Docker-Compose Part 2

Trần Minh Tiến

21010611

Phần 1: Một số lệnh Docker Compose cơ bản

1. Kiểm tra phiên bản docker compose

docker compose version

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose version
Docker Compose version v2.33.1-desktop.1

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>
```

2. Chạy các container theo định nghĩa trong docker-compose.yml (foreground)

docker compose up

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose up
[+] Running 1/12
 - db [▒▒▒▒▒▒▒▒▒▒] Pulling                                                    6.0s
   - 548990e33276 Pulling fs layer                                          0.6s
   - 79f239a40e62 Pulling fs layer                                          0.6s
   - c11056354384 Pulling fs layer                                          0.6s
   - 769c3ac51f88 Pulling fs layer                                          0.6s
   - 49978e7ccddf Pulling fs layer                                          0.6s
   - cb8acbf2440c Pulling fs layer                                          0.6s
   - b2ead3e96e6b Pulling fs layer                                          0.6s
   - fae51f7de1fb Pulling fs layer                                          0.6s
   - daac2c594bdd Downloading    [==============================================>] ...  0.6s
   - cea172a6e83b Pulling fs layer                                          0.6s
 ▒ web Pulled                                                               3.7s
```

```
C:\Windows\System32\cmd.exe - docker compose up
Microsoft Windows [Version 10.0.19045.5608]
(c) Microsoft Corporation. All rights reserved.

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose up
[+] Running 12/12
 ▒ db Pulled                                                              104.2s
   ▒ 548990e33276 Pull complete                                            98.6s
   ▒ 79f239a40e62 Pull complete                                            73.7s
   ▒ c11056354384 Pull complete                                             3.8s
   ▒ 769c3ac51f88 Pull complete                                            70.8s
   ▒ 49978e7ccddf Pull complete                                            94.3s
   ▒ cb8acbf2440c Pull complete                                            16.1s
   ▒ b2ead3e96e6b Pull complete                                             7.0s
   ▒ fae51f7de1fb Pull complete                                            70.5s
   ▒ daac2c594bdd Pull complete                                             3.8s
   ▒ cea172a6e83b Pull complete                                            69.9s
 ▒ web Pulled                                                               3.7s
[+] Running 3/3
 ▒ Network week7_default  Created                                           0.2s
 ▒ Container week7-web-1  Created                                           9.2s
 ▒ Container week7-db-1   Created                                           9.2s
Attaching to db-1, web-1
db-1   | 2025-04-10 05:51:42+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.
web-1  | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
web-1  | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
web-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
web-1  | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
web-1  | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
web-1  | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
```

3. Chạy các container ở chế độ nền (backgrou)

docker compose up -d

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose up -d
[+] Running 2/2
 ✔ Container week7-web-1  Started                                                0.7s
 ✔ Container week7-db-1   Started                                                0.8s

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>
```

4. Liệt kê các container đang chạy bằng Docker Compose

docker compose ps

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose ps
NAME          IMAGE     COMMAND                  SERVICE   CREATED             STATUS          PORTS
week7-db-1    mysql     "docker-entrypoint.s…"   db        2 minutes ago       Up 15 seconds   0.0.0.0:3306->3306/tcp,
33060/tcp
week7-web-1   nginx     "/docker-entrypoint.…"   web       About a minute ago  Up 15 seconds   0.0.0.0:8080->80/tcp

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>
```

5. Dừng và xóa container, network, volume... được tạo ra

docker compose down

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose down
[+] Running 2/3
 ✔ Container week7-db-1   Removed                                                2.0s
 ✔ Container week7-web-1  Removed                                                1.1s
 - Network week7_default  Removing                                               0.9s

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose ps
NAME      IMAGE     COMMAND   SERVICE   CREATED   STATUS    PORTS
```

6. Khởi động lại các service đang chạy

docker compose restart

```
>[+] Restarting 2/2TranMinhTien-Architectural-Software\WEEK7>docker compose restart
 ✔ Container week7-db-1   Started                                                0.5s
 ✔ Container week7-web-1  Started                                                0.4s
```

7. Xem log các container (realtime, tương đương tail -f)

docker compose logs -f

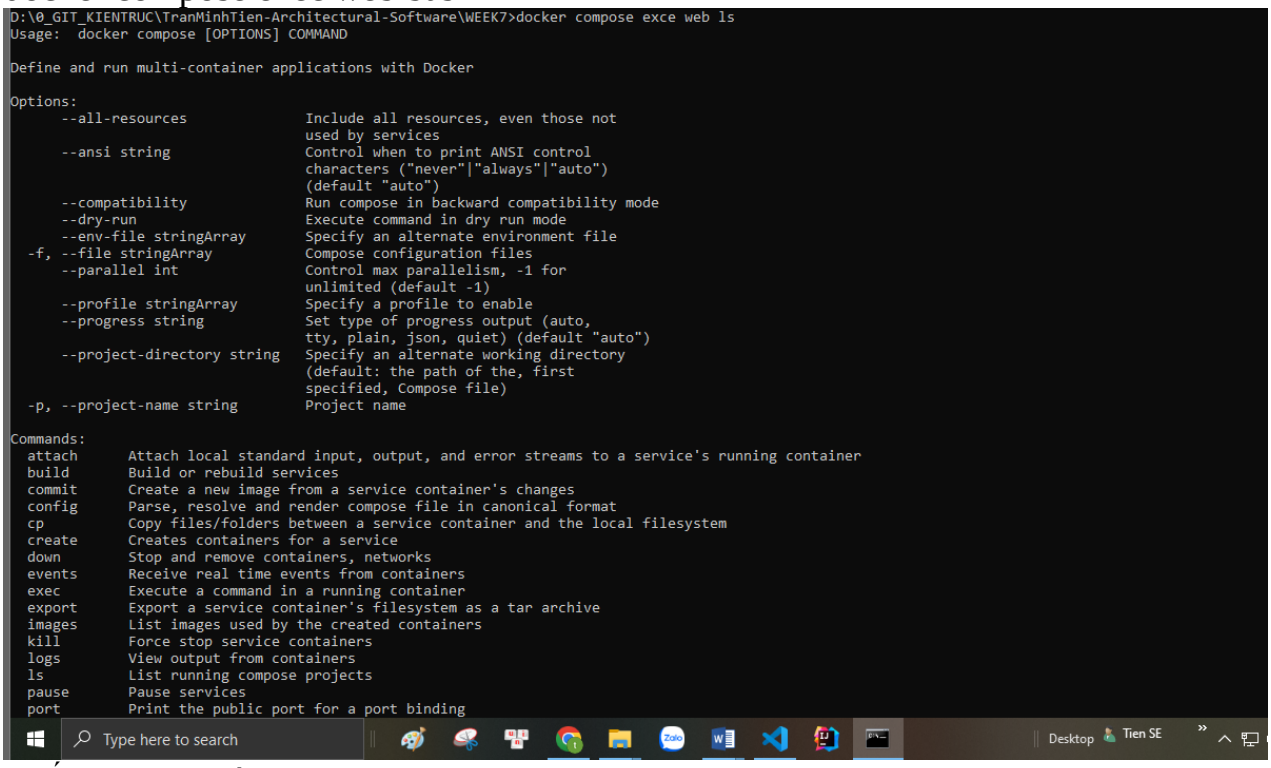
```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose logs -f
web-1  | /docker-entrypoint.sh: /docker-entrypoint.d/ is not empty, will attempt to perform configuration
db-1   | 2025-04-10 05:56:44+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.
web-1  | /docker-entrypoint.sh: Looking for shell scripts in /docker-entrypoint.d/
web-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/10-listen-on-ipv6-by-default.sh
web-1  | 10-listen-on-ipv6-by-default.sh: info: Getting the checksum of /etc/nginx/conf.d/default.conf
web-1  | 10-listen-on-ipv6-by-default.sh: info: Enabled listen on IPv6 in /etc/nginx/conf.d/default.conf
web-1  | /docker-entrypoint.sh: Sourcing /docker-entrypoint.d/15-local-resolvers.envsh
db-1   | 2025-04-10 05:56:45+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
db-1   | 2025-04-10 05:56:45+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 9.2.0-1.el9 started.
web-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/20-envsubst-on-templates.sh
web-1  | /docker-entrypoint.sh: Launching /docker-entrypoint.d/30-tune-worker-processes.sh
web-1  | /docker-entrypoint.sh: Configuration complete; ready for start up
web-1  | 2025/04/10 05:56:44 [notice] 1#1: using the "epoll" event method
web-1  | 2025/04/10 05:56:44 [notice] 1#1: nginx/1.27.4
web-1  | 2025/04/10 05:56:44 [notice] 1#1: built by gcc 12.2.0 (Debian 12.2.0-14)
web-1  | 2025/04/10 05:56:44 [notice] 1#1: OS: Linux 5.15.167.4-microsoft-standard-WSL2
web-1  | 2025/04/10 05:56:44 [notice] 1#1: getrlimit(RLIMIT_NOFILE): 1048576:1048576
web-1  | 2025/04/10 05:56:44 [notice] 1#1: start worker processes
web-1  | 2025/04/10 05:56:44 [notice] 1#1: start worker process 29
web-1  | 2025/04/10 05:56:44 [notice] 1#1: start worker process 30
web-1  | 2025/04/10 05:56:44 [notice] 1#1: start worker process 31
web-1  | 2025/04/10 05:56:44 [notice] 1#1: start worker process 32
```

8. Build lại các image

docker compose build

9. Thực thi lệnh trong container đang chạy (ví dụ mở bash trong service "web")

docker compose exec web bash



- Một số command thông dụng:

· bash hoặc sh: Mở một terminal shell trong container

· ls: Liệt kê các file và thư mục trong thư mục hiện tại của container

· pwd: Hiển thị đường dẫn thư mục hiện tại trong container

· cat <filename>: Đọc và hiển thị nội dung của một file trong container

· top: Hiển thị danh sách các tiến trình đang chạy trong container

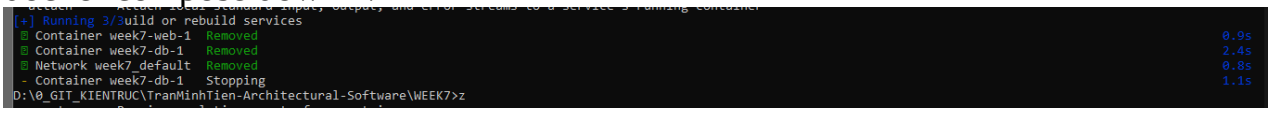
· ps: Hiển thị danh sách các tiến trình trong container

· df -h: Hiển thị thông tin về dung lượng đĩa trong container

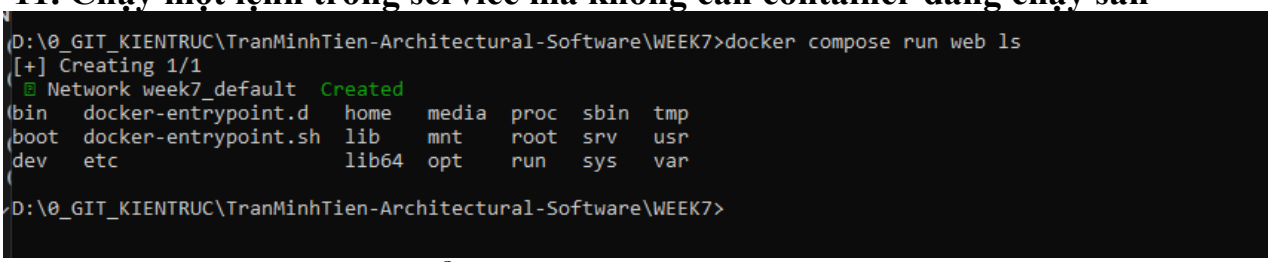
· curl: Gửi yêu cầu HTTP từ trong container

10. Dừng và xóa container, đồng thời xóa volume

docker compose down -v



11. Chạy một lệnh trong service mà không cần container đang chạy sẵn



12. Dừng một service cụ thể

```
E:\Worksapce\Project\Kientruc\Lab\week7>docker compose ps
NAME          IMAGE     COMMAND                  SERVICE   CREATED          STATUS          PORTS
week7-db-1    mysql     "docker-entrypoint.s…"   db        51 seconds ago   Up 19 seconds   0.0.0.0:3306->3306/tcp, 3306
0/tcp
week7-web-1   nginx     "/docker-entrypoint.…"   web       51 seconds ago   Up 19 seconds   0.0.0.0:8080->80/tcp

E:\Worksapce\Project\Kientruc\Lab\week7>docker compose stop web
[+] Stopping 1/1
 ✔Container week7-web-1  Stopped                                                                                0.7s

E:\Worksapce\Project\Kientruc\Lab\week7>
```

13. Xóa container đã dừng của một service

```
E:\Worksapce\Project\Kientruc\Lab\week7>docker compose stop web
[+] Stopping 1/1
 ✔Container week7-web-1  Stopped

? Going to remove week7-web-1 Yes
[+] Removing 1/1e week7-web-1 (y/N) y
 ✔Container week7-web-1  Removed

E:\Worksapce\Project\Kientruc\Lab\week7>
```

14. Hiển thị file docker-compose.yml sau khi hợp nhất và xử lý biến

docker compose config

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose config
name: week7
services:
  db:
    environment:
      MYSQL_DATABASE: db_test
      MYSQL_ROOT_PASSWORD: root
    image: mysql
    networks:
      default: null
    ports:
      - mode: ingress
        target: 3306
        published: "3306"
        protocol: tcp
  web:
    image: nginx
    networks:
      default: null
    ports:
      - mode: ingress
        target: 80
        published: "8080"
        protocol: tcp
networks:
  default:
    name: week7_default

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>
```

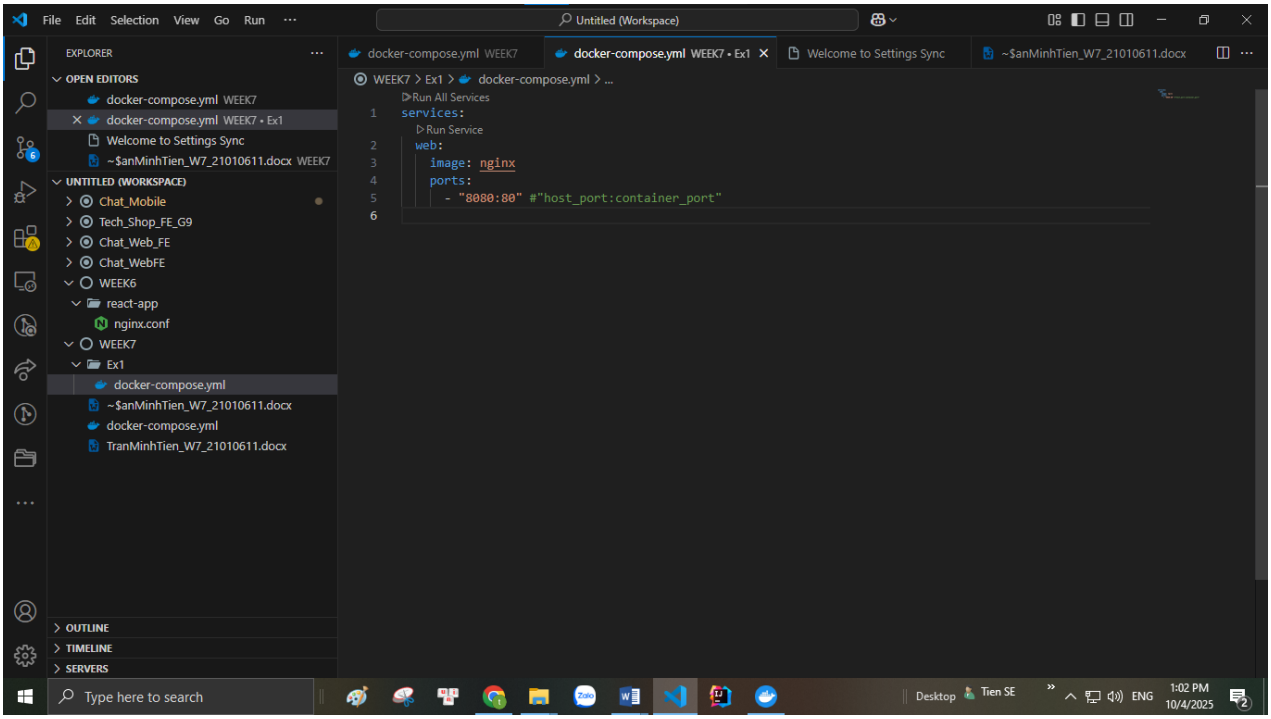
15. Build lại image và chạy các container ở chế độ nền

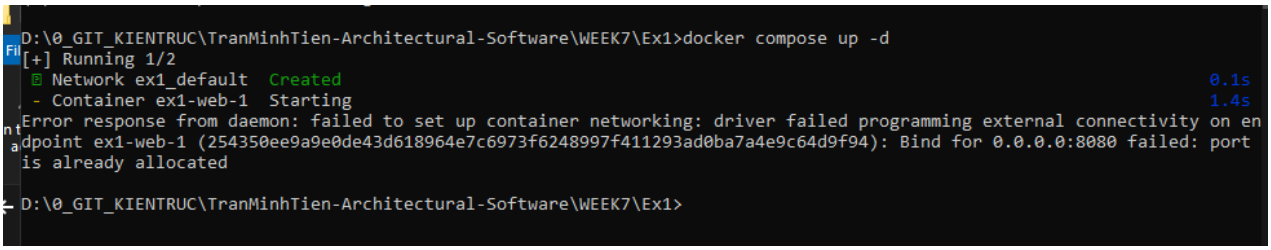
docker compose up -d --build

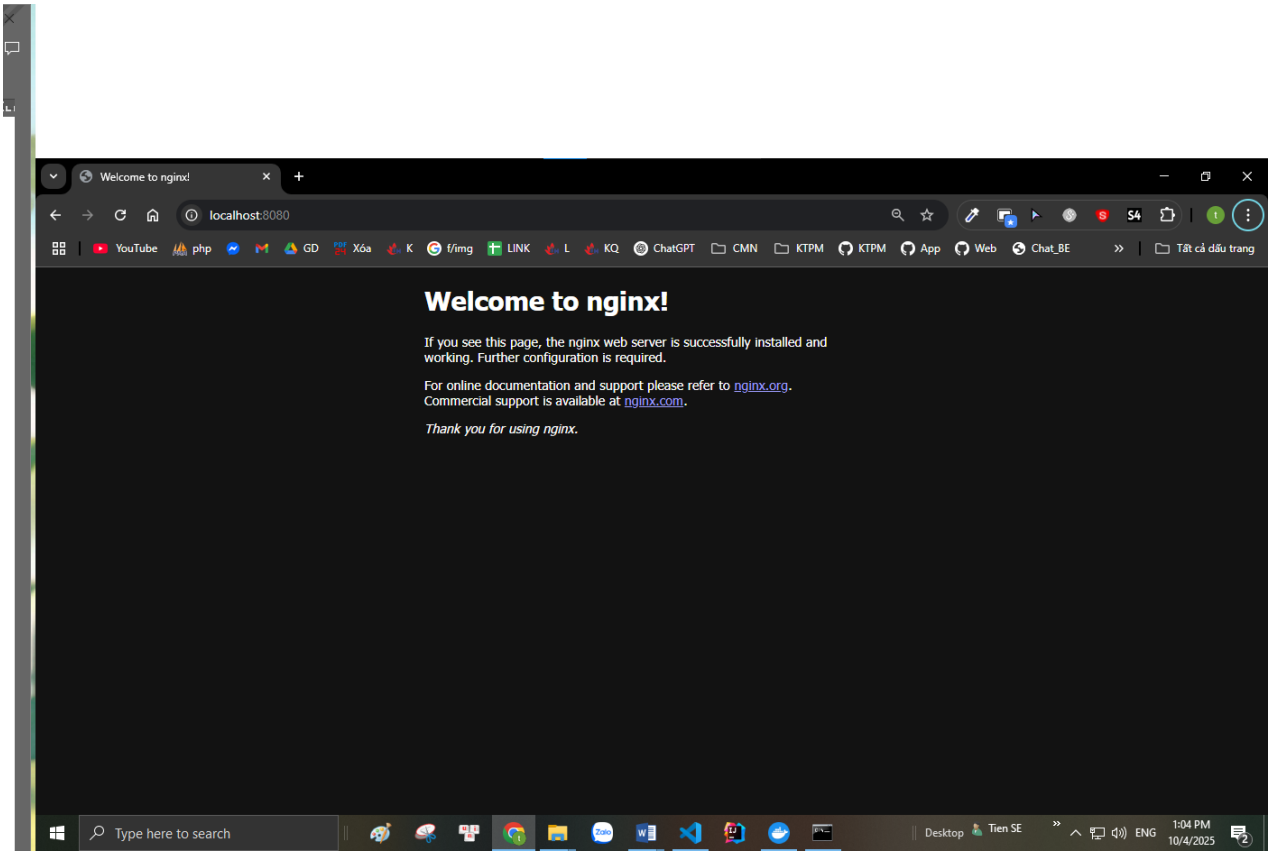
```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>docker compose up -d --build
time="2025-04-10T13:02:18+07:00" level=warning msg="Found orphan containers ([week7-web-run-cbe1986cdfdd]) for this proj
ect. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans fla
g to clean it up."
[+] Running 2/2
 ⠿ Container week7-db-1   Running                                                                                0.0s
 ⠿ Container week7-web-1  Running                                                                                0.0s

D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7>
```

Phần 2: Docker Compose file

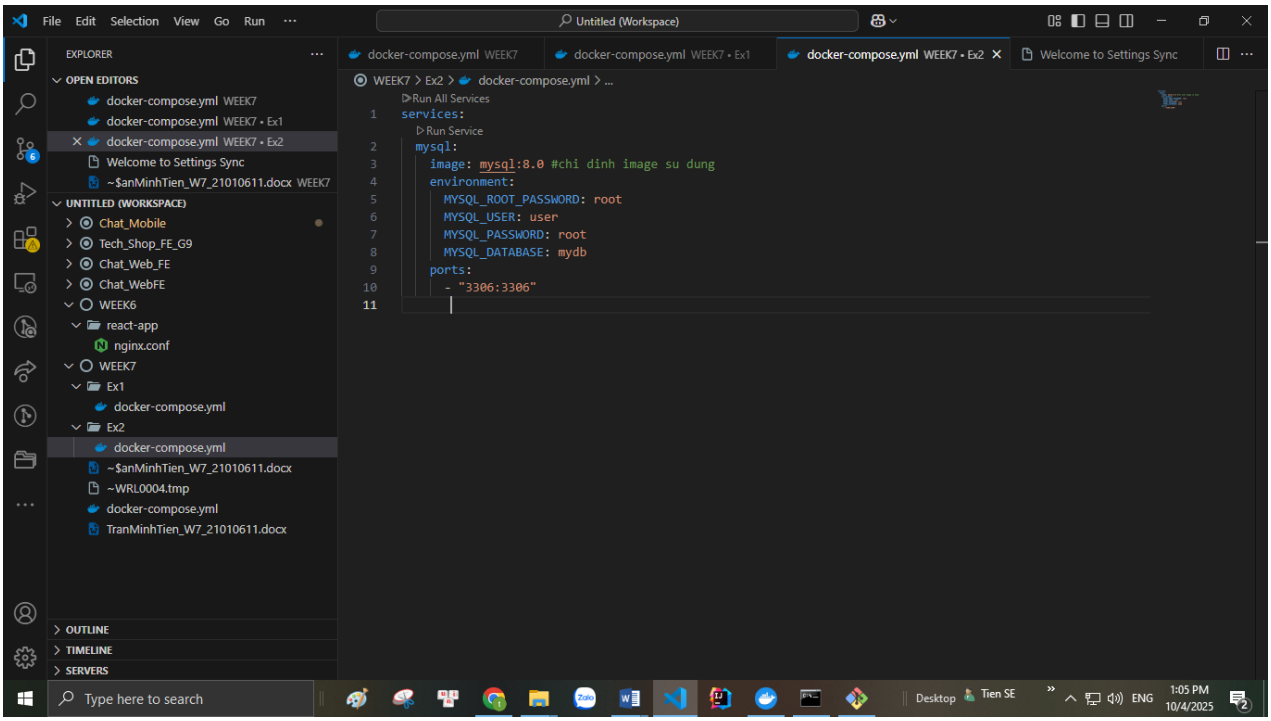
Bài 1: Chạy một container đơn giản với Docker Compose







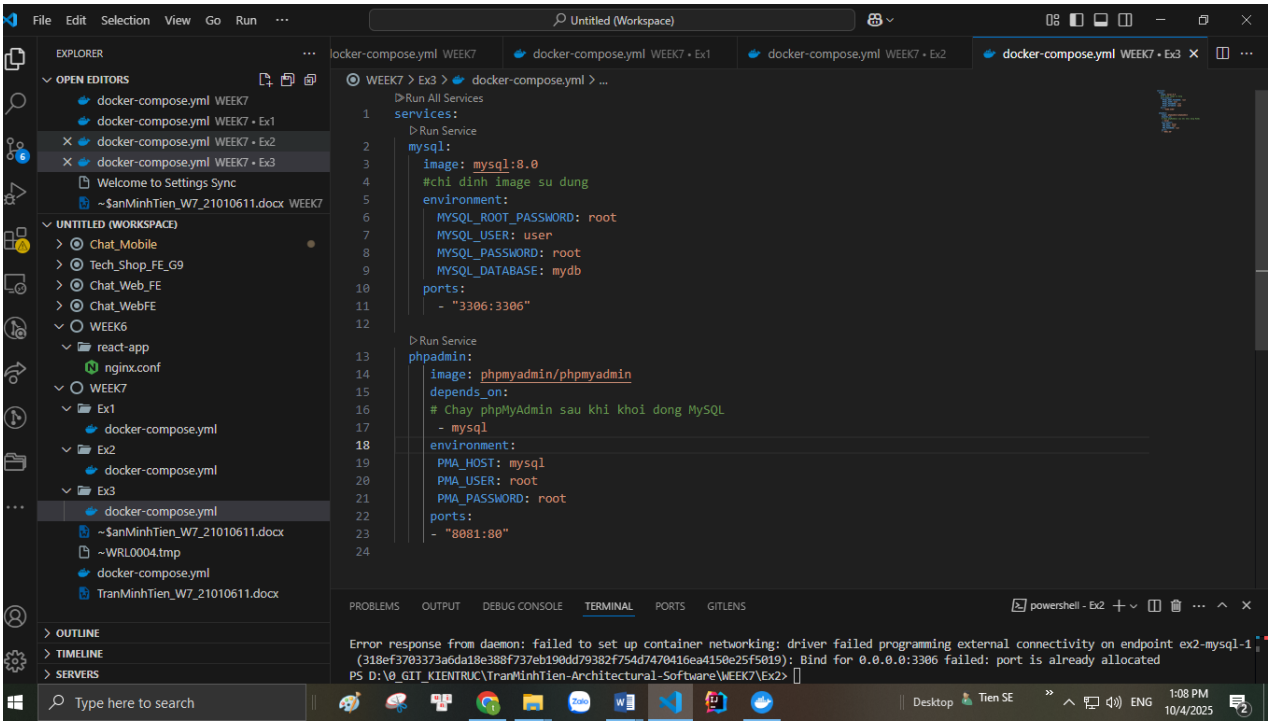
Bài 2: Chạy MySQL với Docker Compose

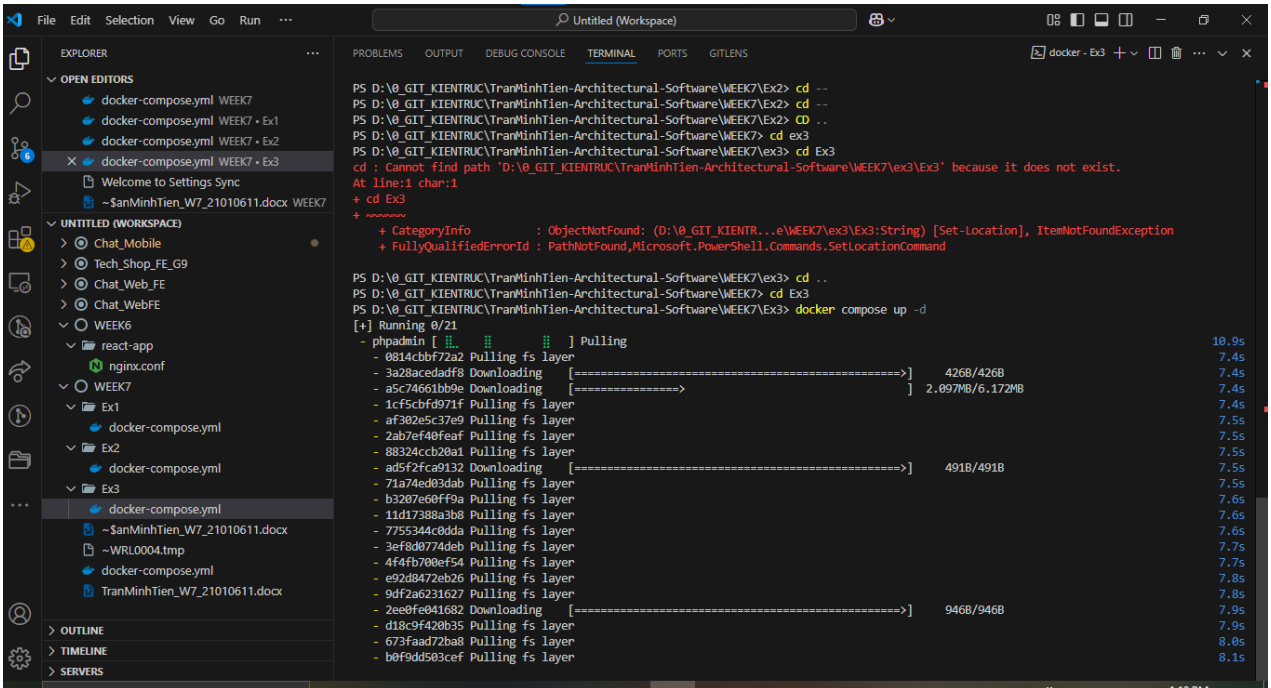


```
PS D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex2> docker compose up -d
[+] Running 2/11
 - mysql [  ⣿⣿    ⣿] Pulling                                                                                    8.5s
   - 28e01aa53f13 Pulling fs layer                                                                            4.6s
   - 753b8441f7e6 Pulling fs layer                                                                            4.6s
   ✓ b1339a14fa1a Download complete                                                                           0.4s
   - e5fa3211d7a7 Pulling fs layer                                                                            4.7s
   - c106a4902288 Downloading    [==================================================>]     315B/315B         4.7s
   - de67a2f637e5 Pulling fs layer                                                                            4.7s
   - be386ff914e3 Pulling fs layer                                                                            4.7s
   - 93272c957f26 Pulling fs layer                                                                            4.8s
   - 036f4325df2d Pulling fs layer                                                                            4.8s
   ✓ 1d34979e7120 Download complete                                                                           3.6s
```

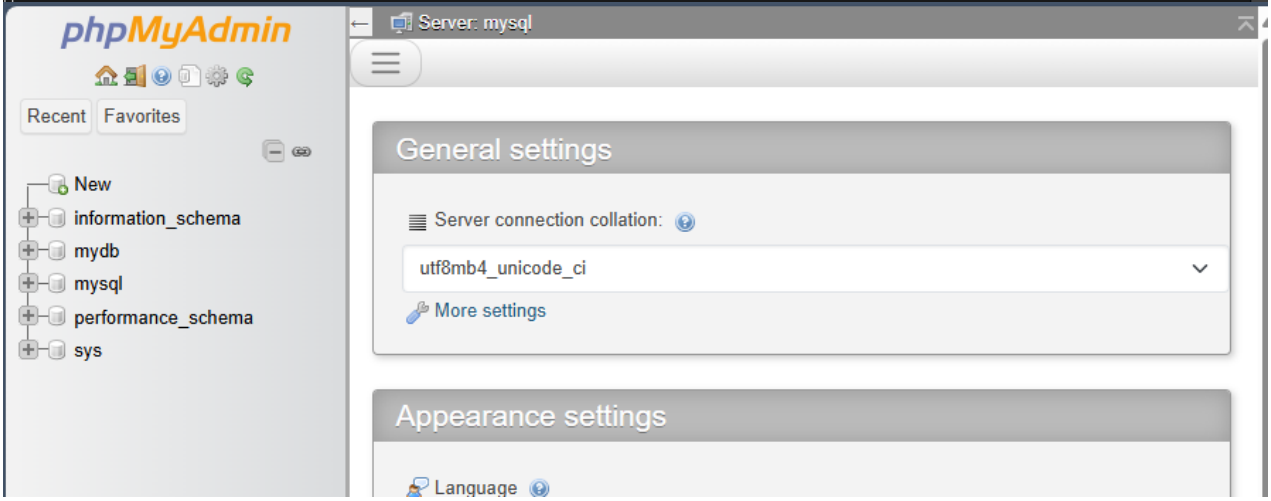
```
PS D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex2> docker compose up -d
[+] Running 11/11
 ✓ mysql Pulled                                                                                              80.3s
   ✓ 28e01aa53f13 Pull complete                                                                              13.3s
   ✓ 753b8441f7e6 Pull complete                                                                              14.3s
   ✓ b1339a14fa1a Pull complete                                                                              17.7s
   ✓ e5fa3211d7a7 Pull complete                                                                              13.2s
   ✓ c106a4902288 Pull complete                                                                               8.2s
   ✓ de67a2f637e5 Pull complete                                                                              76.3s
   ✓ be386ff914e3 Pull complete                                                                              17.7s
   ✓ 93272c957f26 Pull complete                                                                              54.8s
   ✓ 036f4325df2d Pull complete                                                                              75.4s
   ✓ 1d34979e7120 Pull complete                                                                               3.6s
[+] Running 1/2
 ✓ Network ex2_default    Created                                                                             0.3s
 - Container ex2-mysql-1  Starting                                                                           15.0s
Error response from daemon: failed to set up container networking: driver failed programming external connectivity on endpoint ex2-mysql-1
 (318ef3703373a6da18e388f737eb190dd79382f754d7470416ea4150e25f5019): Bind for 0.0.0.0:3306 failed: port is already allocated
PS D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex2>
```

Bài 3: Kết nối MySQL với PHPMyAdmin





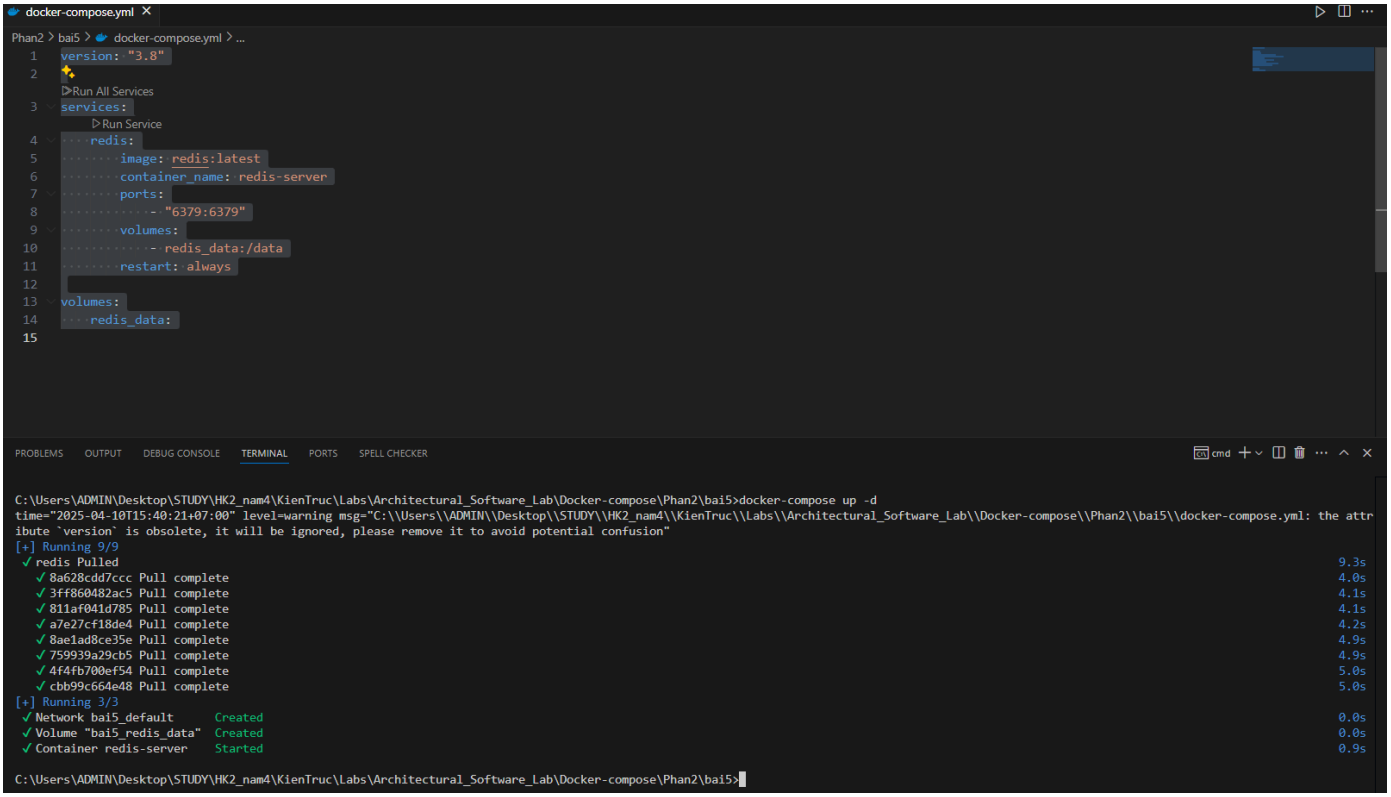

```
PS D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex3> docker compose up -d
[+] Running 21/21
 ✓ phpadmin Pulled                                                    103.6s
   ✓ 0814cbbf72a2 Pull complete                                        81.8s
   ✓ 3a28acedadf8 Pull complete                                        81.8s
   ✓ a5c74661bb9e Pull complete                                        87.0s
   ✓ 1cf5cbfd971f Pull complete                                        87.3s
   ✓ af302e5c37e9 Pull complete                                        59.9s
   ✓ 2ab7ef40feaf Pull complete                                        34.8s
   ✓ 88324ccb20a1 Pull complete                                        82.5s
   ✓ ad5f2fca9132 Pull complete                                        82.7s
   ✓ 71a74ed03dab Pull complete                                        60.1s
   ✓ b3207e60ff9a Pull complete                                        34.8s
   ✓ 11d17388a3b8 Pull complete                                        34.8s
   ✓ 7755344c0dda Pull complete                                        34.8s
   ✓ 3ef8d0774deb Pull complete                                        79.0s
   ✓ 4f4fb700ef54 Pull complete                                        34.9s
   ✓ e92d8472eb26 Pull complete                                        99.8s
   ✓ 9df2a6231627 Pull complete                                        84.5s
   ✓ 2ee0fe041682 Pull complete                                        32.8s
   ✓ d18c9f420b35 Pull complete                                        35.0s
   ✓ 673faad72ba8 Pull complete                                        34.9s
   ✓ b0f9dd503cef Pull complete                                       100.0s
[+] Running 1/2
 ✓ Network ex3_default     Created                                     0.2s
 - Container ex3-mysql-1   Creating                                    4.7s
```



- Tạo file docker-compose.yml

```
version: "3.8"

services:
    redis:
        image: redis:lates
        container_nam : redis-serve
        ports:          r
            - "6379:637
        volumes:
            - redis_data:/dat
        restart: always

volumes:
    redis_dat :
    a
```

- Chạy redis



- Kiểm tra

```
ibute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 9/9
 ✓ redis Pulled
   ✓ 8a628cdd7ccc Pull complete
   ✓ 3ff860482ac5 Pull complete
   ✓ 811af041d785 Pull complete
   ✓ a7e27cf18de4 Pull complete
   ✓ 8ae1ad8ce35e Pull complete
   ✓ 759939a29cb5 Pull complete
   ✓ 4f4fb700ef54 Pull complete
   ✓ cbb99c664e48 Pull complete
[+] Running 3/3
 ✓ Network bai5_default      Created
 ✓ Volume "bai5_redis_data"  Created
 ✓ Container redis-server    Started

C:\Users\ADMIN\Desktop\STUDY\HK2_nam4\KienTruc\Labs\Architectural_Software_Lab\Docker-compose\Phan2\bai5>docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS          PORTS                    NAMES
0efd268de412   redis:latest   "docker-entrypoint.s…"   36 seconds ago   Up 35 seconds   0.0.0.0:6379->6379/tcp   redis-server

C:\Users\ADMIN\Desktop\STUDY\HK2_nam4\KienTruc\Labs\Architectural_Software_Lab\Docker-compose\Phan2\bai5>docker exec -it redis-server redis-cli
127.0.0.1:6379>
```

Bài 6: Chạy WordPress với MySQL

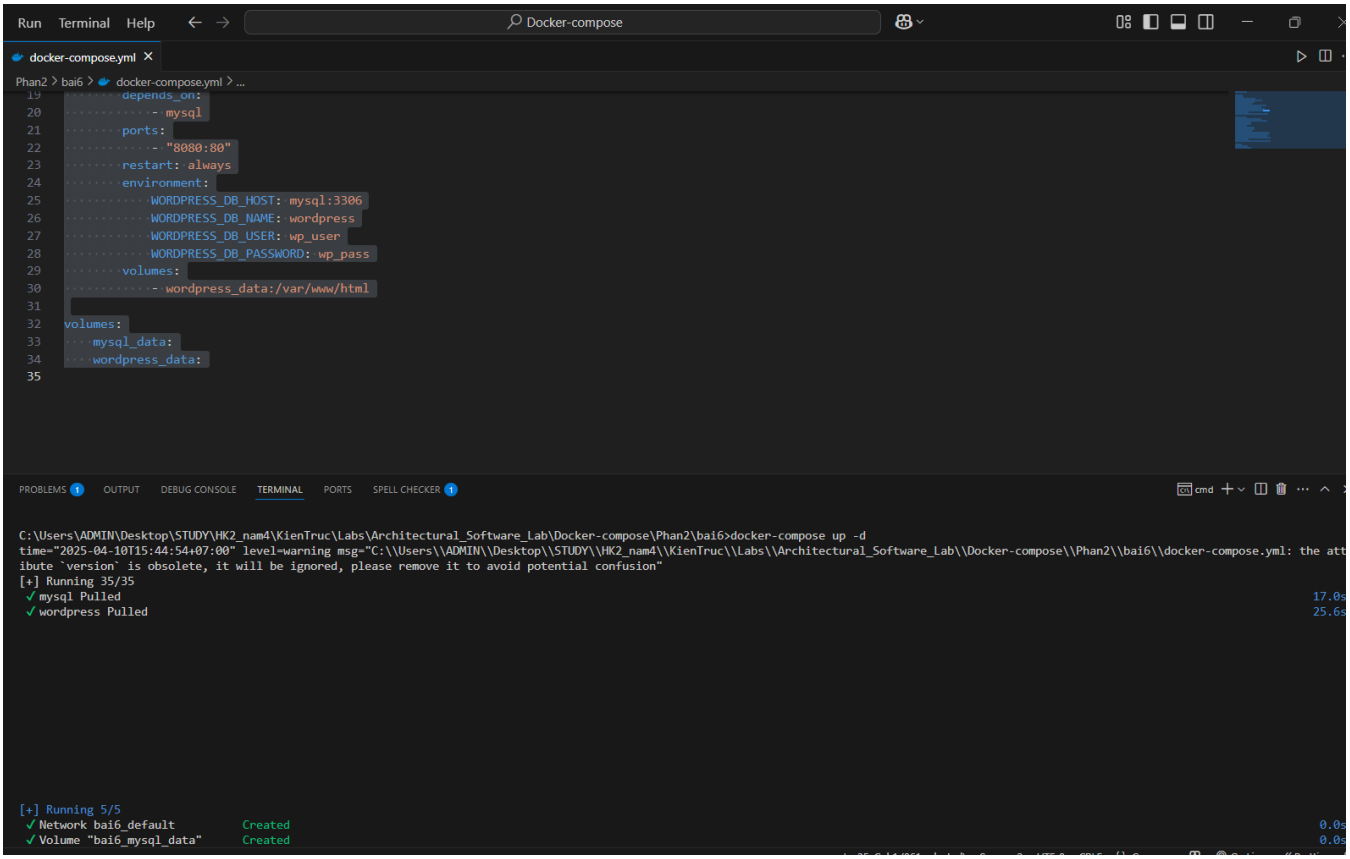
Yêu cầu:

Chạy WordPress với MySQL bằng Docker Compose.

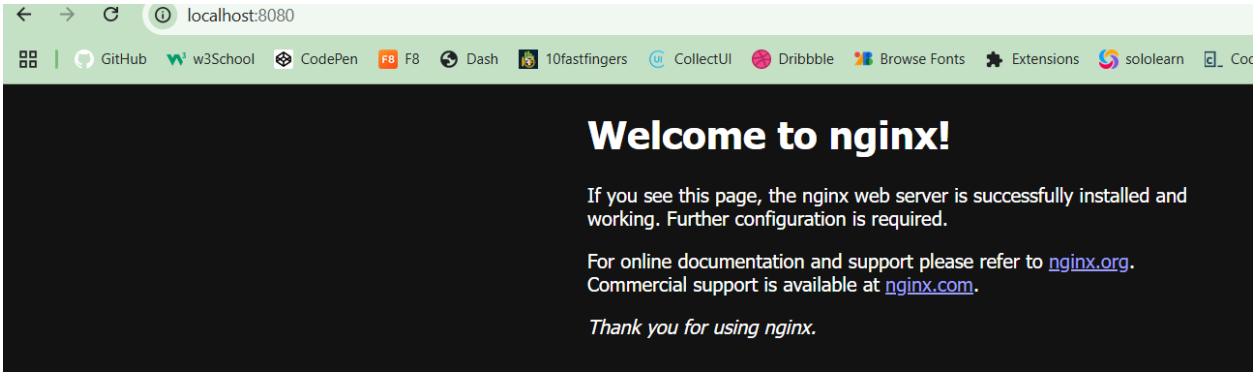
Tạo docker-compose.yml

```
version: "3.8"

services:
    mysql:
        image: mysql:5.
        container_nam : mysql-db
        restart: always
        environmen :
        t   MYSQL_DATABAS : wordpres
            MYSQL_USE : wp_user
            MYSQL_PASSWOR : wp_pass
            MYSQL_ROOT_PASSWOR : rootpass
        volumes:
            - mysql_data:/var/lib/mysq
              l
    wordpres :
    s   image: wordpress:lates
        container_nam : wordpress-sit
        depends_o :     e
        n   - mysql
        ports:
            - "8080:8
        restart: always
        environmen :
        t   WORDPRESS_DB_HOS : mysql:330
            WORDPRESS_DB_NAM : wordpres
            WORDPRESS_DB_USE : wp_user
            WORDPRESS_DB_PASSWOR : wp_pass
        volumes:
            - wordpress_data:/var/www/htm
              l
volumes:
    mysql_dat :
    wordpress_dat :
    a
```

Chạy WordPress và MySQL



Kiểm tra



Bài 7: Chạy MongoDB với Docker Compose

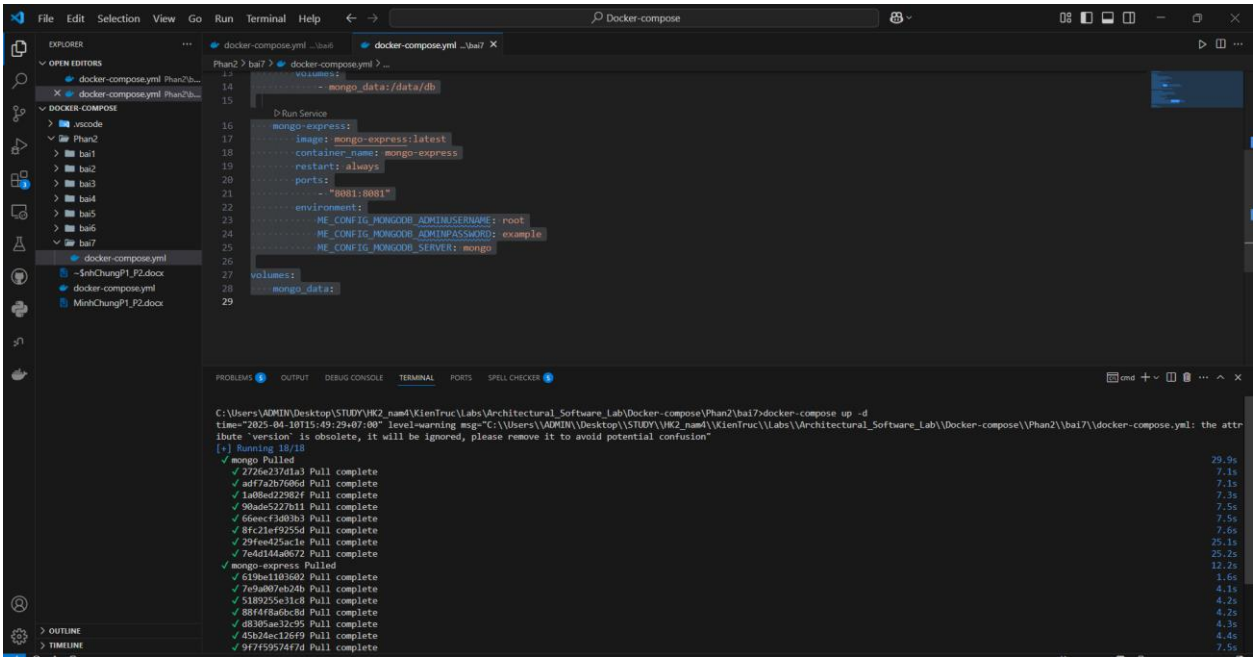
Yêu cầu:

Chạy MongoDB và Mongo Express để quản lý.

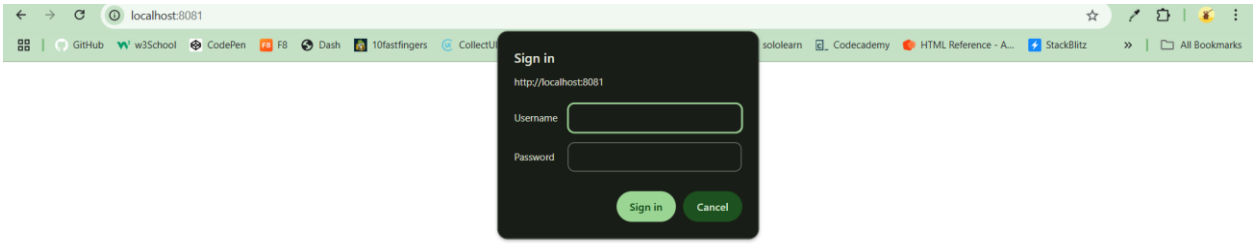
- Tạo docker-compose.yml

```yaml
version: "3.8"

services:
    mongo:
        image: mongo:lates
        container_nam : mongo-db
        restart: always
        ports:
            - "27017:2701
        environmen :
        t   MONGO_INITDB_ROOT_USERNAM : root
            MONGO_INITDB_ROOT_PASSWOR : example
        volumes:
            - mongo_data:/data/d
              b
    mongo-expres :
    s   image: mongo-express:lates
        container_nam : mongo-expres
        restart: always s
        ports:
            - "8081:808
        environmen :
        t   ME_CONFIG_MONGODB_ADMINUSERNAM : root
            ME_CONFIG_MONGODB_ADMINPASSWOR : example
            ME_CONFIG_MONGODB_SERVE : mongo
            R
volumes:
    mongo_dat :
    a
```

- Chạy MongoDB và MongoExpress


File Edit Selection View Go Run Terminal Help
Docker-compose
EXPLORER
OPEN EDITORS
docker-compose.yml Phan2\b...
docker-compose.yml Phan2\b...
DOCKER-COMPOSE
.vscode
Phan2
bai1
bai2
bai3
bai4
bai5
bai6
bai7
docker-compose.yml
~$nhChungP1_P2.docx
docker-compose.yml
MinhChungP1_P2.docx
docker-compose.yml ...\bai6
docker-compose.yml ...\bai7
Phan2 > bai7 > docker-compose.yml > ...
- mongo_data:/data/db
Run Service
mongo-express:
image: mongo-express:latest
container_name: mongo-express
restart: always
ports:
- "8081:8081"
environment:
ME_CONFIG_MONGODB_ADMINUSERNAME: root
ME_CONFIG_MONGODB_ADMINPASSWORD: example
ME_CONFIG_MONGODB_SERVER: mongo
volumes:
mongo_data:
PROBLEMS 5
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
SPELL CHECKER 5
cmd
C:\Users\ADMIN\Desktop\STUDY\HK2_nam4\KienTruc\Labs\Architectural_Software_Lab\Docker-compose\Phan2\bai7>docker-compose up -d
time="2025-04-10T15:49:29+07:00" level=warning msg="C:\\Users\\ADMIN\\Desktop\\STUDY\\HK2_nam4\\KienTruc\\Labs\\Architectural_Software_Lab\\Docker-compose\\Phan2\\bai7\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 10/10
✓ mongo Pulled 29.9s
✓ 2726e237d1a3 Pull complete 7.1s
✓ adf7a2b7606d Pull complete 7.1s
✓ 1a08ed22982f Pull complete 7.3s
✓ 90ade5227b11 Pull complete 7.5s
✓ 66eecf3d03b3 Pull complete 7.5s
✓ 8fc21ef9255d Pull complete 7.6s
✓ 29fee425ac1e Pull complete 25.1s
✓ 7e4d144a0672 Pull complete 25.2s
✓ mongo-express Pulled 12.2s
✓ 619be1103602 Pull complete 1.6s
✓ 7e9a807eb24b Pull complete 4.1s
✓ 5189255e31c8 Pull complete 4.2s
✓ 88f4f8a6bc8d Pull complete 4.2s
✓ d8305ae32c95 Pull complete 4.3s
✓ 45b24ec126f9 Pull complete 4.4s
✓ 9f7f59574f7d Pull complete 7.5s
OUTLINE
TIMELINE

- Kiểm tra


localhost:8081
GitHub
w3School
CodePen
F8
Dash
10fastfingers
sololearn
Codecademy
HTML Reference - A...
StackBlitz
All Bookmarks
Sign in
http://localhost:8081
Username
Password
Sign in
Cancel

Bài 8: Kết nối nhiều dịch vụ với Docker Compose

Yêu cầu:

Chạy Node.js kết nối với MySQL.

B1:Tạo project nodejs với docker-compose.yml

package.json

```
{
    "name": "bai8",
    "versio  : "1.0.0",
    "main": "index.j  ,
    "script s": {
    s"  "test": "echo \"Error: no test specifie \" && exit   ,
        "start": "node iddex.j                    1"
    },          s"
    "keyword  : [],
    "author": "",
    "licens  : "ISC",
    "descriptio  : "",
    "dependencie  : {
    s"  "expres  : "^5.1.0"
    }   s"
}
```

index.js

```
const mysql = require("mysql2");

const connectio  = mysql.createConnectio ({
    host: "mysql", // tênnservice trong docker-compos
    user: "root",  e
    password: "rootpas  ,
    database: "testdb",
});

connectio .connect((err) => {
n   if (err) {
        console.error("❌ Kết nối thất bạ   , err);
        return;       i:"
    }
    console.log("✅ Kết nối MySQL thành côn   );
                g!"
    connectio .query("SELECT NOW() AS no  , (err, results) => {
    n   if (err) throw"err;
        console.log("🕒 Giờ hệ thốn   , results[0].now);
        connectio .end();
    }); n
});
```

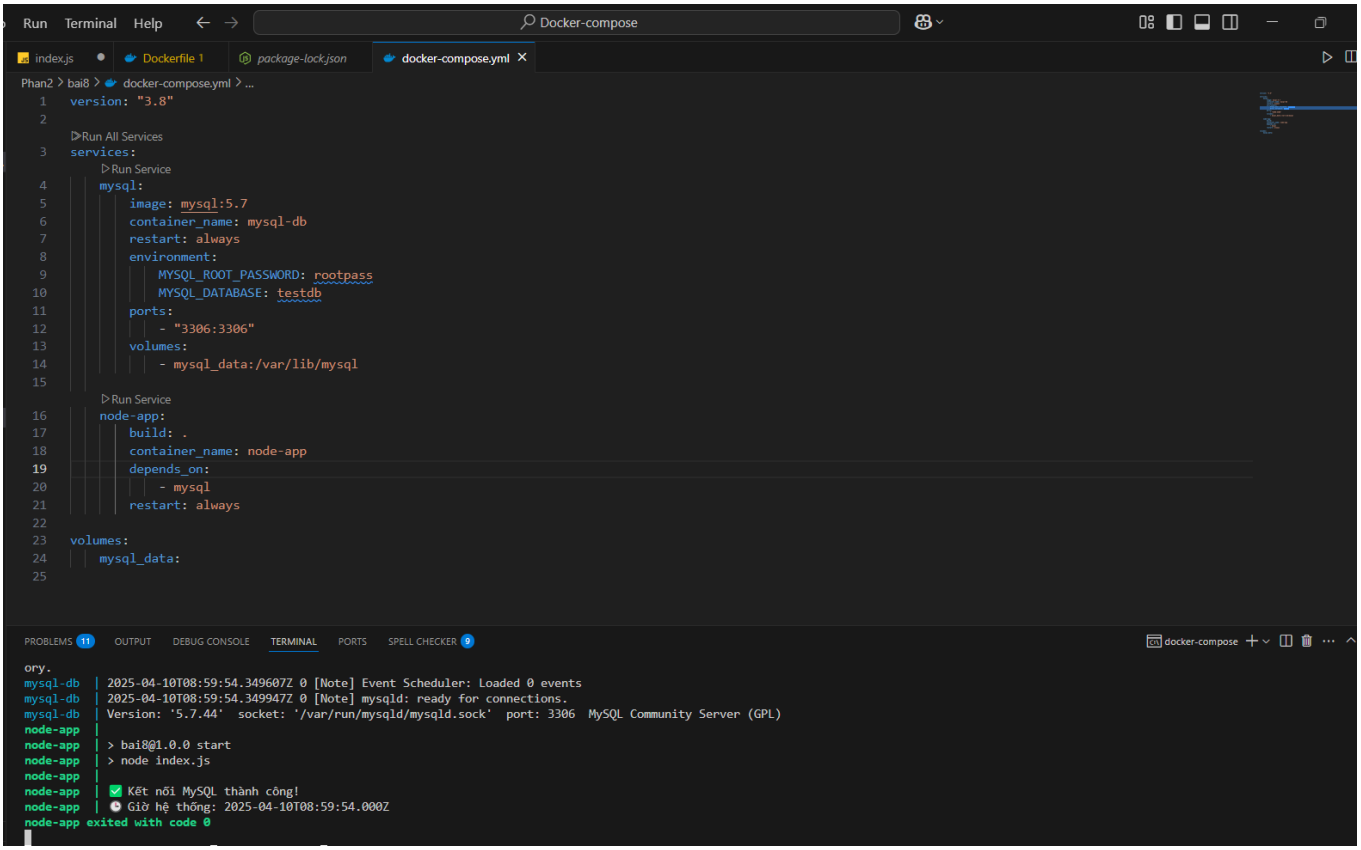
Dockerfile

```
FROM node:18

WORKDIR /app

COPY package*.jso  ./
RUN npm install

COPY . .

CMD ["npm", "start"]
```

docker-compose.yml

```
version: '3.8'

services:
  mysql:
    image: mysql:5.
    container_nam : mysql-db
    restart: always
    environmen :
    t MYSQL_ROOT_PASSWOR : rootpass
      MYSQL_DATABAS : testdb
    ports:
      - "3306:330
    volumes:
      - mysql_data:/var/lib/mysq
        l
  node-app:
    build: .
    container_nam : node-app
    depends_o :
    n - mysql
    restart: always

volumes:
  mysql_dat :
  a
```

- Chạy



Bài 9: Chạy ứng dụng Python Flask với Docker Compose

Yêu cầu:

Chạy ứng dụng Flask đơn giản với Docker Compose.

- Tạo ứng dụng Flask cơ bản

app.py

```
from flask import Flask
app = Flask(__name__)

@app.route('/')
def hello():
    return "✅ Xin chào từ Flask trong Docker Compos
           e!"
if __name__ == '__main_  :
    app.run(host='0.0.0.  , port=5000)
                  0'
```

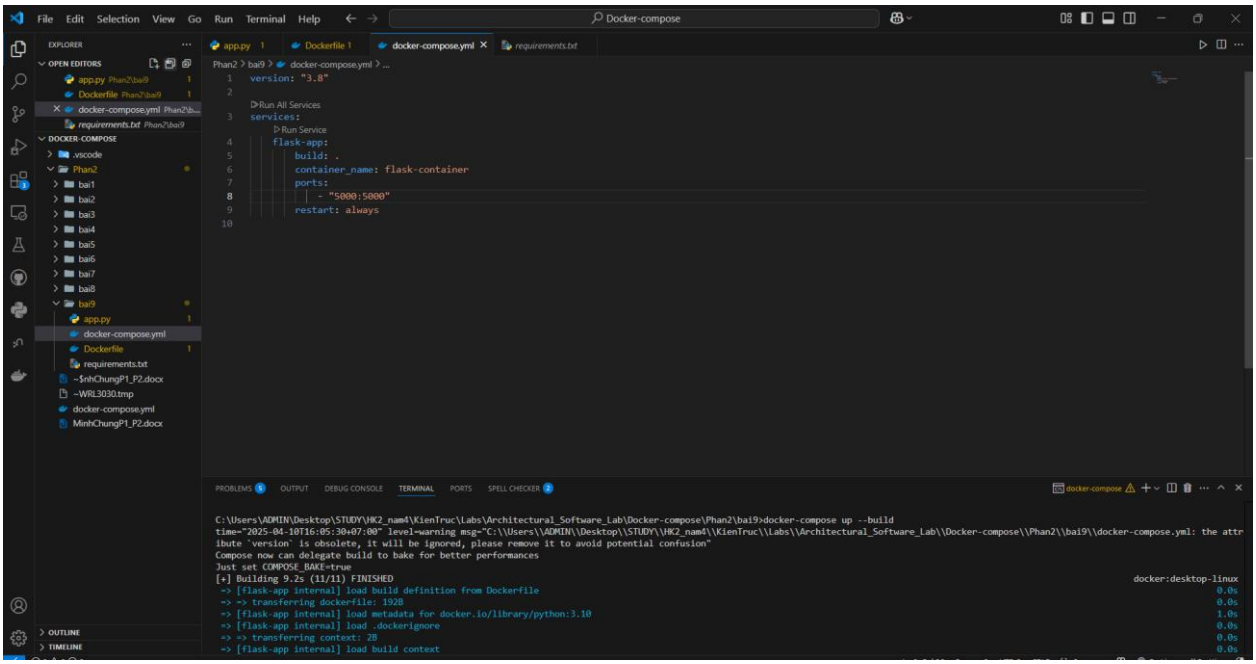
Dockerfile

```
FROM python:3.10

WORKDIR /app

COPY requirements.tx  .
RUN pip install --no-cache-di  -r requirements.tx
                r                    t
COPY . .

CMD ["python", "app.py"]
```

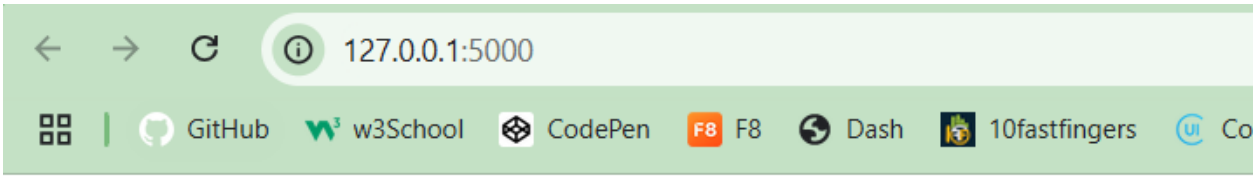
docker-compose.yml

```
version: "3.8"

services:
    flask-ap :
    p   build: .
        container_nam : flask-containe
        ports:          r
            - "5000:500
        restart: always
```

- Build và chạy ứng dụng



- Kiểm tra



Bài 10: Lưu trữ dữ liệu với Docker Volumes

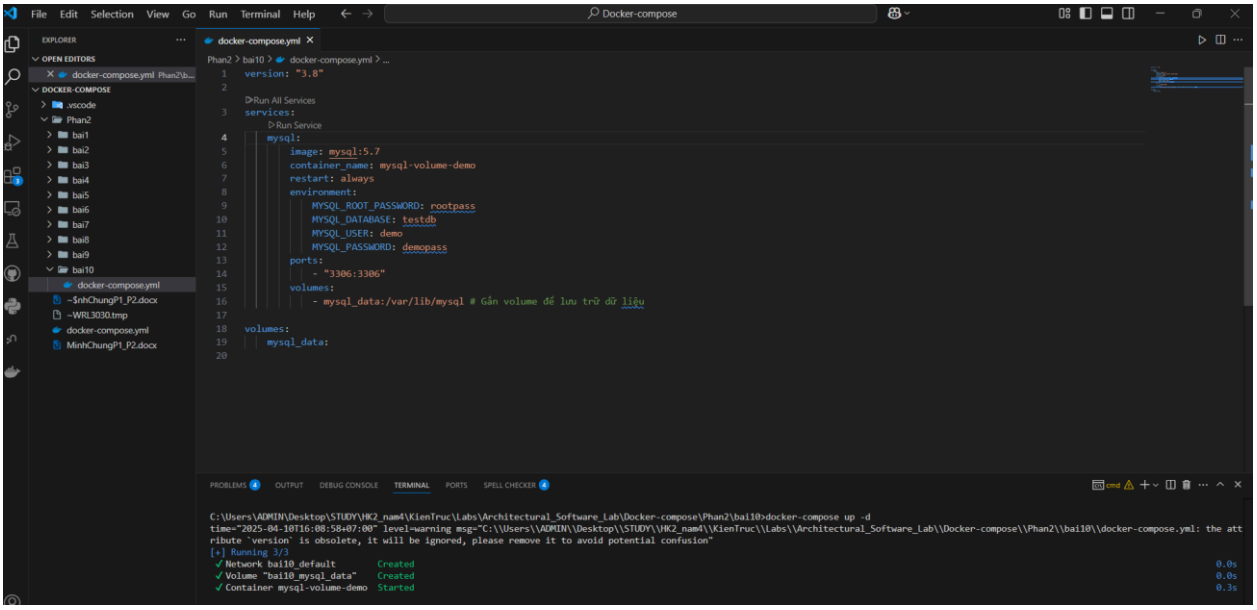
Yêu cầu:

Chạy MySQL và gắn volume để dữ liệu không bị mất.

- Tạo docker-compose.yml

```
version: "3.8"

services:
    mysql:
        image: mysql:5.
        container_nam : mysql-volume-dem
        restart: always o
        environmen :
        t    MYSQL_ROOT_PASSWOR : rootpass
             MYSQL_DATABAS : testdb
             MYSQL_USE : demo
             MYSQL_PASSWOR : demopass
        ports:
             - "3306:330
        volumes:
             - mysql_data:/var/lib/mysq  # Gắn volume để lưu trữ dữ liệ
               l                         u
volumes:
    mysql_dat :
    a
```

- Build và run



- Kiểm tra

```
C:\Users\ADMIN\Desktop\STUDY\HK2_nam4\KienTruc\Labs\Architectural_Software_Lab\Docker-compose\Phan2\bai10>docker volume ls
DRIVER    VOLUME NAME
local     1a9fdfb408d477d16f22c2db230d1867696102637852ed79e7bd9d62c88c578e
local     bai5_redis_data
local     bai6_mysql_data
local     bai6_wordpress_data
local     bai7_mongo_data
local     bai8_mysql_data
local     bai10_mysql_data
local     e719c80828e207228820f86e74173fca2119bbf679a2998995c3700714d27898
```

Phần 3: Docker Compose file

Bài tập 1: Triển khai WordPress với MySQL

Mục tiêu: Tạo stack WordPress kết nối với MySQL, sử dụng volumes để lưu trữ dữ liệu.

Yêu cầu:

1. Sử dụng image wordpress:latest (port 80).
2. Sử dụng image mysql:5.7 (port 3306).
3. Volume cho database (/var/lib/mysql).
4. Biến môi trường cho MySQL:

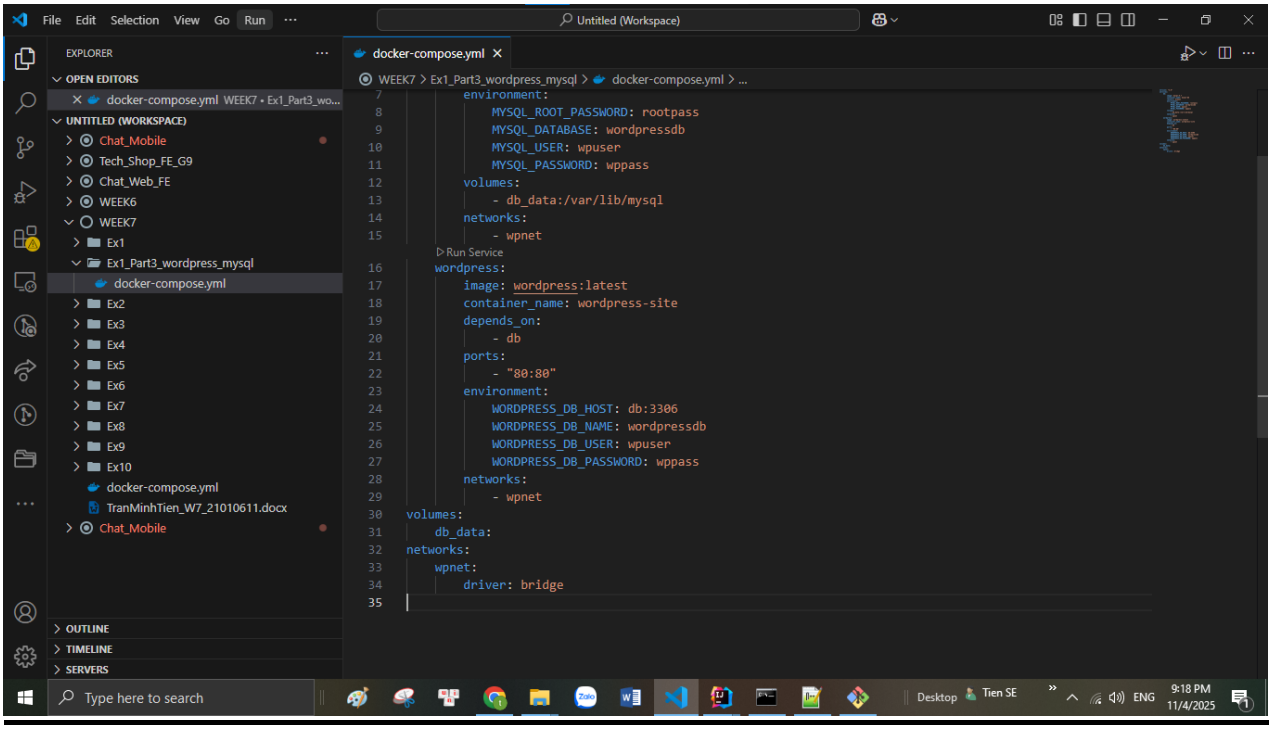
MYSQL_ROOT_PASSWORD, MYSQL_DATABASE, MYSQL_USER, MYSQL_PASSWORD

Gợi ý:

WordPress cần khai báo depends_on MySQL.

Sử dụng network tùy chỉnh để kết nối giữa 2 service.

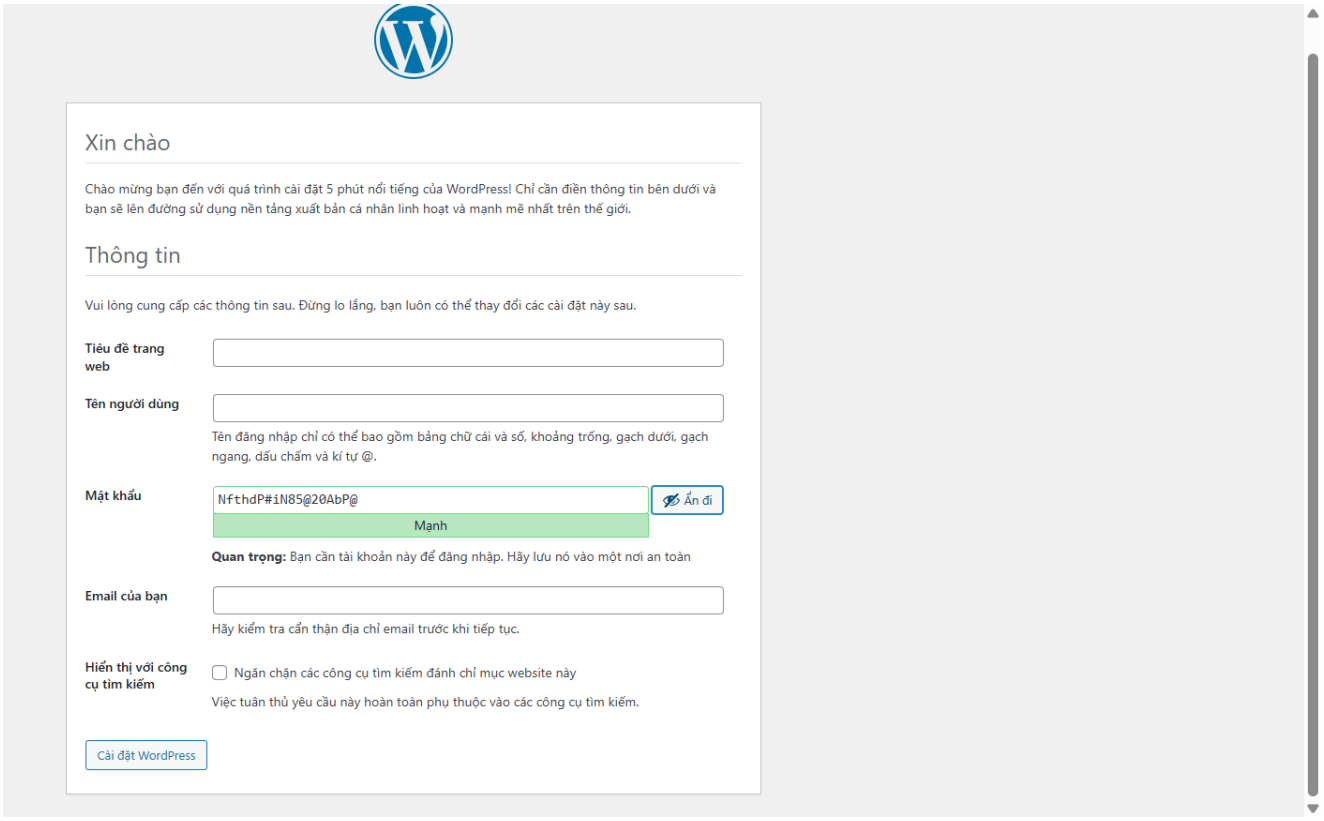
B1: Tạo docker-compose.yml

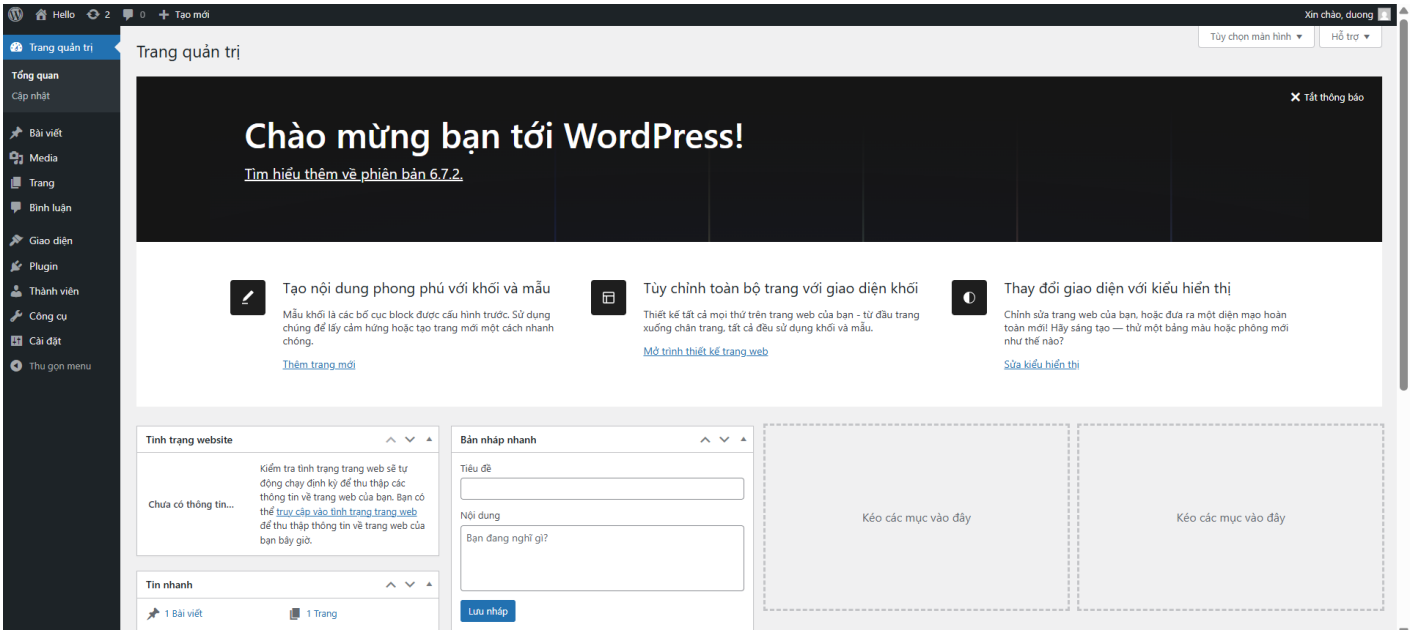


B2: Build và run

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex1_Part3_wordpress_mysql>docker-compose up -d
time="2025-04-11T21:26:07+07:00" level=warning msg="D:\\0_GIT_KIENTRUC\\TranMinhTien-Architectural-Software\\WEEK7\\Ex1_
Part3_wordpress_mysql\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to
avoid potential confusion"
unable to get image 'mysql:5.7': error during connect: Get "http://%2F%2F.%2Fpipe%2FdockerDesktopLinuxEngine/v1.48/image
s/mysql:5.7/json": open //./pipe/dockerDesktopLinuxEngine: The system cannot find the file specified.

[+] Running 0/2UC\TranMinhTien-Architectural-Software\WEEK7\Ex1_Part3_wordpress_mysql>docker-compose up -d
 - db Pulling                                                                                    4.8s
 - wordpress Pulling                                                                             4.8s
```

B3: Truy cập localhost



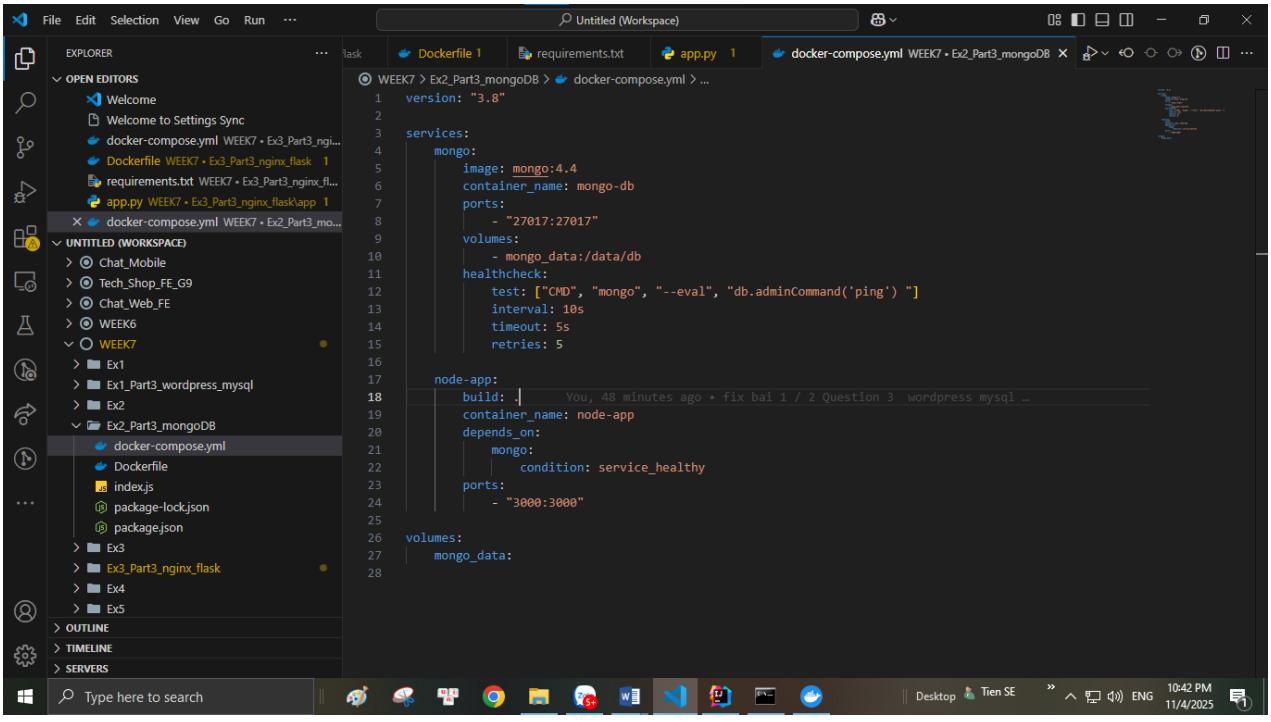


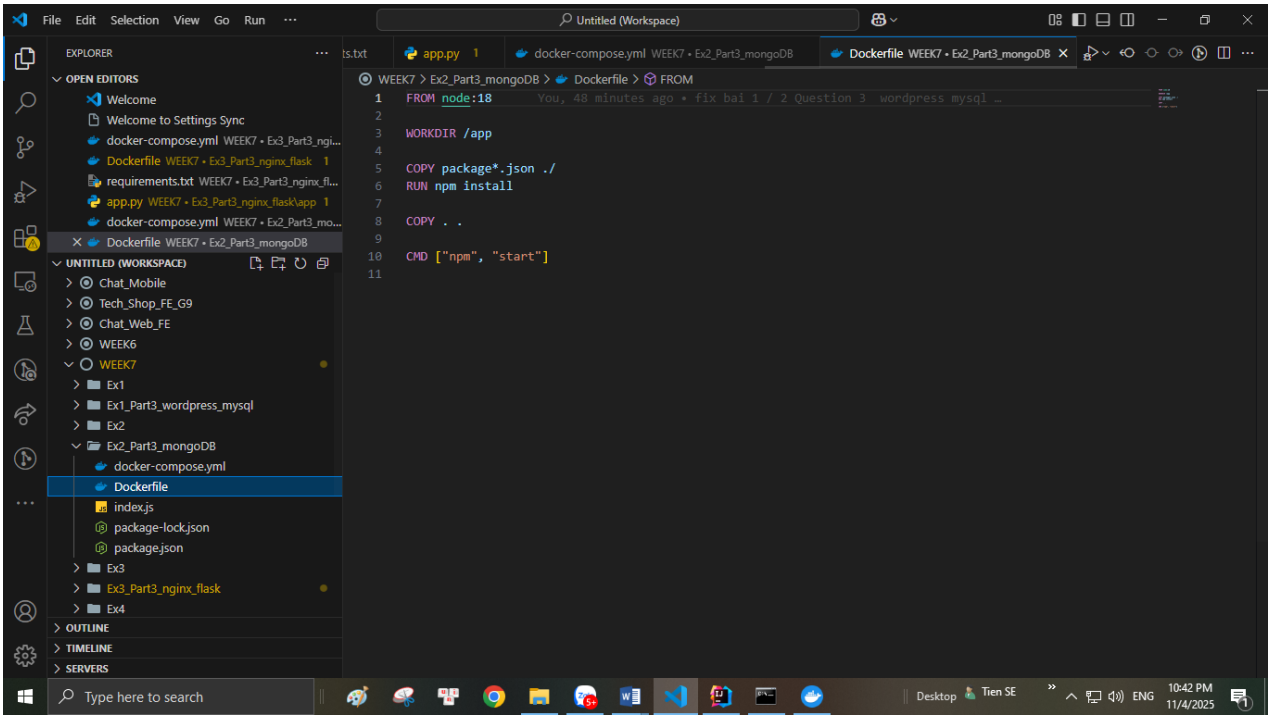
Bài tập 2: Ứng dụng Node.js + MongoDB

Mục tiêu: Triển khai ứng dụng Node.js (lưu dữ liệu vào MongoDB) và MongoDB với volume.

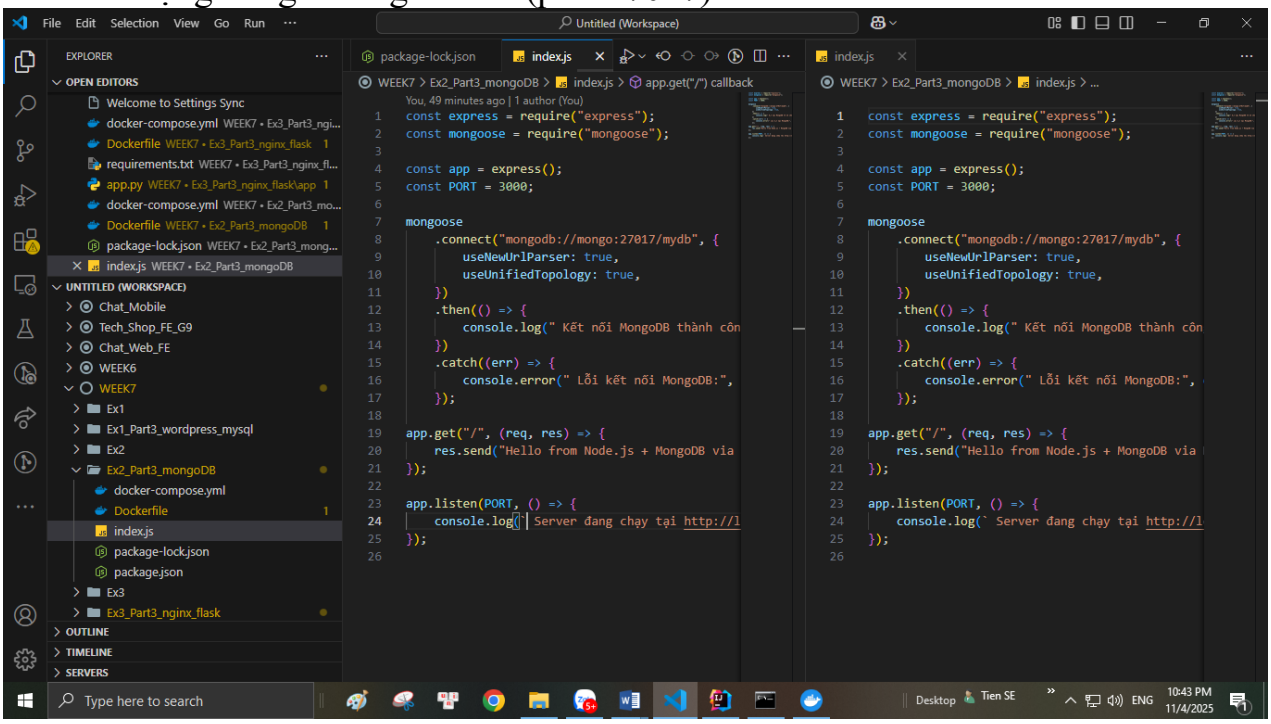
Yêu cầu:

1. Viết Dockerfile cho ứng dụng Node.js (ví dụ: REST API đơn giản).



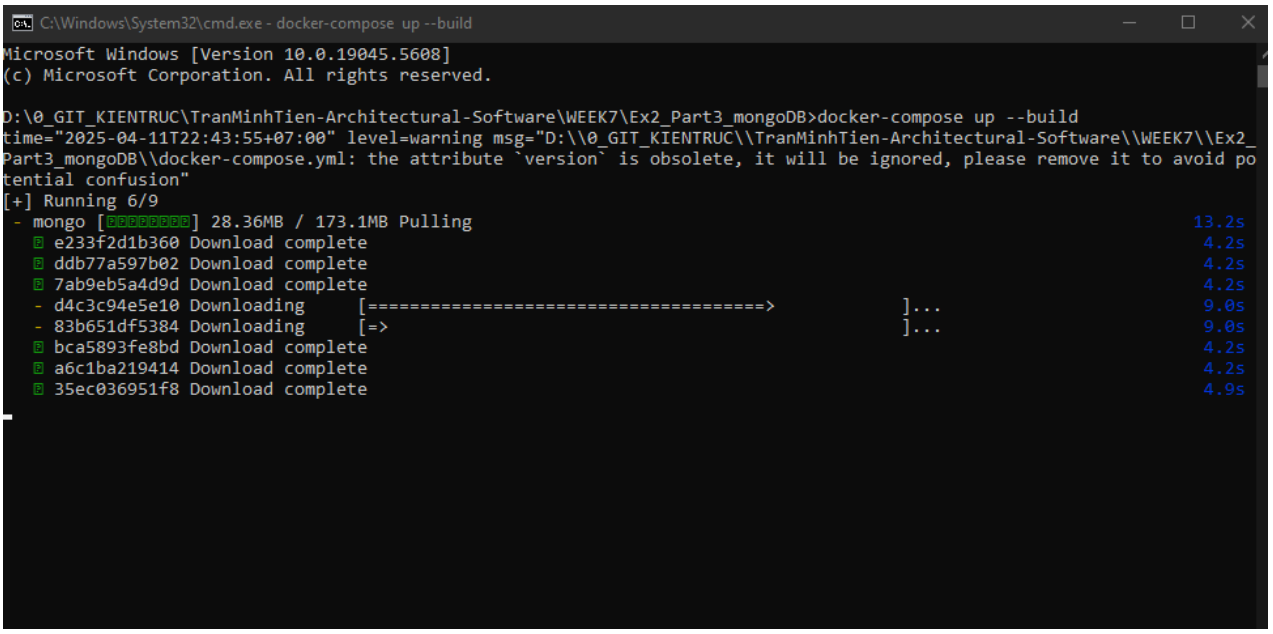


2. Sử dụng image mongo:latest (port 27017).

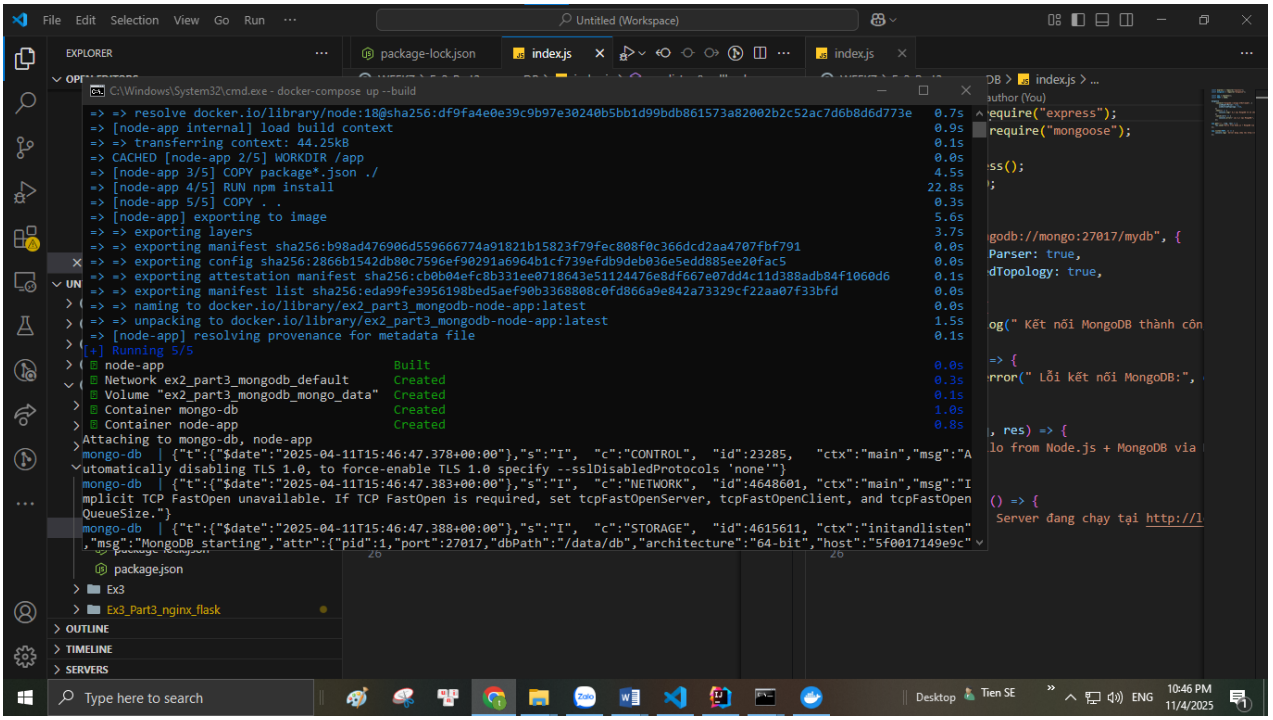


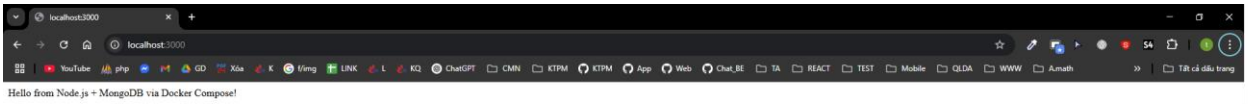
3. Volume cho MongoDB (/data/db).

```
D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex2_Part3_mongoDB>docker-compose up --build
time="2025-04-11T22:43:55+07:00" level=warning msg="D:\\0_GIT_KIENTRUC\\TranMinhTien-Architectural-Software\\WEEK7\\Ex2_
Part3_mongoDB\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid po
tential confusion"
[+] Running 0/1
 - mongo Pulling                                                                                          3.1s
```



4. Đảm bảo Node.js service khởi động sau MongoDB (depends_on + healthcheck).





Bài tập 3: Load Balancing với Nginx + Flask

Mục tiêu: Cân bằng tải giữa 2 instance Flask dùng Nginx.

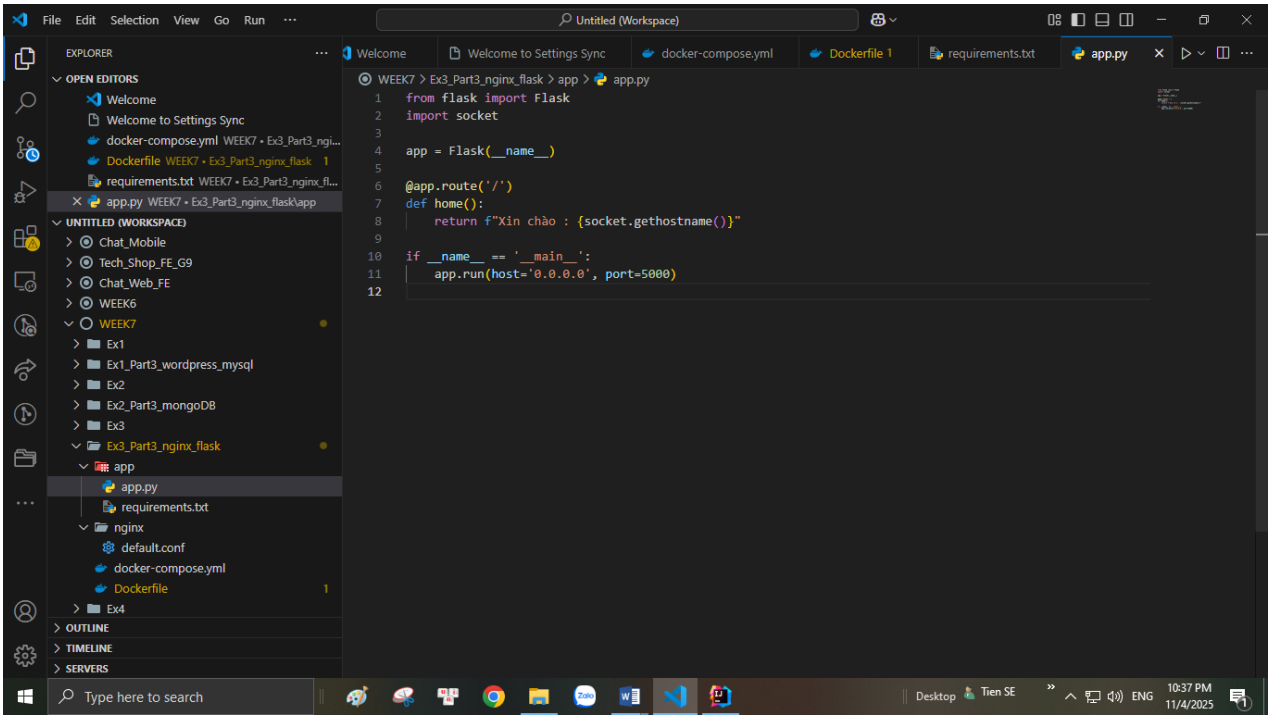
Yêu cầu:

1. 2 service Flask (sử dụng app.py từ bài tập trước, port 5000).
2. 1 service Nginx (port 8080) cấu hình làm reverse proxy:

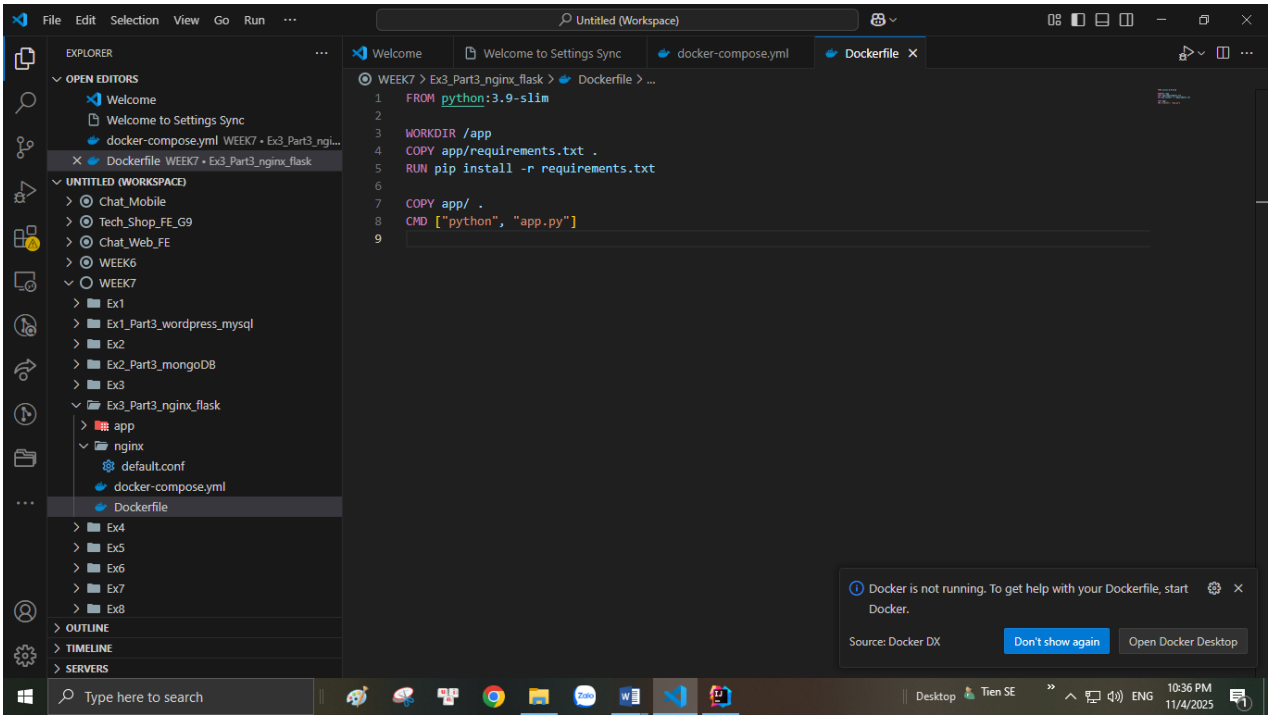
Chuyển request / đến các Flask instance (round-robin).

3. Tạo custom network và Nginx config.

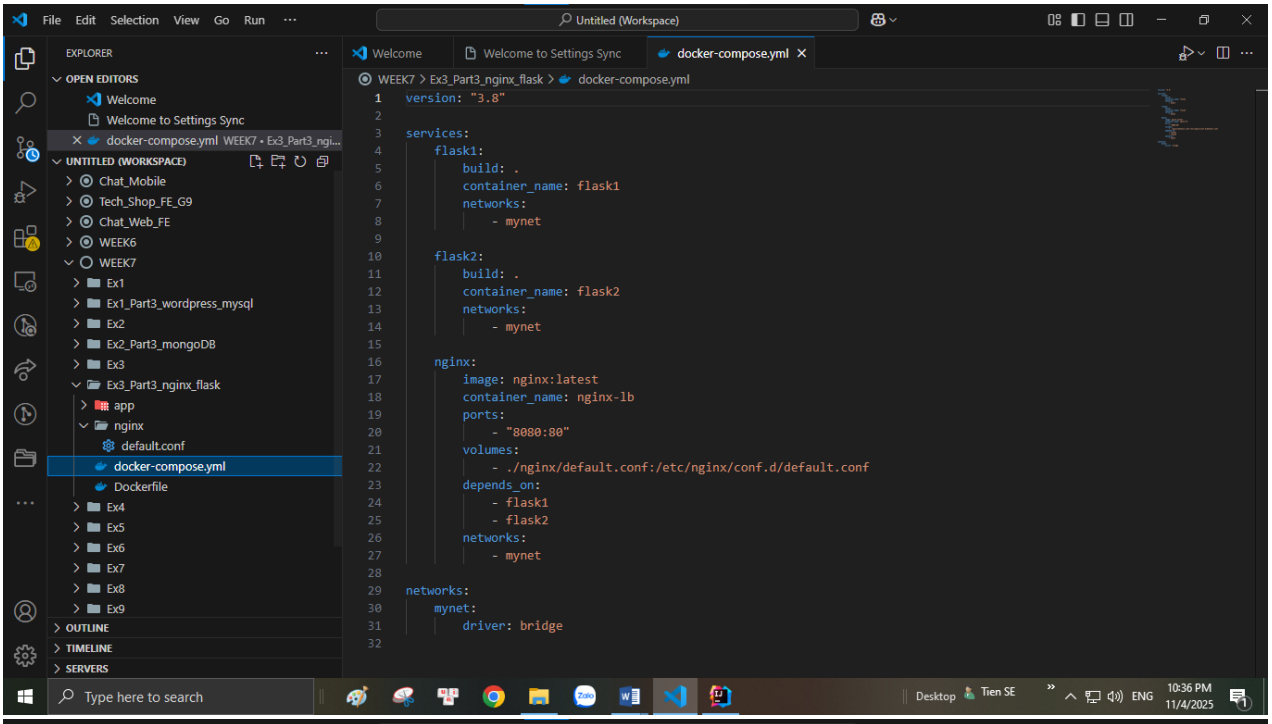
B1: Tạo project nodejs app.

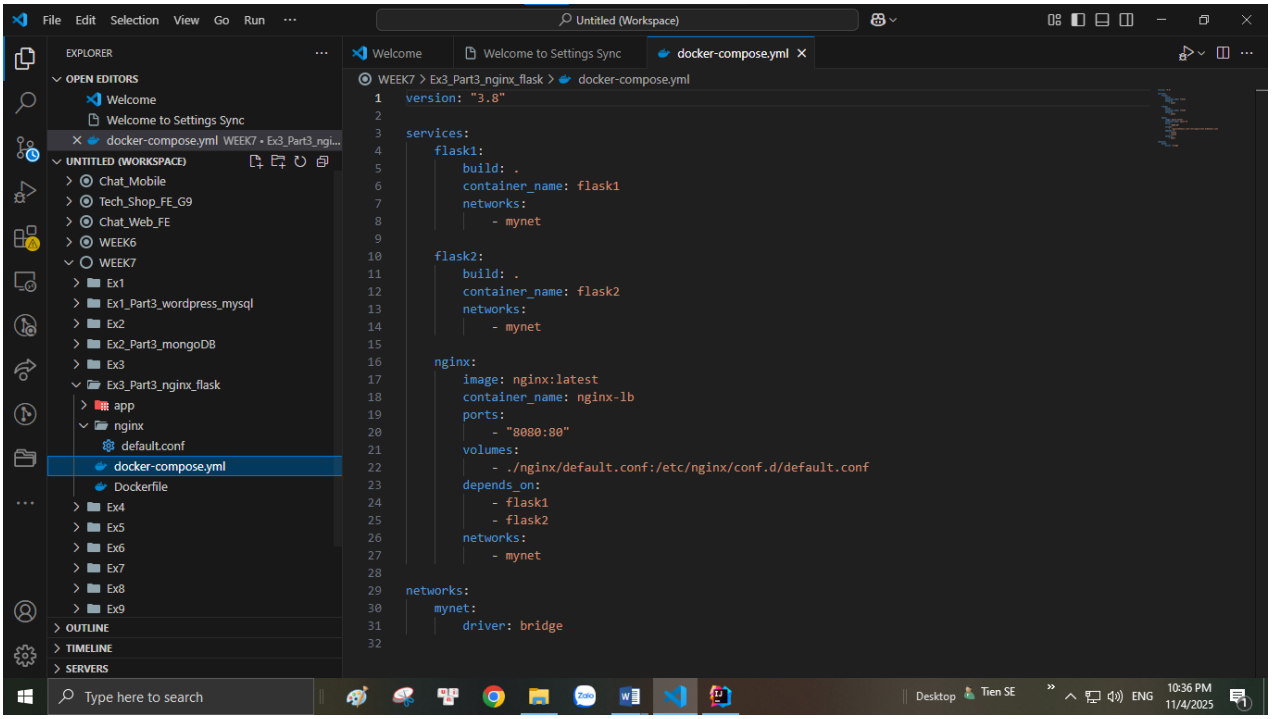


Dockerfile



docker-compose.yml





Bài tập 4: Prometheus + Grafana Monitoring

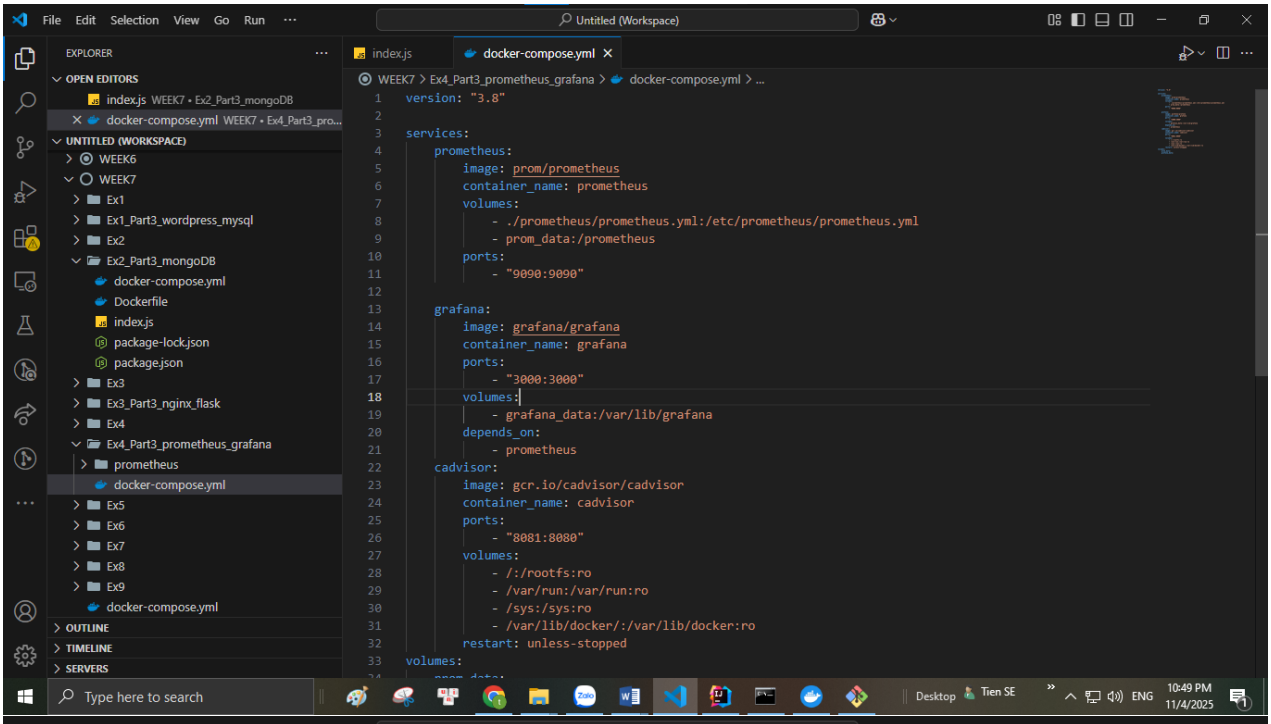
Mục tiêu: Giám sát Docker containers dùng Prometheus và Grafana.

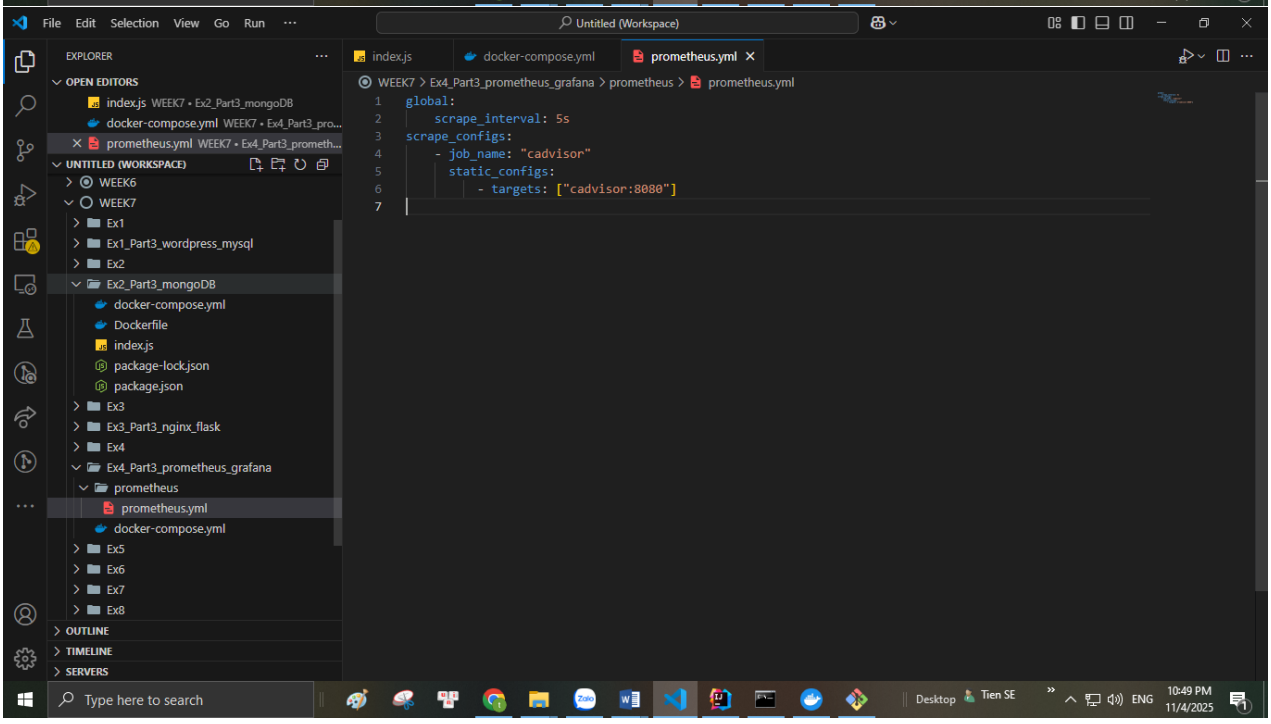
Yêu cầu:

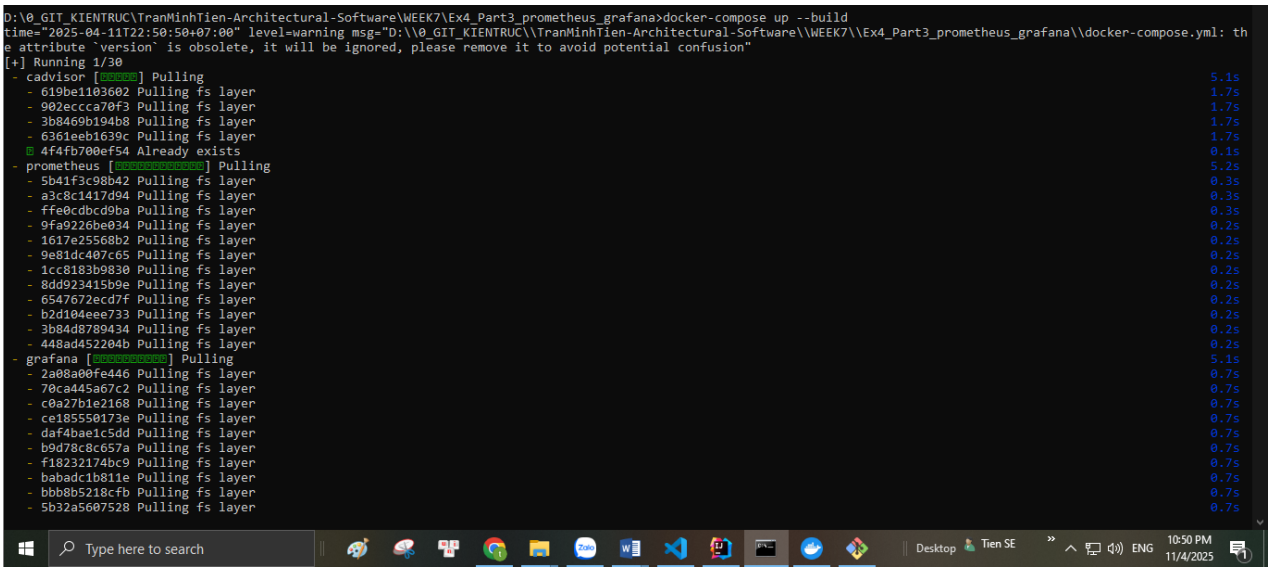
1. Service Prometheus (port 9090) với file cấu hình thu thập metrics từ Docker.

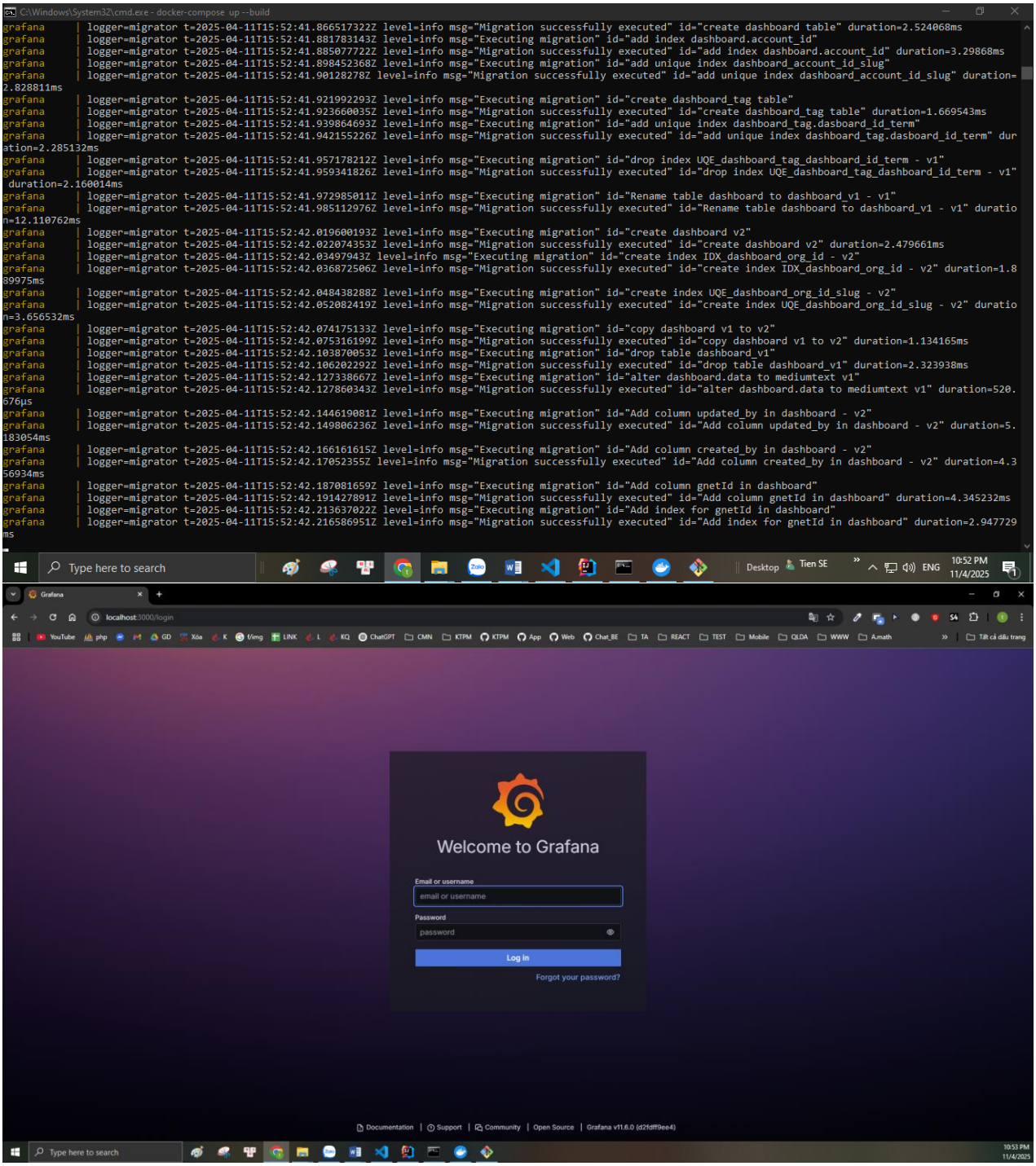
2. Service Grafana (port 3000) kết nối đến Prometheus.

3. Volume để lưu dữ liệu Prometheus và Grafana.






D:\0_GIT_KIENTRUC\TranMinhTien-Architectural-Software\WEEK7\Ex4_Part3_prometheus_grafana>docker-compose up --build
time="2025-04-11T22:50:50+07:00" level=warning msg="D:\\0_GIT_KIENTRUC\\TranMinhTien-Architectural-Software\\WEEK7\\Ex4_Part3_prometheus_grafana\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 1/30
- cadvisor [] Pulling 5.1s
- 619be1103602 Pulling fs layer 1.7s
- 902eccca70f3 Pulling fs layer 1.7s
- 3b8469b194b8 Pulling fs layer 1.7s
- 6361eeb1639c Pulling fs layer 1.7s
4f4fb700ef54 Already exists 0.1s
- prometheus [] Pulling 5.2s
- 5b41f3c98b42 Pulling fs layer 0.3s
- a3c8c1417d94 Pulling fs layer 0.3s
- ffe0cdbcd9ba Pulling fs layer 0.3s
- 9fa9226be034 Pulling fs layer 0.2s
- 1617e25568b2 Pulling fs layer 0.2s
- 9e81dc407c65 Pulling fs layer 0.2s
- 1cc8183b9830 Pulling fs layer 0.2s
- 8dd923415b9e Pulling fs layer 0.2s
- 6547672ecd7f Pulling fs layer 0.2s
- b2d104eee733 Pulling fs layer 0.2s
- 3b84d8789434 Pulling fs layer 0.2s
- 448ad452204b Pulling fs layer 0.2s
- grafana [] Pulling 5.1s
- 2a08a00fe446 Pulling fs layer 0.7s
- 70ca445a67c2 Pulling fs layer 0.7s
- c0a27b1e2168 Pulling fs layer 0.7s
- ce185550173e Pulling fs layer 0.7s
- daf4bae1c5dd Pulling fs layer 0.7s
- b9d78c8c657a Pulling fs layer 0.7s
- f18232174bc9 Pulling fs layer 0.7s
- babadc1b811e Pulling fs layer 0.7s
- bbb8b5218cfb Pulling fs layer 0.7s
- 5b32a5607528 Pulling fs layer 0.7s
Type here to search
Desktop
Tien SE
ENG
10:50 PM
11/4/2025



Bài tập 5: Multi-tier Voting App
Mục tiêu: Triển khai ứng dụng voting gồm 5 services (Tham khảo từ Docker Docs).

Yêu cầu:

1. Frontend: vote (Python, port 5000).
2. Backend: result (Node.js, port 5001).
3. Redis (lưu tạm vote).
4. Worker (Java) xử lý vote từ Redis sang DB.
5. Postgres (lưu kết quả).

Bài tập 6: CI/CD Pipeline với Docker Compose
Mục tiêu: Mô phỏng pipeline dev/test bằng Docker Compose.

Yêu cầu:

1. Service app (Python/Node.js) với code được mount từ host (development mode).

2. Service tests chạy unit tests khi code thay đổi (sử dụng volumes + entrypoint).

3. Service nginx (production mode) dùng image build sẵn từ app.

Gợi ý:

1. Dùng 2 file compose khác nhau (docker-compose-dev.yml và docker-compose-prod.yml).
2. Sử dụng docker-compose -f <file> up để chọn môi trường.

Bài tập 7: Elasticsearch + Kibana
Mục tiêu: Triển khai ELK stack đơn giản.
Yêu cầu:

1. Service Elasticsearch (port 9200) với volume.
2. Service Kibana (port 5601) kết nối với Elasticsearch.
3. Thiết lập environment variables cho credentials.

Bài tập 8: Django + Celery + Redis
Mục tiêu: Triển khai Django với Celery worker và Redis làm message broker.

Yêu cầu:

1. Django app (port 8000).
2. Celery worker chạy song song.
3. Redis service cho task queue.

Bài tập 9: Nextcloud với MariaDB + Redis Caching
Mục tiêu: Triển khai Nextcloud (self-hosted cloud) với MariaDB và Redis.

Yêu cầu:

1. Nextcloud (port 80).
2. MariaDB (volume cho dữ liệu).
3. Redis cache.

Bài tập 10: Traefik as Reverse Proxy
Mục tiêu: Dùng Traefik để định tuyến request đến các service (Flask, WordPress, etc.).

Yêu cầu:

1. Cấu hình Traefik với Docker provider.
2. Đặt labels cho services để Traefik nhận diện.

Tips khi làm bài tập:

1. Luôn dùng docker-compose down -v để xóa volumes khi test lại.

2. Kiểm tra log bằng docker-compose logs <service>.
3. Sử dụng docker-compose config để validate file YAML.