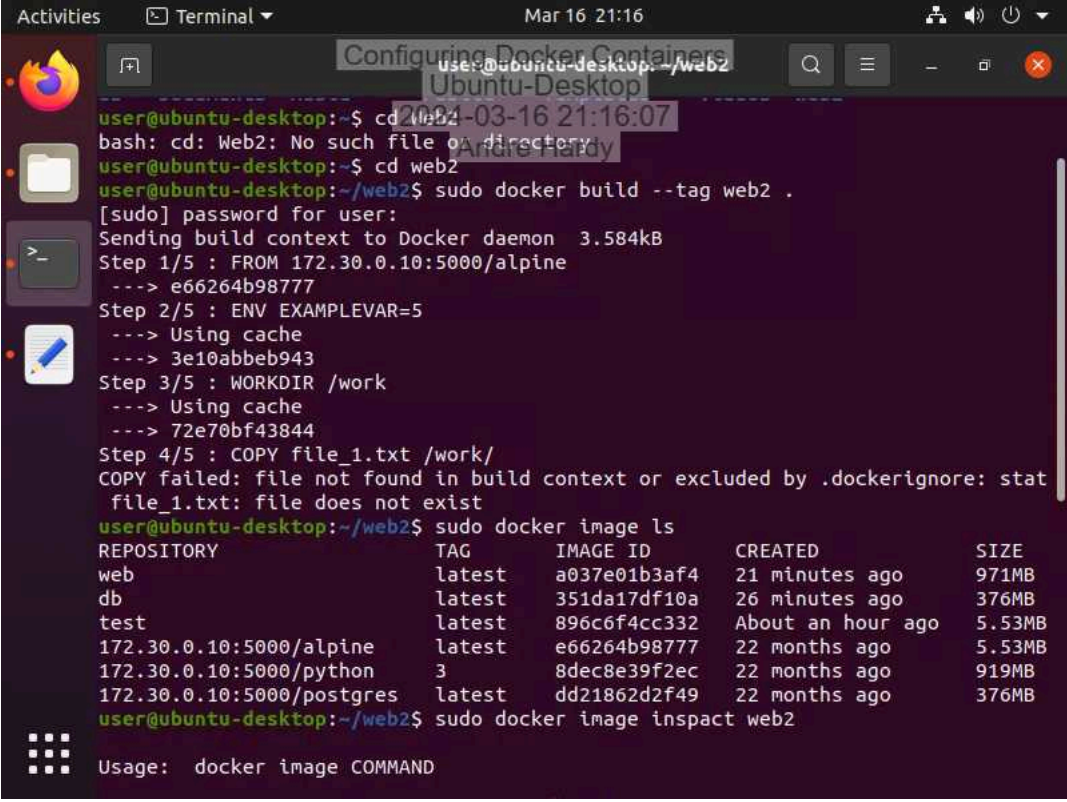
Cloud Computing, Second Edition - Lab 03

Make a screen capture showing the **creation of the web, web2, and db containers** in the docker compose output.

A terminal window titled 'Terminal' with a date and time of 'Mar 16 21:16'. The prompt is 'user@ubuntu-desktop: ~/web2'. The user enters 'cd /web2', which results in a 'bash: cd: /web2: No such file or directory' error. Then, the user enters 'cd web2'. Next, the user runs 'sudo docker build --tag web2 .' and provides a password. The build process shows steps for FROM, ENV, WORKDIR, and COPY, with the COPY step failing due to a missing file. Finally, the user runs 'sudo docker image ls' and 'sudo docker image inspect web2'.

```
user@ubuntu-desktop:~$ cd /web2
bash: cd: /web2: No such file or directory
user@ubuntu-desktop:~$ cd web2
user@ubuntu-desktop:~/web2$ sudo docker build --tag web2 .
[sudo] password for user:
Sending build context to Docker daemon  3.584kB
Step 1/5 : FROM 172.30.0.10:5000/alpine
--> e66264b98777
Step 2/5 : ENV EXAMPLEVAR=5
--> Using cache
--> 3e10abbeb943
Step 3/5 : WORKDIR /work
--> Using cache
--> 72e70bf43844
Step 4/5 : COPY file_1.txt /work/
COPY failed: file not found in build context or excluded by .dockerignore: stat
file_1.txt: file does not exist
user@ubuntu-desktop:~/web2$ sudo docker image ls
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
web                  latest      a037e01b3af4  21 minutes ago 971MB
db                   latest      351da17df10a  26 minutes ago 376MB
test                latest      896c6f4cc332  About an hour ago 5.53MB
172.30.0.10:5000/alpine latest      e66264b98777  22 months ago 5.53MB
172.30.0.10:5000/python 3           8dec8e39f2ec  22 months ago 919MB
172.30.0.10:5000/postgres latest      dd21862d2f49  22 months ago 376MB
user@ubuntu-desktop:~/web2$ sudo docker image inspect web2
Usage:  docker image COMMAND
```

Make a screen capture showing the **directory listing** served on port 8080.

A terminal window titled 'Terminal' with a date and time of 'Mar 16 21:16'. The terminal shows a user at 'ubuntu-desktop' navigating to a directory and running a Docker build command. The build process shows steps for context, environment variables, workdir, and copying a file. The build fails due to a missing file. The user then lists Docker images, showing a table with repository, tag, image ID, creation time, and size. Finally, the user runs the 'inspect' command on the 'web2' image.

```
user@ubuntu-desktop:~$ cd /web2
bash: cd: Web2: No such file or directory
user@ubuntu-desktop:~$ cd web2
user@ubuntu-desktop:~/web2$ sudo docker build --tag web2 .
[sudo] password for user:
Sending build context to Docker daemon 3.584kB
Step 1/5 : FROM 172.30.0.10:5000/alpine
--> e66264b98777
Step 2/5 : ENV EXAMPLEVAR=5
--> Using cache
--> 3e10abbeb943
Step 3/5 : WORKDIR /work
--> Using cache
--> 72e70bf43844
Step 4/5 : COPY file_1.txt /work/
COPY failed: file not found in build context or excluded by .dockerignore: stat
file_1.txt: file does not exist
user@ubuntu-desktop:~/web2$ sudo docker image ls
REPOSITORY          TAG         IMAGE ID      CREATED        SIZE
web                  latest      a037e01b3af4  21 minutes ago 971MB
db                   latest      351da17df10a  26 minutes ago 376MB
test                 latest      896c6f4cc332  About an hour ago 5.53MB
172.30.0.10:5000/alpine latest      e66264b98777  22 months ago 5.53MB
172.30.0.10:5000/python 3           8dec8e39f2ec  22 months ago 919MB
172.30.0.10:5000/postgres latest      dd21862d2f49  22 months ago 376MB
user@ubuntu-desktop:~/web2$ sudo docker image inspect web2
Usage:  docker image COMMAND
```