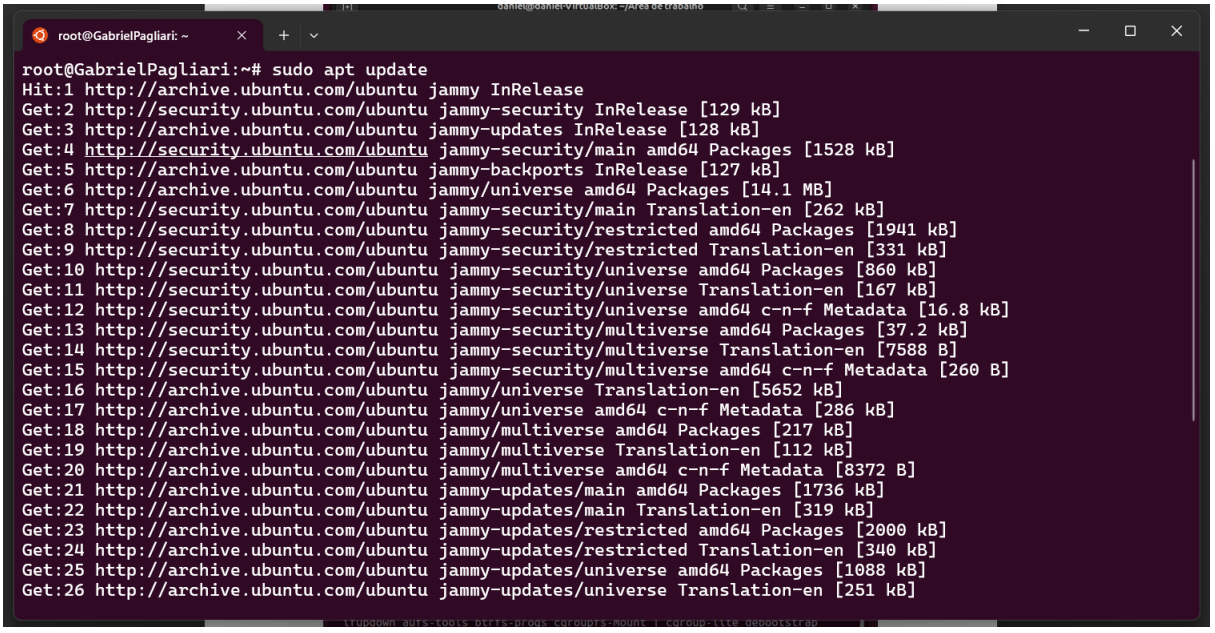
Ambiente Docker

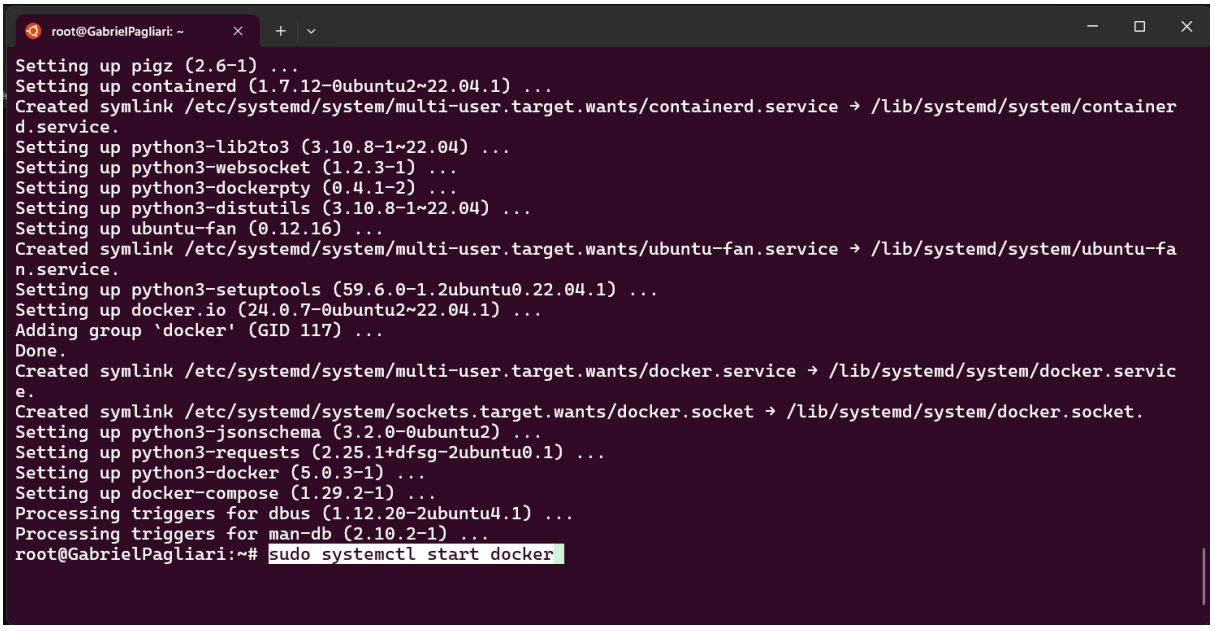
Primeiro Passo - Instalações
sudo apt update

sudo apt install -y docker.io docker-compose

sudo systemctl start docker

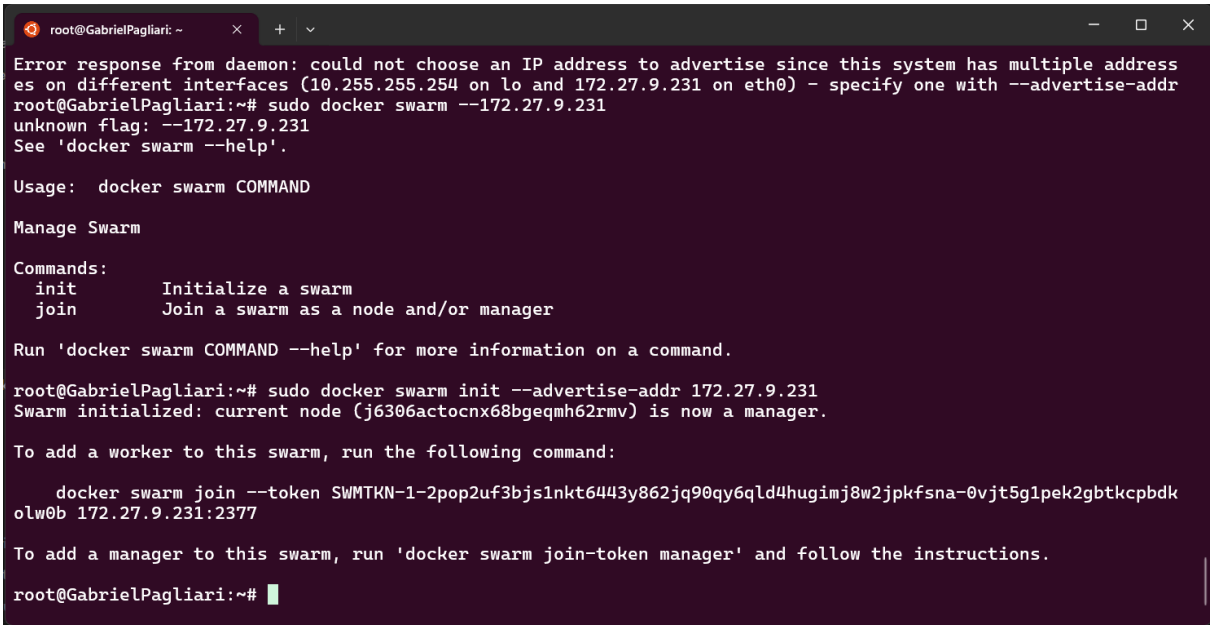
sudo systemctl enable docker





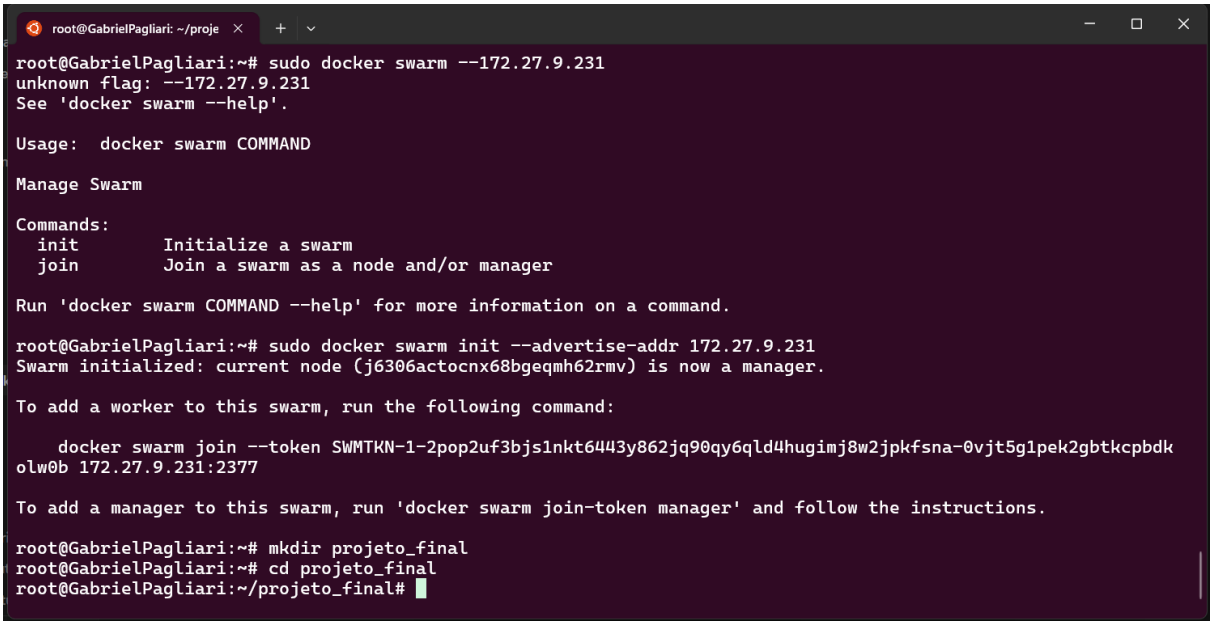
Passo 2 - Iniciar o Swarm

sudo docker swarm init

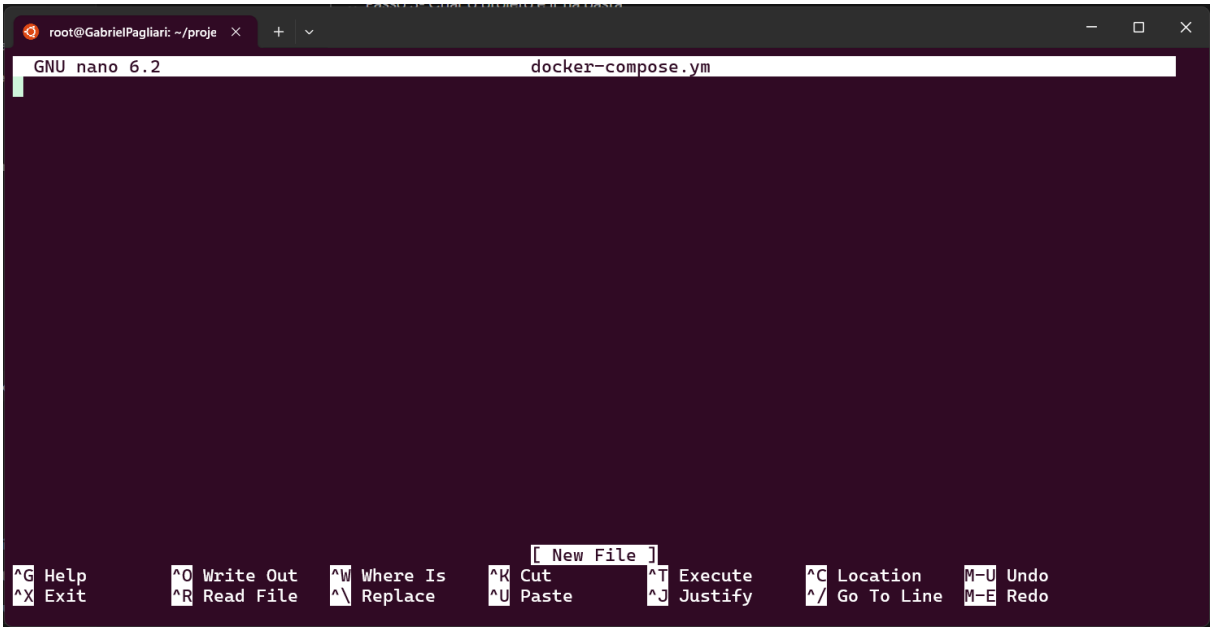


Passo 3- Criar o projeto e ir na pasta

mkdir projeto_final
cd projeto_final



Passo 4- Criar o arquivo docker-compose.yml:
nano docker-compose.ym



Passo 5- Configure dessa forma:

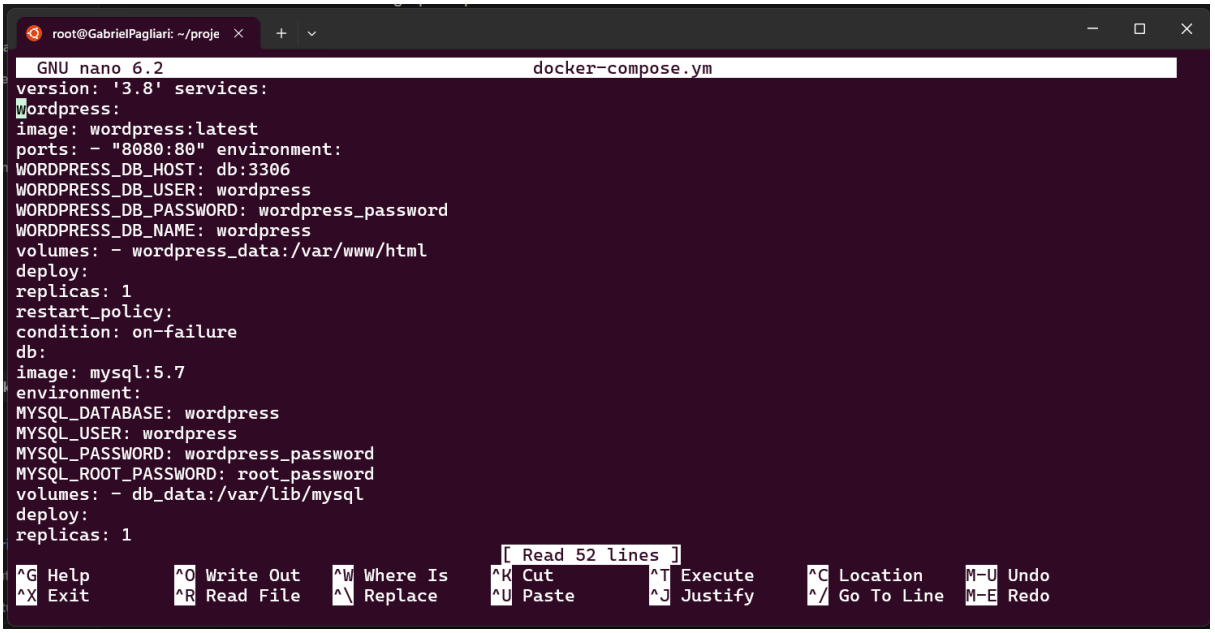
version: '3.8'
services:
wordpress:
image: wordpress:latest
ports:
- "8080:80"
environment:
WORDPRESS_DB_HOST: db:3306
WORDPRESS_DB_USER: wordpress
WORDPRESS_DB_PASSWORD: wordpress_password
WORDPRESS_DB_NAME: wordpress
volumes:
- wordpress_data:/var/www/html
deploy:
replicas: 1
restart_policy:
condition: on-failure

db:
image: mysql:5.7
environment:
MYSQL_DATABASE: wordpress
MYSQL_USER: wordpress
MYSQL_PASSWORD: wordpress_password

```
MYSQL_ROOT_PASSWORD: root_password
volumes:
- db_data:/var/lib/mysql
deploy:
replicas: 1
restart_policy:
condition: on-failure

redis:
image: redis:latest
ports:
- "6379:6379"
volumes:
- redis_data:/data
deploy:
replicas: 1
restart_policy:
condition: on-failure

prometheus:
image: prom/prometheus:latest
ports:
- "9090:9090"
volumes:
- ./prometheus.yml:/etc/prometheus/prometheus.yml
deploy:
replicas: 1
restart_policy:
condition: on-failure

grafana:
image: grafana/grafana:latest
ports:
- "3000:3000"
environment:
- GF_SECURITY_ADMIN_PASSWORD=admin
volumes:
- grafana_data:/var/lib/grafana
deploy:
replicas: 1
```

restart_policy:
condition: on-failure

volumes:
wordpress_data:
db_data:
redis_data:
grafana_data:



Passo 6- Criar e configurar o Prometheus:

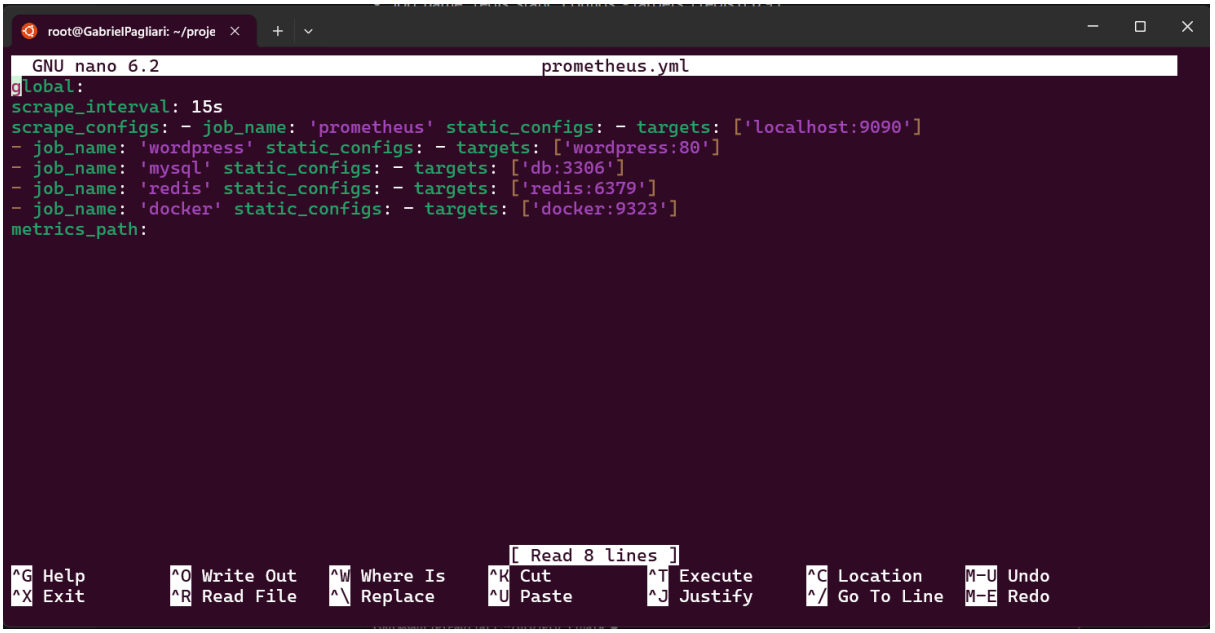
nano prometheus.yml

global:
scrape_interval: 15s

scrape_configs:

- job_name: 'prometheus'
  static_configs:
  - targets: ['localhost:9090']
- job_name: 'wordpress'
  static_configs:
  - targets: ['wordpress:80']
- job_name: 'mysql'
  static_configs:

    - targets: ['db:3306']
- job_name: 'redis'
  static_configs:
    - targets: ['redis:6379']
- job_name: 'docker'
  static_configs:
    - targets: ['docker:9323']

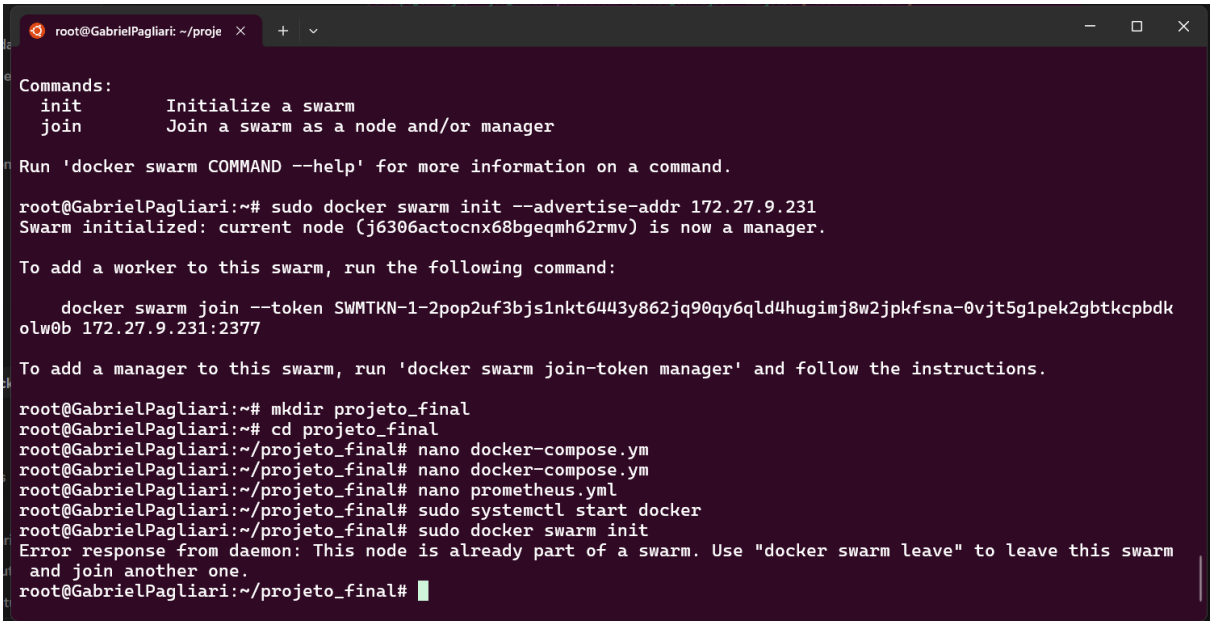


Passo 7- Iniciar o Docker

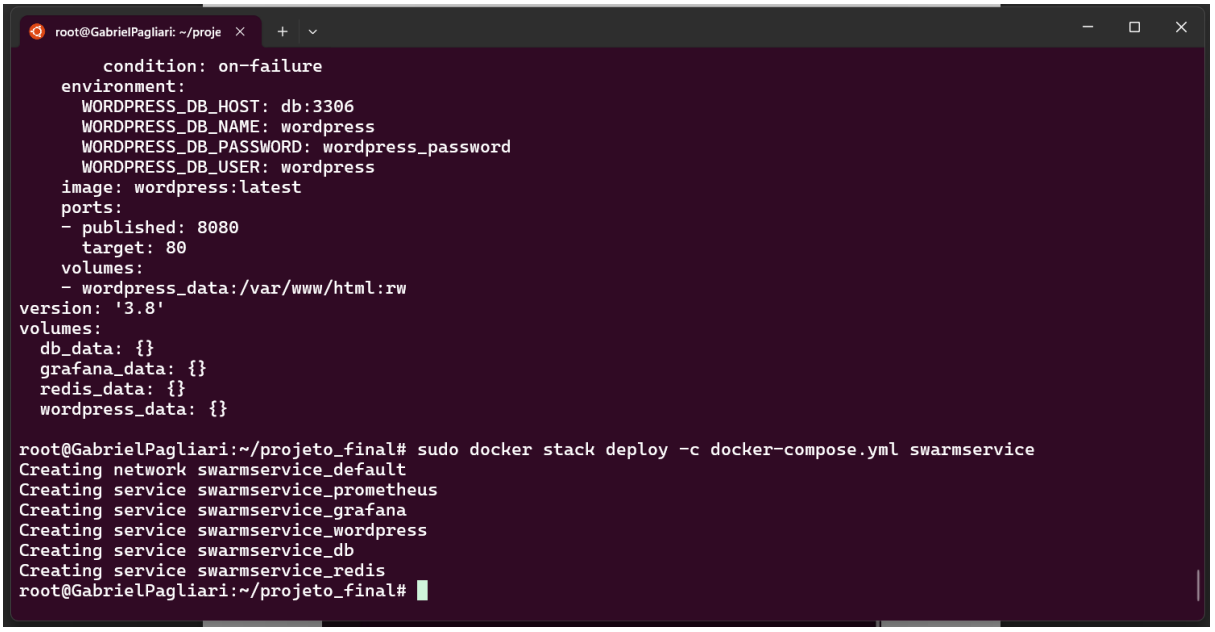
sudo systemctl start docker

Passo 8- Iniciar o Docker Swarm

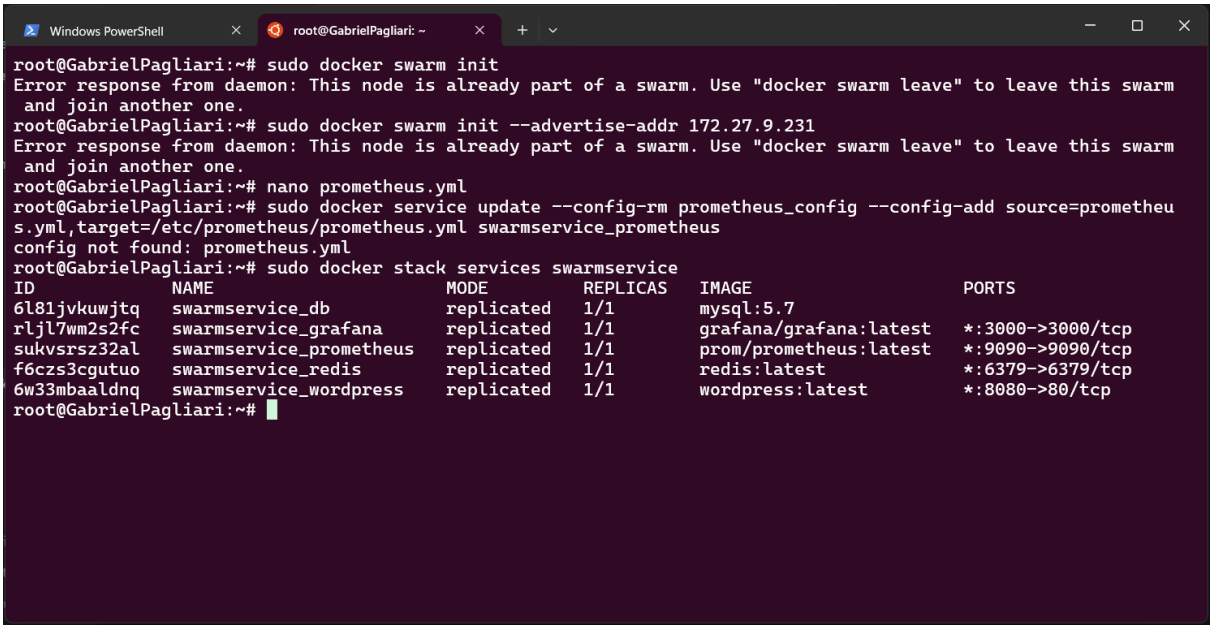
sudo docker swarm init



Passo 9- Colocar os serviços:
sudo docker stack deploy -c docker-compose.yml swarmservice



Passo 10- Verificando:
sudo docker stack services swarmservice


Windows PowerShell
root@GabrielPagliari: ~
root@GabrielPagliari:~# sudo docker swarm init
Error response from daemon: This node is already part of a swarm. Use "docker swarm leave" to leave this swarm
and join another one.
root@GabrielPagliari:~# sudo docker swarm init --advertise-addr 172.27.9.231
Error response from daemon: This node is already part of a swarm. Use "docker swarm leave" to leave this swarm
and join another one.
root@GabrielPagliari:~# nano prometheus.yml
root@GabrielPagliari:~# sudo docker service update --config-rm prometheus_config --config-add source=prometheu
s.yml,target=/etc/prometheus/prometheus.yml swarmservice_prometheus
config not found: prometheus.yml
root@GabrielPagliari:~# sudo docker stack services swarmservice
ID NAME MODE REPLICAS IMAGE PORTS
6l81jvkuwjtq swarmservice_db replicated 1/1 mysql:5.7
rljl7wm2s2fc swarmservice_grafana replicated 1/1 grafana/grafana:latest *:3000->3000/tcp
sukvsrsz32al swarmservice_prometheus replicated 1/1 prom/prometheus:latest *:9090->9090/tcp
f6czs3cgutuo swarmservice_redis replicated 1/1 redis:latest *:6379->6379/tcp
6w33mbaaldnq swarmservice_wordpress replicated 1/1 wordpress:latest *:8080->80/tcp
root@GabrielPagliari:~#