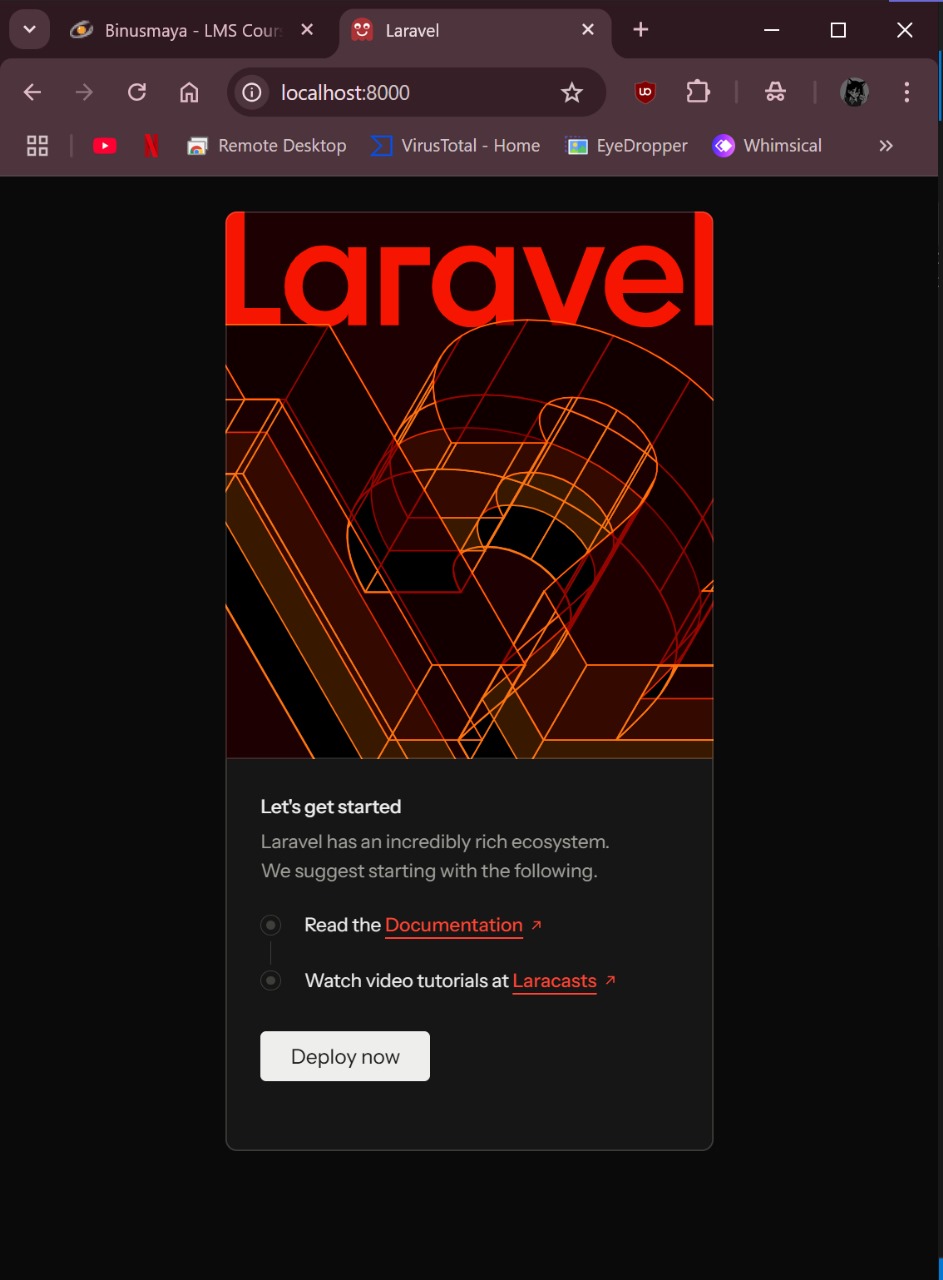
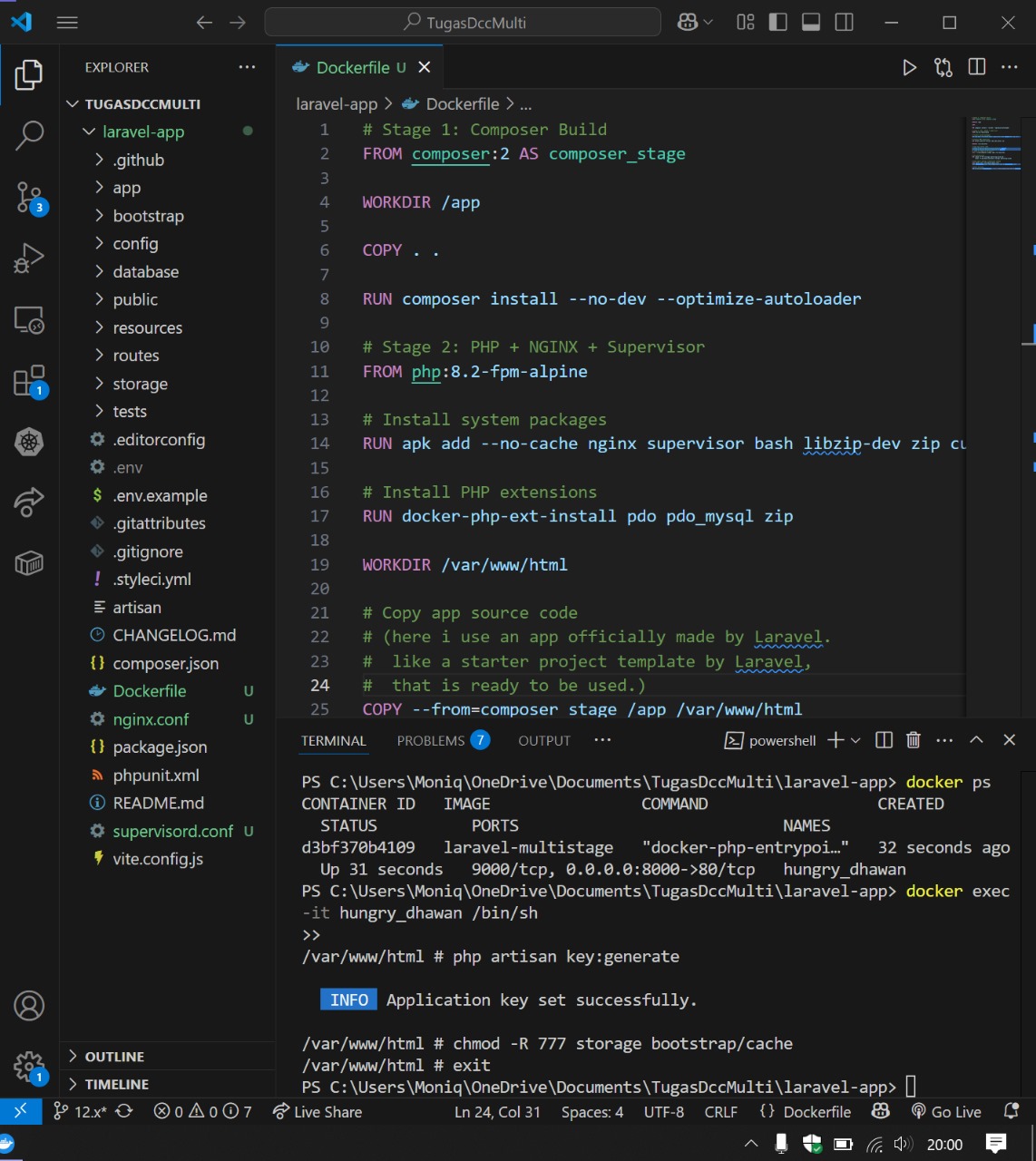
# **Docker Challenge (Multi-staged Docker Build)**

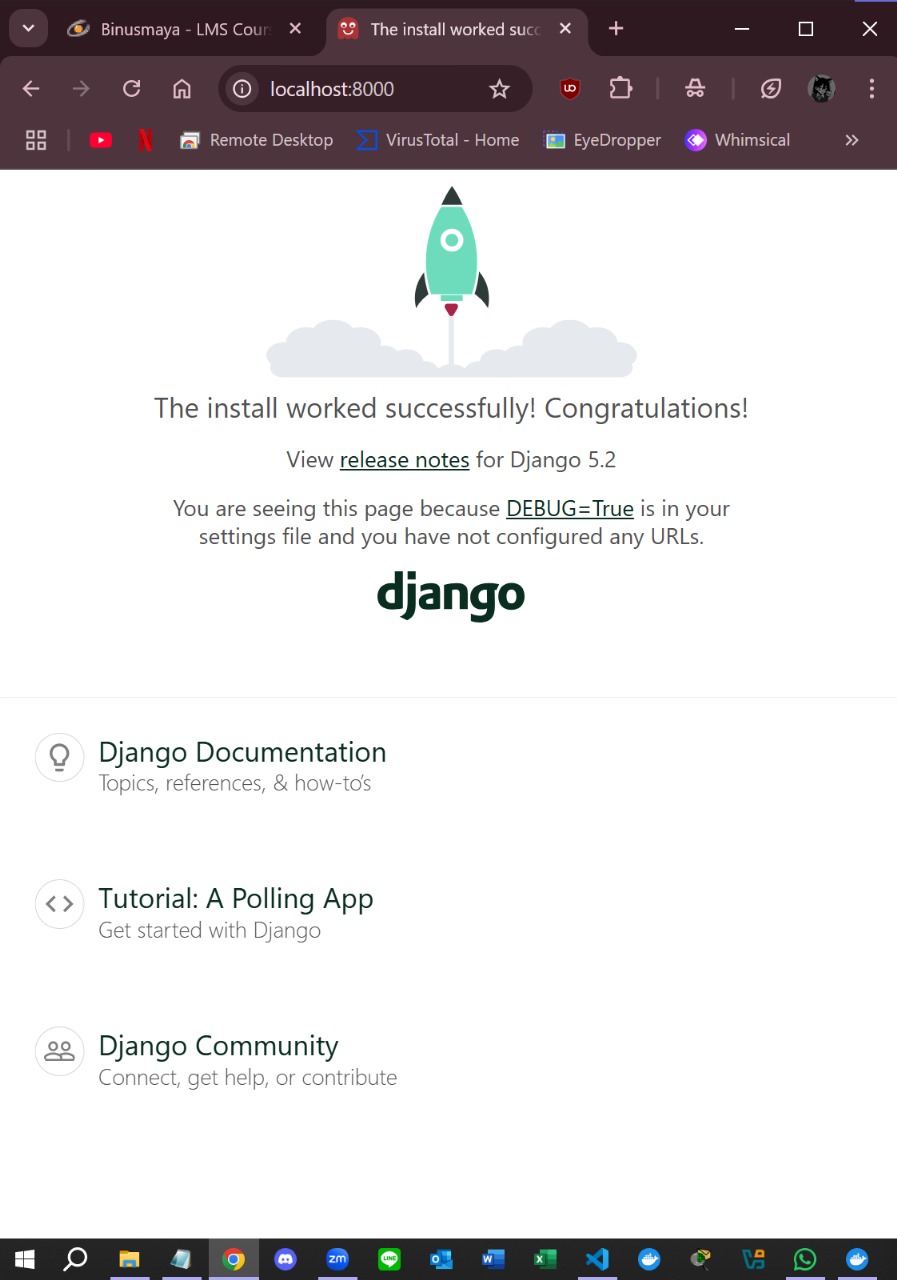
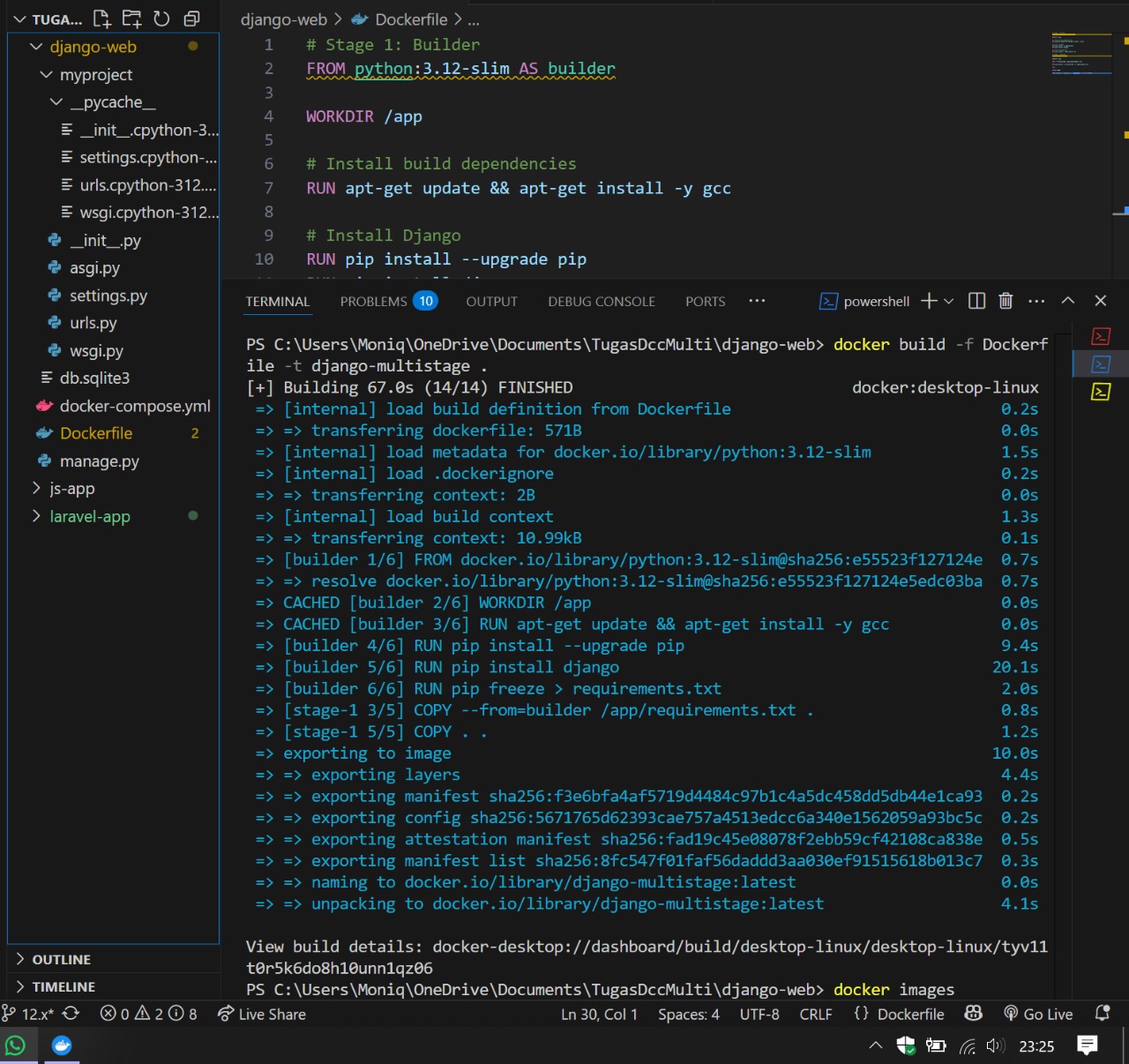
Monique Cloudya Bagensa

2702279621

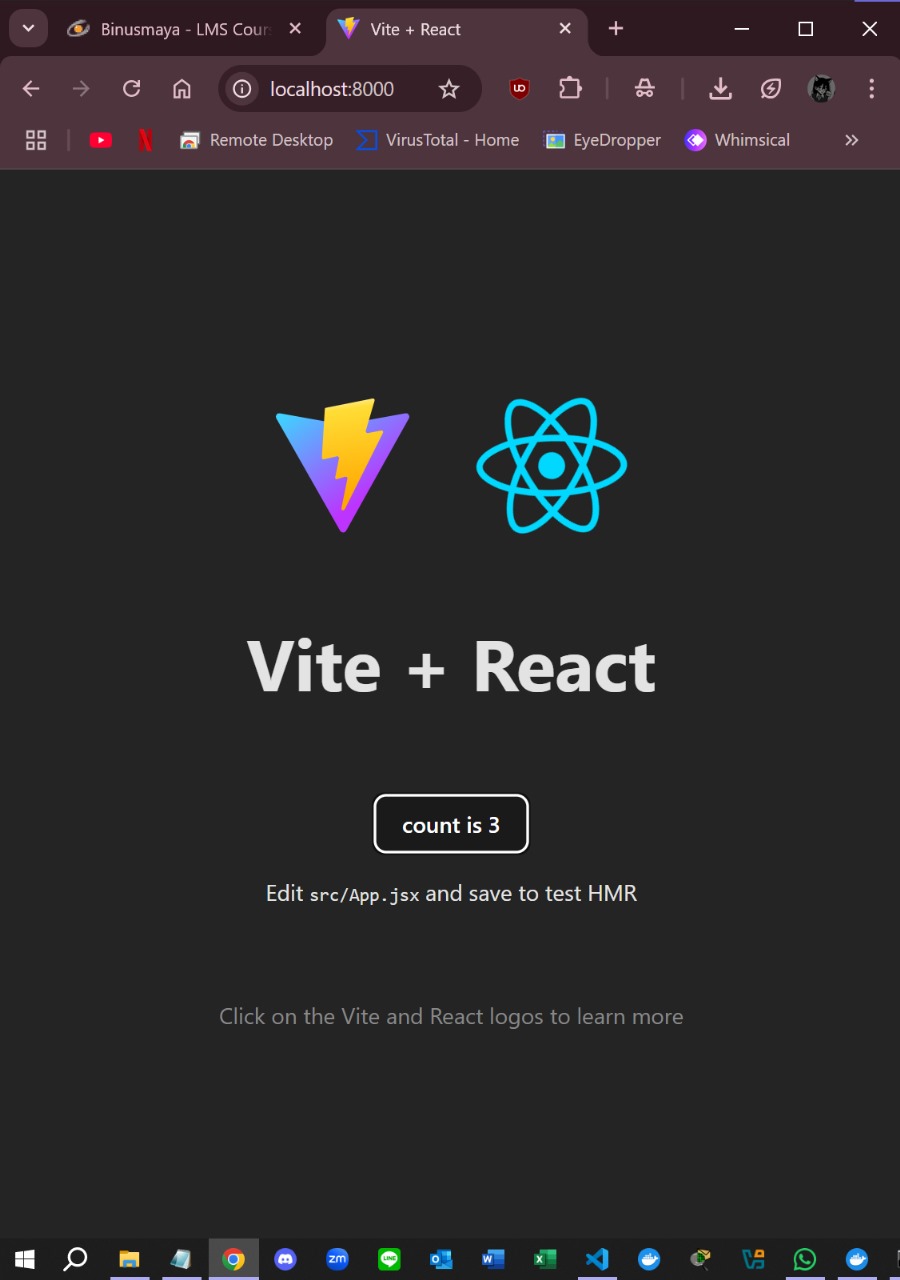
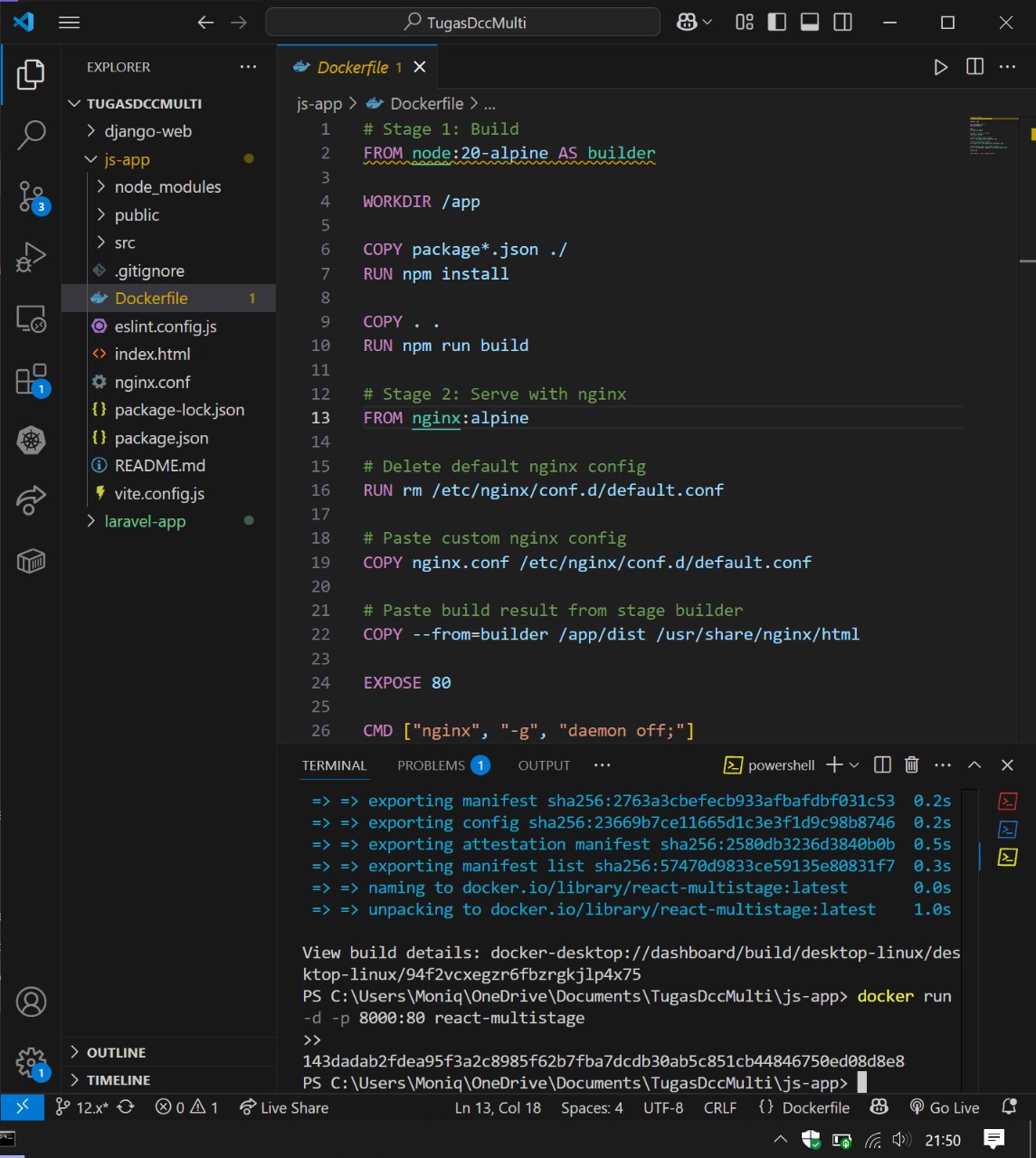
1. **php -> basic laravel app**

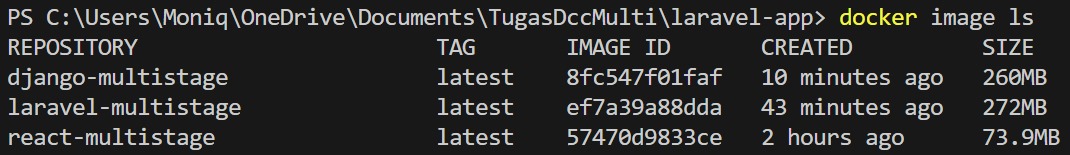


1. **Python Django -> basic web server with Django Framework**



1. **JS Framework (bisa framework apapun) -> basic app**

****

**Sizes:  
**

**Steps to build and run**

**PHP -> Basic laravel app:**

# Buka terminal VSCode

# Change directory dulu ke laravel-app

1. docker build -t laravel-multistage .

2. docker run -d -p 8000:80 laravel-multistage

Kalau berhasil, lanjut step 3. Tapi kalau tidak bisa:

1. Cek container:

docker ps

2. Kalau ada, stop dulu:

docker stop [ContainerID or ContainerName]

3. Lalu run ulang step 2.

3. docker exec -it [ContainerID or ContainerName] /bin/sh

4. php artisan key:generate

5. chmod -R 777 storage bootstrap/cache

6. exit

7. Cek size: docker images

8. Buka browser -> http://localhost:8000

**Python Django -> Basic web server with Django Framework:**

# Buka terminal VSCode

# Change directory dulu ke django-web

# Stop container yang sedang berjalan di folder yang lain terlebih dahulu, karena portnya sama.

1. docker build -t django-multistage .

2. docker run -d -p 8000:8000 django-multistage

3. Refresh kembali web http://localhost:8000 yang terbuka sebelumnya.

4. Stop container:

docker stop [ContainerID or ContainerName]

**JS Framework (React + Vite) -> basic app**

# Buka terminal VSCode

# Change directory dulu ke js-app

# Stop container yang sedang berjalan di folder yang lain terlebih dahulu, karena portnya sama.

1. docker build -t react-multistage .

2. docker run -d -p 8000:80 react-multistage

3. Refresh kembali web http://localhost:8000 yang terbuka sebelumnya.

4. Stop container:

docker stop [ContainerID or ContainerName]