
Tugas Hasil Uji Coba

Sistem Paralel dan Terdistribusi A Docker Compose



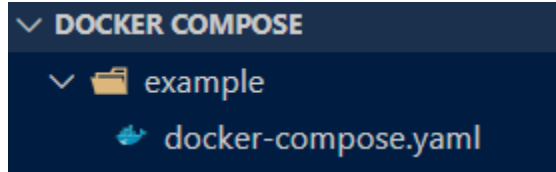
Disusun Oleh :

Zakaria Fattawari 11231092

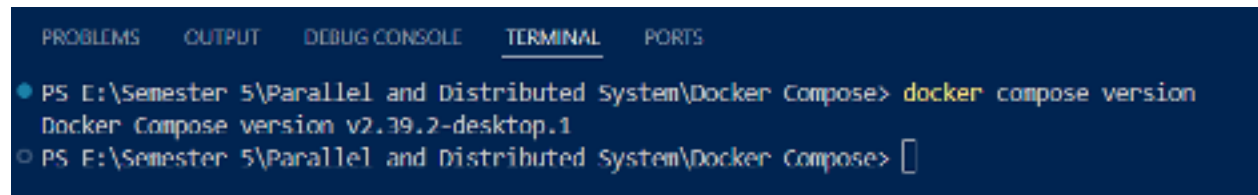
10 November 2025

DOCKER COMPOSE

Apa itu Docker Compose?



Docker Compose: Version



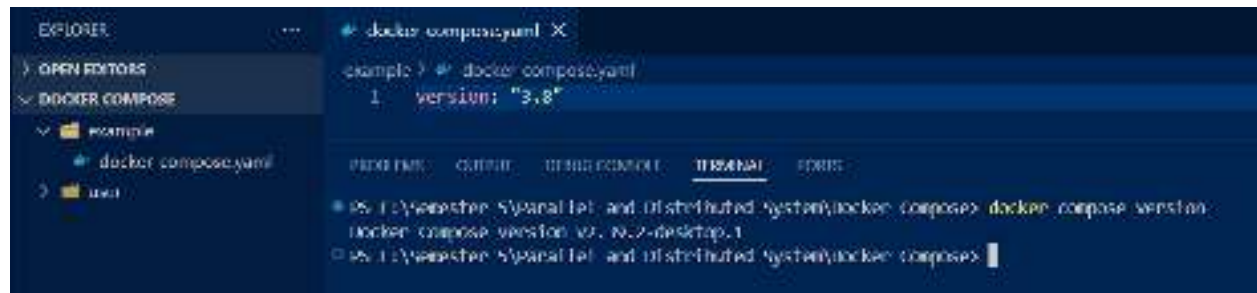
```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
● PS E:\Semester 5\Parallel and Distributed System\Docker Compose> docker compose version
  Docker Compose version v2.39.2-desktop.1
○ PS E:\Semester 5\Parallel and Distributed System\Docker Compose> 
```



CONFIGURATION FILE

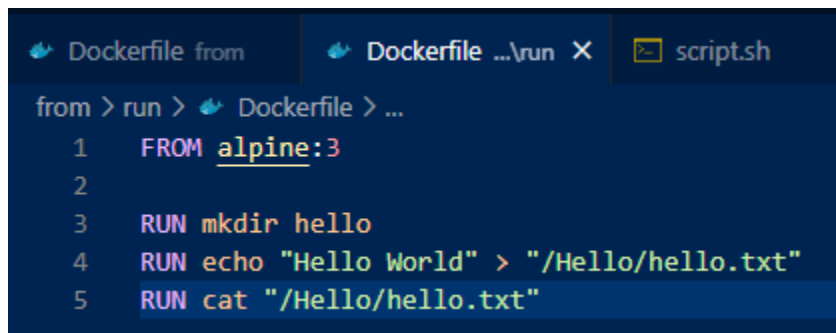
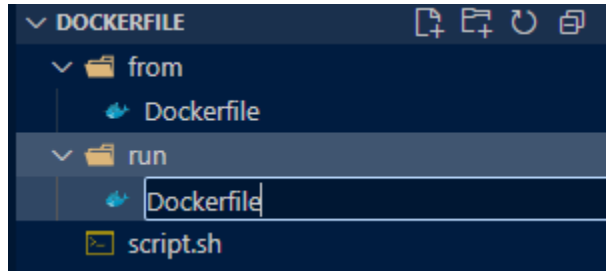
Docker Compose: Versi Konfigurasi

Docker Compose: Docker Compose File




YAML

Pengenalan YAML



Docker Compose: YAML Attribute

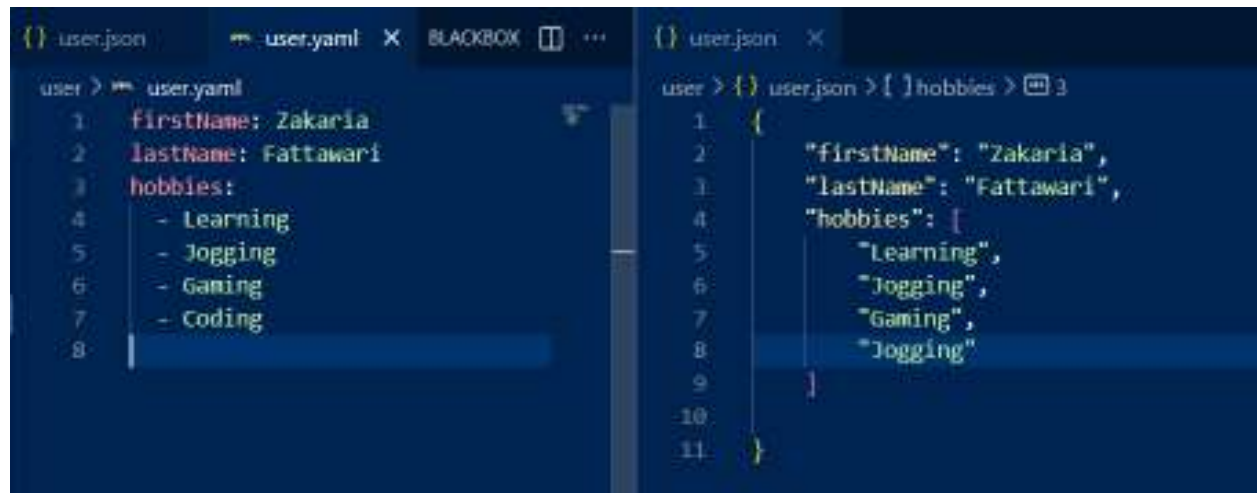


The image shows a code editor with two panels. The left panel displays a Docker Compose YAML file named `user.yaml`. The right panel displays a JSON file named `user.json`.

```
user > user.yaml
1 firstName: Zakaria
2 lastName: Fattawari
3 S|
```

```
user > {} user.json > ...
1 {
2   "firstName": "Zakaria",
3   "lastName": "Fattawari"
4 }
```

Docker Compose: YAML Array



```
user > user.yaml
1 firstName: Zakaria
2 lastName: Fattawari
3 hobbies:
4   - Learning
5   - Jogging
6   - Gaming
7   - Coding
8

user > user.json
1 {
2   "firstName": "Zakaria",
3   "lastName": "Fattawari",
4   "hobbies": [
5     "Learning",
6     "Jogging",
7     "Gaming",
8     "Jogging"
9   ]
10 }
11
```

=

Docker Compose: YAML Nested Object

```
user> cat user.yaml
1 first_name: Zakaria
2 last_name: Fattowari
3 hobbies:
4   - Learning
5   - Dogging
6   - Gaming
7   - Coding
8 address:
9   street: Jalan Soekarno Hatta KM 24
10  city: North Balikpapan
11  province: East Kalimantan
12  country: Indonesia
13  postalCode: "76122"
14  phoneNumber: "855651169435"

user> cat user.json > | address > | phoneNumber
1 {
2   "first_name": "Zakaria",
3   "last_name": "Fattowari",
4   "hobbies": [
5     "Learning",
6     "Dogging",
7     "Gaming",
8     "Coding"
9   ],
10  "address": {
11    "street": "Jalan Soekarno Hatta KM 24",
12    "city": "North Balikpapan",
13    "province": "East Kalimantan",
14    "country": "Indonesia",
15    "postalCode": "76122",
16    "phoneNumber": "855651169435"
17  }
18 }
```

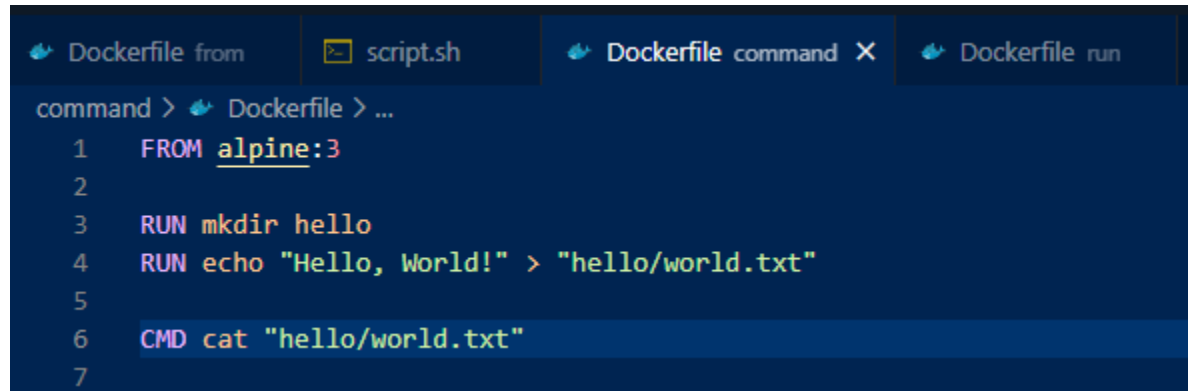
Docker Compose: YAML Array Nested Object

```
user > cat user.yaml
1 firstName: Zakaria
2 lastName: Fattawari
3 hobbies:
4   - Learning
5   - Juggling
6   - Gaming
7   - Coding
8 address:
9   street: Jalan Sukarno Hatta KM 24
10  city: North Balikpapan
11  province: East Kalimantan
12  country: Indonesia
13  postalCode: "76122"
14  phoneNumber: "85551144108"
15  emails:
16    - zakariafattawari@gmail.com
17    - alathawaripenitk@gmail.com
18    - 1121188@student.itk.ac.id
19 wallet:
20   type: cash
21   amount: 50000
22   - type: credit
23     amount: 42000

user > cat user.json
1 {}
2 {"firstName": "Zakaria",
3  "lastName": "Fattawari",
4  "hobbies": [
5    "Learning",
6    "Juggling",
7    "Gaming",
8    "Coding"
9  ],
10 "address": {
11   "street": "Jalan Sukarno Hatta KM 24",
12   "city": "North Balikpapan",
13   "province": "East Kalimantan",
14   "country": "Indonesia",
15   "postalCode": "76122",
16   "phoneNumber": "85551144108",
17   "emails": [
18     "zakariafattawari@gmail.com",
19     "alathawaripenitk@gmail.com",
20     "1121188@student.itk.ac.id"
21   ]
22 },
23 "wallet": {
24   "type": "cash",
25   "amount": 50000
26 },
27 {
28   "type": "credit",
29   "amount": 42000
30 }
31 }
32 }
```

MEMBUAT CONTAINER



Membuat Konfigurasi Kontainer




The image shows a Dockerfile editor interface with four tabs: "Dockerfile from", "script.sh", "Dockerfile command X", and "Dockerfile run". The "Dockerfile command X" tab is active, displaying a Dockerfile with the following content:

```
command > Dockerfile > ...  
1 FROM alpine:3  
2  
3 RUN mkdir hello  
4 RUN echo "Hello, World!" > "hello/world.txt"  
5  
6 CMD cat "hello/world.txt"  
7
```

Docker Compose: Membuat Container

 docker-compose.yml 

example >  docker-compose.yml

1 **version: "3.8"**

2

▶Run All Services

3 **services:**

▶Run Service

4 **nginx-example:**

5 **container_name: nginx-example**

6 **image: nginx:latest**

PS C:\Programmer\Software\docker> docker-compose up --help

PS C:\Programmer\Software\docker> docker-compose up --help

Hint: "docker-compose up" does nothing by default. To start containers, you need to specify the "up" command. To start containers in the background, you need to specify the "d" flag. To start containers in the foreground, you need to specify the "f" flag. To start containers in the background and attach to their logs, you need to specify the "f" flag and the "l" flag. To start containers in the background and attach to their logs and also to the Docker daemon logs, you need to specify the "f" flag, the "l" flag, and the "D" flag.

[1] Creating...

PS C:\Programmer\Software\docker> docker-compose up

PS C:\Programmer\Software\docker> docker-compose up

PS C:\Programmer\Software\docker> docker-compose up --help

CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES

PS C:\Programmer\Software\docker> docker-compose up --help

8257226622 nginx-example

PS C:\Programmer\Software\docker> docker-compose up --help

PS C:\Programmer\Software\docker> docker-compose up --help

PS C:\Programmer\Software\docker> docker-compose up --help

0: daniel@kali:~/git/00

SAVED

sample 1 # intro-example

1 version: "1.0"

2 description:

3 name: intro

4 platform:

5 #this example:

6 #contains a user agent example

7 #agent: user-agent

8 #this is:

9 #this is:

10 #this is: user-agent

11 #agent: user-agent

12 #this is: user-agent

13 #this is: user-agent

14 #this is: user-agent

15 #this is: user-agent

16 #this is: user-agent

17 #this is: user-agent

18 #this is: user-agent

19 #this is: user-agent

20

21 #this is: user-agent

22

23 #this is: user-agent

24 #this is: user-agent

25 #this is: user-agent

26 #this is: user-agent

27

28 #this is: user-agent

29 #this is: user-agent

30

31 #this is: user-agent

32

33 #this is: user-agent

34 #this is: user-agent

35 #this is: user-agent

36 #this is: user-agent

37 #this is: user-agent

38 #this is: user-agent

39 #this is: user-agent

40 #this is: user-agent

41 #this is: user-agent

42 #this is: user-agent

43 #this is: user-agent

44 #this is: user-agent

45 #this is: user-agent

46 #this is: user-agent

47 #this is: user-agent

48 #this is: user-agent

49 #this is: user-agent

50 #this is: user-agent

51 #this is: user-agent

52 #this is: user-agent

53 #this is: user-agent

54 #this is: user-agent

55 #this is: user-agent

56 #this is: user-agent

57 #this is: user-agent

58 #this is: user-agent

59 #this is: user-agent

60 #this is: user-agent

61 #this is: user-agent

62 #this is: user-agent

63 #this is: user-agent

64 #this is: user-agent

65 #this is: user-agent

66 #this is: user-agent

67 #this is: user-agent

68 #this is: user-agent

69 #this is: user-agent

70 #this is: user-agent

71 #this is: user-agent

72 #this is: user-agent

73 #this is: user-agent

74 #this is: user-agent

75 #this is: user-agent

76 #this is: user-agent

77 #this is: user-agent

78 #this is: user-agent

79 #this is: user-agent

80 #this is: user-agent

81 #this is: user-agent

82 #this is: user-agent

83 #this is: user-agent

84 #this is: user-agent

85 #this is: user-agent

86 #this is: user-agent

87 #this is: user-agent

88 #this is: user-agent

89 #this is: user-agent

90 #this is: user-agent

91 #this is: user-agent

92 #this is: user-agent

93 #this is: user-agent

94 #this is: user-agent

95 #this is: user-agent

96 #this is: user-agent

97 #this is: user-agent

98 #this is: user-agent

99 #this is: user-agent

100 #this is: user-agent

© 2006 The Authors

```

PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container create --name minimal zskortalsattaword/minimal
sha256:540f1b1b44f927361e87268e1f4176d1014a2c4a6240
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container start minimal
command
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container ls
CONTAINER ID   IMAGE      COMMAND   CREATED   STATUS    PORTS   NAMES
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container ls | findstr command
d508b5588188   zskortalsattaword/command   /bin/sh -c 'cat /dev/urandom | tr -dc 'a-z0-9' | fold -w 64 | xargs printf "%s\n" ' About a minute ago   Exited (0) About a minute ago   command
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container logs command
Hello, World!
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container start command
command
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container start command
command
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles> docker container logs command
Hello, World!
Hello, World!
Hello, World!
Hello, World!
PS E:\Sonarstor\SVParallel and Distributed System\dockerfiles>

```

MENJALANKAN CONTAINER

Pengenalan Menjalankan Container

```
PS E:\Semester 5\Parallel and Distributed System\Docker Compose> cd example
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose start
time="2025-11-18T23:37:43+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
  ✓ Container nginx-example2 Started                                0.4s
  ✓ Container nginx-example Started                                0.4s
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker container ls
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS        NAMES
5a03ba06d600   nginx:latest   "/docker-entrypoint..." 2 minutes ago  Up 5 seconds  80/tcp       nginx-example2
9851726e90b6   nginx:latest   "/docker-entrypoint..." 8 minutes ago  Up 5 seconds  80/tcp       nginx-example
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example>
```

```
label > Dockerfile > ...
1  FROM alpine:3
2
3  LABEL author="Zakaria Fattawari"
4  LABEL company="A206" website="https://www.instagram.com/alathereonn_"
5
6  RUN mkdir hello
7  RUN echo "Hello, World!" > "hello/world.txt"
8  CMD cat "hello/world.txt"
9
10
```

Docker Compose: Melihat Container

```
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose ps
times="2025-11-18T23:48:13+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"

```

NAME	IMAGE	COMMAND	SERVICE	CREATED	STATUS	PORTS
nginx-example	nginx:latest	"/docker-entrypoint..."	nginx-example	10 minutes ago	Up 2 minutes	88/tcp
nginx-example2	nginx:latest	"/docker-entrypoint..."	nginx-example2	4 minutes ago	Up 2 minutes	88/tcp

```
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example>
```


Docker Compose: Menghentikan Container

```
* PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose stop
time="2025-11-18T23:41:40:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[*] Stopping 2/2
  ✓ Container nginx-example2 Stopped      0.3s
  ✓ Container nginx-example Stopped      0.3s
* PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose ps
time="2025-11-18T23:41:40:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME                IMAGE                COMMAND             SERVICE     CREATED     STATUS      PORTS
```

Docker Compose: Menghapus Container

```
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose start
time="2025-11-18T23:43:36+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
  ✓ Container nginx-example Started                                0.2s
  ✓ Container nginx-example2 Started                               0.2s
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose ps
time="2025-11-18T23:43:54+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME                IMAGE              COMMAND                                SERVICE      CREATED          STATUS          PORTS
nginx-example       nginx:latest       "/docker-entrypoint..."             nginx-example 14 minutes ago   Up 17 seconds   80/tcp
nginx-example2      nginx:latest       "/docker-entrypoint..."             nginx-example2  8 minutes ago   Up 17 seconds   80/tcp
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose down
time="2025-11-18T23:44:00+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 3/3
  ✓ Container nginx-example2 Removed                                0.4s
  ✓ Container nginx-example Removed                                0.3s
  ✓ Network example_default Removed                                0.3s
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> |
```

PROJECT NAME

Pengenalan Project Name

```
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose create
time="2025-11-10T23:46:51+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the at
tribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[*] Creating 3/3
  ✓ Network example_default Created 0.0s
  ✓ Container nginx-example2 created 0.1s
  ✓ Container nginx-example Created 0.1s
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose ps
time="2025-11-10T23:46:56+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the at
tribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME          IMAGE          COMMAND          SERVICE          CREATED          STATUS          PORTS
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose start
time="2025-11-10T23:47:04+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the at
tribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[*] Starting 2/2
  ✓ Container nginx-example2 Started 0.2s
  ✓ Container nginx-example Started 0.2s
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose ps
time="2025-11-10T23:47:10+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the at
tribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME          IMAGE          COMMAND          SERVICE          CREATED          STATUS          PORTS
nginx-example  nginx:latest   "/docker-entrypoint..." nginx-example  19 seconds ago  Up 6 seconds  80/tcp
nginx-example2 nginx:latest   "/docker-entrypoint..." nginx-example2  19 seconds ago  Up 6 seconds  80/tcp
PS E:\Semester 5\Parallel and Distributed System\docker compose\example>
```

```
PS E:\Semester 5\Parallel and Distributed System\docker compose\example> docker compose ps
time="2025-11-10T23:47:10+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\docker compose\example\docker-compose.yaml: the at
tribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
NAME          IMAGE          COMMAND          SERVICE          CREATED          STATUS          PORTS
nginx-example  nginx:latest   "/docker-entrypoint..." nginx-example  19 seconds ago  Up 6 seconds  80/tcp
nginx-example2 nginx:latest   "/docker-entrypoint..." nginx-example2  19 seconds ago  Up 6 seconds  80/tcp
PS E:\Semester 5\Parallel and Distributed System\docker compose\example>
```

SERVICES

Pengenalan SERVICES

```
docker-compose.yml X
example > docker-compose.yml
1  version: "3.9"
2
   ▶ Run All Services
3  services:
   ▶ Run Service
4  | nginx-example:
5  |   image: nginx:latest
6  |   container_name: nginx-example
   ▶ Run Service
7  | mongodb-example:
8  |   image: mongo:latest
9  |   container_name: mongodb-example
10
```

Docker Compose: SERVICES

```
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose create
time="2025-11-10T23:54:16+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 3/3
  ✓ Network example_default    Created                                0.0s
  ✓ Container mongodb-example  Created                                0.1s
  ✓ Container nginx-example    Created                                0.1s
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose start
time="2025-11-10T23:54:29+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
  ✓ Container mongodb-example  Started                                0.2s
  ✓ Container nginx-example    Started                                0.2s
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example>
```

KOMENTAR

Pengenalan Komentor

```
🐳 docker-compose.yml ✕
example > 🐳 docker-compose.yml
1  version: "3.9"
2
   ▶ Run All Services
3  services:
4    # contoh nginx bang
   ▶ Run Service
5    nginx-example:
6      image: nginx:latest
7      container_name: nginx-example
8
9    # kalau yang ini contoh mongodb
   ▶ Run Service
10   mongodb-example:
11     image: mongo:latest
12     container_name: mongodb-example
13
```

PORT

Pengenalan PORT

```
docker-compose.yml X
example > docker-compose.yml
1  version: "3.9"
2
   ▶ Run All Services
3  services:
4    # contoh port dengan long syntax
   ▶ Run Service
5    nginx-port1:
6      image: nginx:latest
7      container_name: nginx-port1
8      ports:
9        - protocol: tcp
10         published: 8080
11         target: 80
12    # contoh port dengan short syntax
   ▶ Run Service
13    nginx-port2:
14      image: nginx:latest
15      container_name: nginx-port2
16      ports:
17        - 8081:80
18
```

Docker Compose: Short Syntax

```
12  # contoh port dengan short syntax
    ▷ Run Service
13  nginx-port2:
14      image: nginx:latest
15      container_name: nginx-port2
16      ports:
17          - 8081:80
18
```

Docker Compose: Long Syntax

```
4 | # contoh port dengan long syntax
5 | > Run Service
6 | nginx-port1:
7 |   image: nginx:latest
8 |   container_name: nginx-port1
9 |   ports:
10 |     - protocol: tcp
11 |       published: 8080
12 |       target: 80
```


Docker Compose: PORT (Menjalankan Container)

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
• PS E:\Semester 5\Parallel and Distributed System\Docker Compose> cd example
• PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose create
time="2025-11-11T00:08:48+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Creating 3/3
  ✓ Network example_default    Created                                0.0s
  ✓ Container nginx-port1      Created                                0.1s
  ✓ Container nginx-port2      Created                                0.1s
• PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> docker compose start
time="2025-11-11T00:08:58+08:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] Running 2/2
  ✓ Container nginx-port1      Started                                0.4s
  ✓ Container nginx-port2      Started                                0.4s
○ PS E:\Semester 5\Parallel and Distributed System\Docker Compose\example> |
```

ENVIRONMENT VARIABLE

Pengenalan ENVIRONMENT VARIABLE

```
docker-compose.yaml X
example > docker-compose.yaml
1  version: "3.9"
2
   Run All Services
3  services:
4    # kalau yang ini contoh mongodb
   Run Service
5    mongodb-example:
6      image: mongo:latest
7      container_name: mongodb-example
8      ports:
9        - 27017:27017
10     environment:
11       MONGO_INITDB_ROOT_USERNAME: alathereon
12       MONGO_INITDB_ROOT_PASSWORD: alathereonn
13       MONGO_INITDB_DATABASE: admin
14
15
```

Docker Compose: ENVIRONMENT VARIABLE

```
PS E:\Semester 5\Parallel and Distributed System\Docker compose\example> docker compose create
time="2025-11-11T00:15:23+00:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] creating 2/2
  ✓ Network example_default    Created                                0.0s
  ✓ Container mongdb-example   Created                                0.1s
PS E:\Semester 5\Parallel and Distributed System\Docker compose\example> docker compose start
time="2025-11-11T00:15:29+00:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
[+] running 1/1
  ✓ Container mongdb-example   Started                                0.2s
PS E:\Semester 5\Parallel and Distributed System\Docker compose\example>
```

```
PS E:\Semester 5\Parallel and Distributed System\Docker compose\example> docker compose exec mongdb-example env
time="2025-11-11T00:16:00+00:00" level=warning msg="E:\Semester 5\Parallel and Distributed System\Docker Compose\example\docker-compose.yml: the attribute 'version' is obsolete, it will be ignored, please remove it to avoid potential confusion"
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=1ff961a73d82
TERM=xterm
MONGO_INITDB_ROOT_PASSWORD=alathereon
MONGO_INITDB_DATABASE=admin
MONGO_INITDB_ROOT_USERNAME=alathereon
OSU_VERSION=1.18
JSWAPL_VERSION=3.13.1
JSWAPL_CHECKSUM=662e323196dd378e91f67578e56a34954bba2e33cat1d78b9f4826af24b941
MONGO_PACKAGE=mongodb-org
MONGO_REPO=repo.mongodb.org
MONGO_ARCH=amd64
MONGO_VERSION=4.0.14
HOME=/data/db
GLIBC_TUNABLES=glibc pthread.cseq=0
```

BIND MOUNT

Pengenalan BIND MOUNT

```
docker-compose.yml X
example > docker-compose.yml
1  version: "3.9"
2
3  # Run All Services
4  services:
5    # kalau yang ini contoh mongodh
6    # Run Service
7    mongodh1:
8      image: mongo:latest
9      container_name: mongodh1
10     ports:
11       - 27017:27017
12     environment:
13       MONGO_INITDB_ROOT_USERNAME: alathereon
14       MONGO_INITDB_ROOT_PASSWORD: alathereonn
15       MONGO_INITDB_DATABASE: admin
16     volumes:
17       - ./data-mongo1:/data/db
18
19   # mongodh2
20   # Run Service
21   mongodh2:
22     image: mongo:latest
23     container_name: mongodh2
24     ports:
25       - 27018:27017
26     environment:
27       MONGO_INITDB_ROOT_USERNAME: alathereon
28       MONGO_INITDB_ROOT_PASSWORD: alathereonn
29       MONGO_INITDB_DATABASE: admin
30     volumes:
31       - type: bind
32         source: ./data-mongodh2
33         target: /data/db
34         read only: false
```

VOLUME

Pengenalan VOLUME

```
docker-compose.yaml X
example > docker-compose.yaml
1  version: "3.9"
2
3  volumes:
4    mongo-data1:
5      name: mongo-data1
6    mongo-data2:
7      name: mongo-data2
8
```

Docker Compose: SHORT SYNTAX VOLUME

```
4  # kalau yang ini contoh mongodb
   D Run Service
5  mongodbi:
6      image: mongo:latest
7      container_name: mongodbi
8      ports:
9          - 27017:27017
10     environment:
11         MONGO_INITDB_ROOT_USERNAME: alathereon
12         MONGO_INITDB_ROOT_PASSWORD: alathereonn
13         MONGO_INITDB_DATABASE: admin
14     volumes:
15         - ./data-mongo1:/data/db
```

Docker Compose: LONG SYNTAX VOLUME

```
17  # mongodb2
    ▷ Run Service
18  mongodb2:
19    image: mongo:latest
20    container_name: mongodb2
21    ports:
22      - 27018:27017
23    environment:
24      MONGO_INITDB_ROOT_USERNAME: alathereon
25      MONGO_INITDB_ROOT_PASSWORD: alathereonn
26      MONGO_INITDB_DATABASE: admin
27    volumes:
28      - type: bind
29        source: ./data-mongodb2
30        target: /data/db
31        read_only: false
32
```



Docker Compose: Menghapus VOLUME

© 2006 The Authors

Pengenalan NETWORK

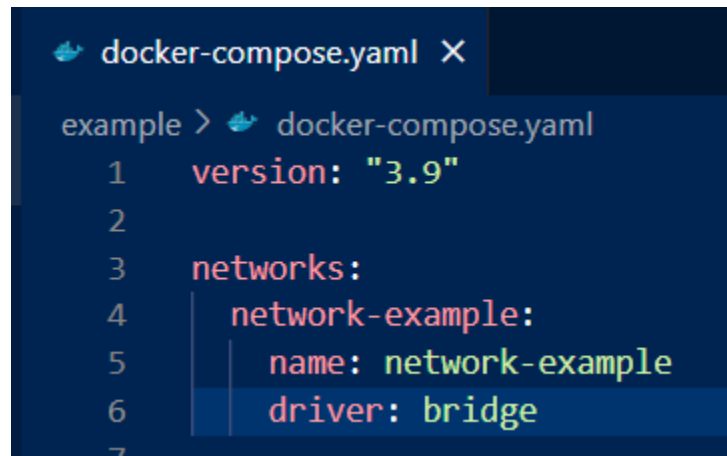
```

PS C:\yosemite> docker inspect --format '{{.NetworkSettings.Networks.mongodh1.IPv4Address}}' $(docker ps -q)
172.17.0.3

```

```
user > Dockerfile > ...
1 FROM golang:1.18-alpine
2
3 RUN mkdir /app
4
5 RUN addgroup -S a206group
6 RUN adduser -S -D -h /app a206user a206group
7 RUN chown -R a206user:a206group /app
8 USER a206user
9
10 COPY main.go /app
11
12 EXPOSE 8080
13 CMD go run /app/main.go
```

Docker Compose: Membuat NETWORK



The screenshot shows a code editor with a file named `docker-compose.yaml`. The content of the file is as follows:

```
example > docker-compose.yaml
1  version: "3.9"
2
3  networks:
4    network-example:
5      name: network-example
6      driver: bridge
7
```

Docker Compose: Menggunakan NETWORK

```
docker-compose.yml X
example > docker-compose.yml
1  version: "3.9"
2
3  networks:
4    network_example:
5      name: network_example
6      driver: bridge
7
8  services:
9    # kalau yang ini contoh mongodb
10   mongodb1:
11     image: mongo:latest
12     container_name: mongodb1
13     ports:
14       - 27017:27017
15     environment:
16       MONGO_INITDB_ROOT_USERNAME: alathereon
17       MONGO_INITDB_ROOT_PASSWORD: alathereonn
18       MONGO_INITDB_DATABASE: admin
19     networks:
20       - network_example
```

```
PS C:\Semester 5\Parallel and Distributed System\docker-compose\examples\docker-inspect-mongodbi
```

```
  "Networks": {  
    "network_comple": {  
      "TPMConfig": null,  
      "Links": null,  
      "Aliases": [  
        "mongodbi",  
        "mongodbi"  
      ],  
      "MacAddress": "ea:f1:a9:8d:f9:3a",  
      "DriverOpts": null,  
      "GwPriority": 0,  
      "NetworkID": "84441968174d0f6a562676171b6ed62e0ecd9428c0bae3edabf1aa9104a5881b",  
      "EndpointID": "d1a89e67d488467c24e047a944a66f82cecd171008808076fe99c96c0bedc1",  
      "Gateway": "172.19.0.1",  
      "IPAddress": "172.19.0.2",  
      "IPPrefixlen": 16,  
      "IPv6Gateway": "",  
      "GlobalIPv6Address": "",  
      "GlobalIPv6Prefixlen": 0,  
      "DNSNames": [  
        "mongodbi",  
        "easthe.net/mas"  
      ]  
    }  
  }  
}
```

DEPENDS ON

Pengenalan DEPENDS ON

```
docker-compose.yml X
example > docker-compose.yml
1  version: "3.9"
2
3  networks:
4    network_example:
5      name: network_example
6      driver: bridge
7
8  services:
9    mongodb-express-example:
10     image: mongo-express-latest
11     container_name: mongodb-express-example
12     depends_on:
13       - mongodb-example
14     ports:
15       - 8081:8081
16     envirnoment:
17       ME_CONFIG_MONGODB_ADMINUSERNAME: alathereon
18       ME_CONFIG_MONGODB_ADMINPASSWORD: alathereonn
19       ME_CONFIG_MONGODB_SERVER: mongodb-example
20     networks:
21       - network_example
```

RESTART

Pengenalan RESTART

docker-compose.yaml X

example > docker-compose.yaml

```
1  version: "3.9"
2
3  networks:
4    network_example:
5      name: network_example
6      driver: bridge
7
8  services:
9    mongodb-express-example:
10     image: mongo-express-latest
11     container_name: mongodb-express-example
12     restart: always
13     depends_on:
14       - mongodb-example
15     ports:
16       - 8081:8081
17     envirnoment:
18       ME_CONFIG_MONGODB_ADMINUSERNAME: alathereon
19       ME_CONFIG_MONGODB_ADMINPASSWORD: alathereonn
20       ME_CONFIG_MONGODB_SERVER: mongodb-example
21     networks:
22       - network_example
```

Docker Compose: Monitor Docker Events

```
PS E:\Semester 5\Parallel and Distributed System\Dockerfile> docker build -t zakariafattawari/health health
[+] Building 6.2s (0/0) FINISHED
-> [internal] load build definition from Dockerfile
-> => transferring dockerfile: 253B
-> [internal] load metadata for docker.io/library/golang:1.18-alpine
-> [internal] load .dockerignore
-> => transferring context: 2B
-> CACHED [1/4] FROM docker.io/library/golang:1.18-alpine@sha256:77f25081bd57e68a518165f3be89c901aec90453fd0f1c5a45601f6cb1528807
-> => resolve docker.io/library/golang:1.18-alpine@sha256:77f25081bd57e68a518165f3be89c901aec90453fd0f1c5a45601f6cb1528807
-> [internal] load build context
-> => transferring context: 531B
-> [2/4] RUN apk --no-cache add curl
-> [3/4] RUN mkdir app
-> [4/4] COPY main.go app
-> exporting to image
-> => exporting layers
-> => exporting manifest sha256:b7de3a8668c2c919d85a65a4fa0c208574a88955e4bbb4eade5e9da651543d8b
-> => exporting config sha256:35ce73ae1c8f1cd7f2c498718a622b456d1654f28fd8551ba254c3e4129b80b4
-> => exporting attestation manifest sha256:7eed8c23bc4f87c88b8884b89bd38b075a4f0baa5881ef1eef5633ded853ddfc
-> => exporting manifest list sha256:1c68c8a75128acbe0e55d54041b84ccc187df7a1ffb61939f131b5e052453d8
-> => naming to docker.io/zakariafattawari/health:latest
-> => unpacking to docker.io/zakariafattawari/health:latest

1 warning found (use docker --debug to expand):
 - JSOWarningsRecommended: JSON arguments recommended for CMD to prevent unintended behavior related to OS signals (line 10)

View build details: docker-desktop://dashboard/build/desktop-linux/desktop-linux/xjkevp0kkgyn8fz8w02k89g
PS E:\Semester 5\Parallel and Distributed System\Dockerfile>
```

RESOURCE LIMIT

Pengenalan RESOURCE LIMIT

```
resource limit > 🐳 docker-compose.yml
1  version: "3.9"
2
3  nginx-example:
4    image: nginx:latest
5    container_name: nginx-example
6    ports:
7      - "8080:80"
8    deploy:
9      resources:
10       reservations:
11         cpus: "0.25"
12         memory: 50M
13       limits:
14         cpus: "0.5"
15         memory: 100M
```


entrypoint >  Dockerfile > ...

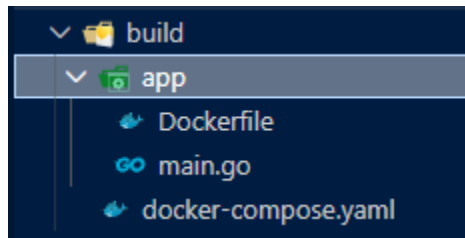
```
1 FROM golang:1.18-alpine
2
3 RUN mkdir /app/
4 COPY main.go /app/
5
6 EXPOSE 8080
7 ENTRYPOINT ["go", "run"]
8 CMD ["/app/main.go"]
```


Docker Compose: CONTAINER STATS

CONTAINER ID	NAME	CPU %	MEM USAGE / LIMIT	MEM %	NET I/O	BLOCK I/O	PIDS
d244126b3dd6	nginx-example	0.00%	17.4MiB / 100MiB	17.40%	1.7kB / 126B	13MB / 12.5kB	17

DOCKERFILE

Membuat Dockerfile



build > app >  main.go

```
1  package main
2
3  import (
4      "fmt"
5      "net/http"
6      "os"
7  )
8
9  func main() {
10     port := os.Getenv("APP_PORT")
11     fmt.Println("Run app in port : " + port)
12     http.HandleFunc("/", HelloServer)
13     http.ListenAndServe(":"+port, nil)
14 }
15
16 func HelloServer(w http.ResponseWriter, r *http.Request) {
17     fmt.Fprintf(w, "Hello, World!")
18 }
```

```
build > app > Dockerfile > FROM
1 FROM golang:1.18-alpine
2
3 ENV APP_PORT=8080
4
5 RUN mkdir app
6 COPY main.go app
7
8 EXPOSE ${APP_PORT}
9
10 CMD go run app/main.go
```

```
build > docker-compose.yml
1 version: "3.9"
2
3 Run All Services
4 services:
5   Run Service
6   app:
7     container_name: app
8     build:
9       context: "./app"
10      dockerfile: Dockerfile
11     image: "app-golang:1.0.0"
12     environment:
13       - "APP_PORT=8080"
14     ports:
15       - "8080:8080"
```

Docker Compose: BUILD DOCKEFILE

```

10. @Parameter(name="/api/submit", description="submit a new report")
11. @PostMapping(consumes = {"application/json"}, produces = {"application/json"})
12. @ResponseBody
13. public void submit(@RequestBody Report report) throws IOException {
14.     // Save the report to the database
15.     reportRepository.save(report);
16.     // Return a 201 status code
17.     return;
18. }
19.
20. @Parameter(name="/api/reports", description="get all reports")
21. @GetMapping
22. @ResponseBody
23. public List<Report> getAllReports() {
24.     return reportRepository.findAll();
25. }
26.
27. @Parameter(name="/api/reports/{id}", description="get report by id")
28. @GetMapping("/{id}")
29. @ResponseBody
30. public Report getReportById(@PathVariable Long id) {
31.     return reportRepository.findById(id).orElse(null);
32. }
33.
34. @Parameter(name="/api/reports/{id}", description="delete report by id")
35. @DeleteMapping("/{id}")
36. @ResponseStatus(HttpStatus.NO_CONTENT)
37. public void deleteReport(@PathVariable Long id) {
38.     reportRepository.deleteById(id);
39. }
40.
41. @Parameter(name="/api/reports/{id}/status", description="update report status")
42. @PutMapping("/{id}/status")
43. @ResponseBody
44. public Report updateReportStatus(@PathVariable Long id, @RequestBody Status status) {
45.     Report report = reportRepository.findById(id).orElse(null);
46.     if (report != null) {
47.         report.setStatus(status);
48.         reportRepository.save(report);
49.     }
50.     return report;
51. }
52.
53. @Parameter(name="/api/reports/{id}/comments", description="add comment to report")
54. @PostMapping("/{id}/comments")
55. @ResponseBody
56. public Report addComment(@PathVariable Long id, @RequestBody Comment comment) {
57.     Report report = reportRepository.findById(id).orElse(null);
58.     if (report != null) {
59.         comment.setReport(report);
60.         commentRepository.save(comment);
61.         report.addComment(comment);
62.         reportRepository.save(report);
63.     }
64.     return report;
65. }
66.
67. @Parameter(name="/api/reports/{id}/comments/{commentId}", description="delete comment from report")
68. @DeleteMapping("/{id}/comments/{commentId}")
69. @ResponseStatus(HttpStatus.NO_CONTENT)
70. public void deleteComment(@PathVariable Long id, @PathVariable Long commentId) {
71.     Report report = reportRepository.findById(id).orElse(null);
72.     if (report != null) {
73.         Comment comment = commentRepository.findById(commentId).orElse(null);
74.         if (comment != null) {
75.             report.removeComment(comment);
76.             commentRepository.deleteById(commentId);
77.             reportRepository.save(report);
78.         }
79.     }
80. }
81.
82. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply", description="reply to comment")
83. @PostMapping("/{id}/comments/{commentId}/reply")
84. @ResponseBody
85. public Report replyToComment(@PathVariable Long id, @PathVariable Long commentId, @RequestBody Reply reply) {
86.     Report report = reportRepository.findById(id).orElse(null);
87.     if (report != null) {
88.         Comment comment = commentRepository.findById(commentId).orElse(null);
89.         if (comment != null) {
90.             reply.setComment(comment);
91.             replyRepository.save(reply);
92.             comment.addReply(reply);
93.             reportRepository.save(report);
94.         }
95.     }
96.     return report;
97. }
98.
99. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}", description="delete reply from comment")
100. @DeleteMapping("/{id}/comments/{commentId}/reply/{replyId}")
101. @ResponseStatus(HttpStatus.NO_CONTENT)
102. public void deleteReply(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId) {
103.     Report report = reportRepository.findById(id).orElse(null);
104.     if (report != null) {
105.         Comment comment = commentRepository.findById(commentId).orElse(null);
106.         if (comment != null) {
107.             Reply reply = replyRepository.findById(replyId).orElse(null);
108.             if (reply != null) {
109.                 comment.removeReply(reply);
110.                 replyRepository.deleteById(replyId);
111.                 reportRepository.save(report);
112.             }
113.         }
114.     }
115. }
116.
117. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status", description="update reply status")
118. @PutMapping("/{id}/comments/{commentId}/reply/{replyId}/status")
119. @ResponseBody
120. public Report updateReplyStatus(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @RequestBody Status status) {
121.     Report report = reportRepository.findById(id).orElse(null);
122.     if (report != null) {
123.         Comment comment = commentRepository.findById(commentId).orElse(null);
124.         if (comment != null) {
125.             Reply reply = replyRepository.findById(replyId).orElse(null);
126.             if (reply != null) {
127.                 reply.setStatus(status);
128.                 replyRepository.save(reply);
129.                 reportRepository.save(report);
130.             }
131.         }
132.     }
133.     return report;
134. }
135.
136. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}", description="delete reply status")
137. @DeleteMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}")
138. @ResponseStatus(HttpStatus.NO_CONTENT)
139. public void deleteReplyStatus(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId) {
140.     Report report = reportRepository.findById(id).orElse(null);
141.     if (report != null) {
142.         Comment comment = commentRepository.findById(commentId).orElse(null);
143.         if (comment != null) {
144.             Reply reply = replyRepository.findById(replyId).orElse(null);
145.             if (reply != null) {
146.                 Status status = statusRepository.findById(statusId).orElse(null);
147.                 if (status != null) {
148.                     reply.removeStatus(status);
149.                     statusRepository.deleteById(statusId);
150.                     replyRepository.save(reply);
151.                     reportRepository.save(report);
152.                 }
153.             }
154.         }
155.     }
156. }
157.
158. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment", description="add comment to reply")
159. @PostMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment")
160. @ResponseBody
161. public Report addCommentToReply(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId, @RequestBody Comment comment) {
162.     Report report = reportRepository.findById(id).orElse(null);
163.     if (report != null) {
164.         Comment comment1 = commentRepository.findById(commentId).orElse(null);
165.         if (comment1 != null) {
166.             Reply reply = replyRepository.findById(replyId).orElse(null);
167.             if (reply != null) {
168.                 Status status = statusRepository.findById(statusId).orElse(null);
169.                 if (status != null) {
170.                     comment.setReply(reply);
171.                     commentRepository.save(comment);
172.                     reportRepository.save(report);
173.                 }
174.             }
175.         }
176.     }
177.     return report;
178. }
179.
180. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}", description="delete comment from reply")
181. @DeleteMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}")
182. @ResponseStatus(HttpStatus.NO_CONTENT)
183. public void deleteCommentFromReply(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId, @PathVariable Long commentId1) {
184.     Report report = reportRepository.findById(id).orElse(null);
185.     if (report != null) {
186.         Comment comment1 = commentRepository.findById(commentId).orElse(null);
187.         if (comment1 != null) {
188.             Reply reply = replyRepository.findById(replyId).orElse(null);
189.             if (reply != null) {
190.                 Status status = statusRepository.findById(statusId).orElse(null);
191.                 if (status != null) {
192.                     Comment comment = commentRepository.findById(commentId1).orElse(null);
193.                     if (comment != null) {
194.                         commentRepository.deleteById(commentId1);
195.                         reportRepository.save(report);
196.                     }
197.                 }
198.             }
199.         }
200.     }
201. }
202.
203. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply", description="reply to comment from reply")
204. @PostMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply")
205. @ResponseBody
206. public Report replyToCommentFromReply(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId, @PathVariable Long commentId1, @RequestBody Reply reply) {
207.     Report report = reportRepository.findById(id).orElse(null);
208.     if (report != null) {
209.         Comment comment1 = commentRepository.findById(commentId).orElse(null);
210.         if (comment1 != null) {
211.             Reply reply1 = replyRepository.findById(replyId).orElse(null);
212.             if (reply1 != null) {
213.                 Status status = statusRepository.findById(statusId).orElse(null);
214.                 if (status != null) {
215.                     Comment comment = commentRepository.findById(commentId1).orElse(null);
216.                     if (comment != null) {
217.                         reply.setComment(comment);
218.                         replyRepository.save(reply);
219.                         reportRepository.save(report);
220.                     }
221.                 }
222.             }
223.         }
224.     }
225.     return report;
226. }
227.
228. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply/{replyId}", description="delete reply from comment from reply")
229. @DeleteMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply/{replyId}")
230. @ResponseStatus(HttpStatus.NO_CONTENT)
231. public void deleteReplyFromCommentFromReply(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId, @PathVariable Long commentId1, @PathVariable Long replyId1) {
232.     Report report = reportRepository.findById(id).orElse(null);
233.     if (report != null) {
234.         Comment comment1 = commentRepository.findById(commentId).orElse(null);
235.         if (comment1 != null) {
236.             Reply reply1 = replyRepository.findById(replyId).orElse(null);
237.             if (reply1 != null) {
238.                 Status status = statusRepository.findById(statusId).orElse(null);
239.                 if (status != null) {
240.                     Comment comment = commentRepository.findById(commentId1).orElse(null);
241.                     if (comment != null) {
242.                         Reply reply2 = replyRepository.findById(replyId1).orElse(null);
243.                         if (reply2 != null) {
244.                             commentRepository.deleteById(commentId1);
245.                             replyRepository.deleteById(replyId1);
246.                             reportRepository.save(report);
247.                         }
248.                     }
249.                 }
250.             }
251.         }
252.     }
253. }
254.
255. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply/{replyId}/comment", description="add comment to reply from comment")
256. @PostMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply/{replyId}/comment")
257. @ResponseBody
258. public Report addCommentToReplyFromComment(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId, @PathVariable Long commentId1, @PathVariable Long replyId1, @RequestBody Comment comment) {
259.     Report report = reportRepository.findById(id).orElse(null);
260.     if (report != null) {
261.         Comment comment1 = commentRepository.findById(commentId).orElse(null);
262.         if (comment1 != null) {
263.             Reply reply1 = replyRepository.findById(replyId).orElse(null);
264.             if (reply1 != null) {
265.                 Status status = statusRepository.findById(statusId).orElse(null);
266.                 if (status != null) {
267.                     Comment comment2 = commentRepository.findById(commentId1).orElse(null);
268.                     if (comment2 != null) {
269.                         Reply reply2 = replyRepository.findById(replyId1).orElse(null);
270.                         if (reply2 != null) {
271.                             comment.setReply(reply2);
272.                             commentRepository.save(comment);
273.                             reportRepository.save(report);
274.                         }
275.                     }
276.                 }
277.             }
278.         }
279.     }
280.     return report;
281. }
282.
283. @Parameter(name="/api/reports/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply/{replyId}/comment/{commentId}", description="delete comment from reply from comment")
284. @DeleteMapping("/{id}/comments/{commentId}/reply/{replyId}/status/{statusId}/comment/{commentId}/reply/{replyId}/comment/{commentId}")
285. @ResponseStatus(HttpStatus.NO_CONTENT)
286. public void deleteCommentFromReplyFromComment(@PathVariable Long id, @PathVariable Long commentId, @PathVariable Long replyId, @PathVariable Long statusId, @PathVariable Long commentId1, @PathVariable Long replyId1, @PathVariable Long commentId2) {
287.     Report report = reportRepository.findById(id).orElse(null);
288.     if (report != null) {
289.         Comment comment1 = commentRepository.findById(commentId).orElse(null);
290.         if (comment1 != null) {
291.             Reply reply1 = replyRepository.findById(replyId).orElse(null);
292.             if (reply1 != null) {
293.                 Status status = statusRepository.findById(statusId).orElse(null);
294.                 if (status != null) {
295.                     Comment comment2 = commentRepository.findById(commentId1).orElse(null);
296.                     if (comment2 != null) {
297.                         Reply reply2 = replyRepository.findById(replyId1).orElse(null);
298.                         if (reply2 != null) {
299
```

Docker Compose: Menghapus Image

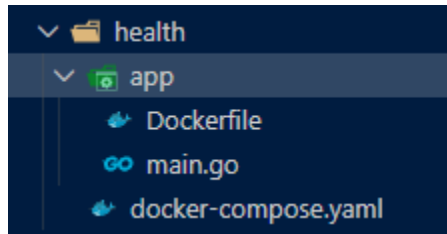
```
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\build> docker image rm app-golang:1.0.0
Untagged: app-golang:1.0.0
Deleted: sha256:4dc623d00af886b4d1a85e110dde3d58188bb91860a328c745474ba5e43783fdc
PS E:\Semester 5\Parallel and Distributed System\Docker Compose\build> █
```

Docker Compose: BUILD Ulang

[illegible]

HEALTHCHECK

Pengenalan HEALTHCHECK



```
health > 🐳 docker-compose.yaml
```

```
1  version: "3.9"
```

```
2
```

```
   ▶ Run All Services
```

```
3  services:
```

```
   ▶ Run Service
```

```
4  app:
```

```
   container_name: app
```

```
   build:
```

```
     context: "./app"
```

```
     dockerfile: Dockerfile
```

```
   image: "app-golang:1.0.0"
```

```
   environment:
```

```
     - "APP_PORT=8080"
```

```
   ports:
```

```
     - "8080:8080"
```

```
   healthcheck:
```

```
     test: ["CMD", "curl", "-f", "http://localhost:8080/health"]
```

```
     interval: 5s
```

```
     timeout: 5s
```

```
     retries: 3
```

```
     start_period: 3s
```

```
health > app > 🐳 Dockerfile > ...
```

```
1  FROM golang:1.18-alpine
```

```
2
```

```
3  RUN apk --no-cache add curl
```

```
4  RUN mkdir app
```

```
5
```

```
6  COPY main.go app
```

```
7
```

```
8  EXPOSE 8080
```

```
9
```

```
10 CMD go run app/main.go
```

```
health > app > main.go
1  package main
2
3  import (
4      "fmt"
5      "net/http"
6  )
7
8  var counter = 0
9
10 func main() {
11     http.HandleFunc("/", HelloServer)
12     http.HandleFunc("/health", HealthCheck)
13
14     http.ListenAndServe(":8080", nil)
15 }
16
17 func HealthCheck(w http.ResponseWriter, r *http.Request) {
18     counter = counter + 1
19     if counter > 5 {
20         w.WriteHeader(500)
21         fmt.Fprintf(w, "KO")
22     } else {
23         fmt.Fprintf(w, "OK")
24     }
25 }
26
27 func HelloServer(w http.ResponseWriter, r *http.Request) {
28     fmt.Fprintf(w, "Hello, World!")
29 }
```

Docker Compose: HEALTHCHECK

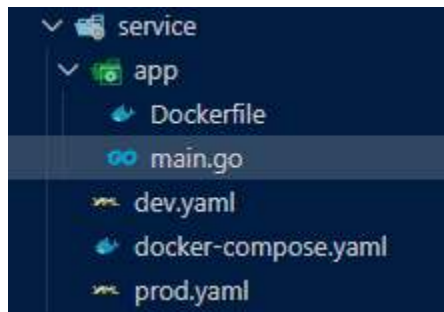
```
PS E:\Workspace\S\parallel and Distributed System\docker-compose> cd health
PS E:\Workspace\S\parallel and Distributed System\docker-compose\health> docker compose up --build -d
time="2025-11-16T22:11:34.881860" level=warning msg="E:\Workspace\S\parallel and Distributed System\docker-compose\health\docker-compose.yml: the attribute 'name' is to avoid potential confusion"
[+] Building 0.9s (1/14) FINISHED
-> [internal] load local build definitions
-> > reading from cache 3800
-> [internal] load build definition from Dockerfile
-> > transferring dockerfile: 350B
-> [internal] load metadata for docker.io/library/golang:1.18-alpine
-> [internal] load .dockerignore
-> > transferring context: 2B
-> [1/4] FROM docker.io/library/golang:1.18-alpine@sha256:77f228011c57a99ad3d166f21e2a583aea594c3f49f111e4e603f3d12c887
-> => resolve docker.io/library/golang:1.18-alpine@sha256:77f228011c57a99ad3d166f21e2a583aea594c3f49f111e4e603f3d12c887
-> [internal] load build context
-> => transferring context: 22B
=> CACHED [2/4] RUN apk --no-cache add curl
=> CACHED [3/4] RUN apk add go
=> CACHED [4/4] COPY main.go app
=> exporting to image
=> => exporting layers
=> => exporting manifest layer: sha256:fe2e6a9a5e1d701e4dd3eaa5f6e5a3f97f44903498d0a5e5a30a72a
=> => exporting config: sha256:8d437b03a237c0ba7e0d5530960304926687a913249e1543c0ab030e
=> => exporting annotations manifest: sha256:5804e91b7d457b2042d616384518173836a2d493126684992d9191a0a3d17
-> => exporting manifest layer: sha256:963d946815a02d889a71a074b07112901e759960e8d1d049339677011d
-> => naming to docker.io/library/app-golang:1.8.0
-> => unpadding to docker.io/library/app-golang:1.0.0
-> naming processname for metadata file
[+] Building 2/2
✓ app-golang:1.0.0 Built
✓ Container app Started
PS E:\Workspace\S\parallel and Distributed System\docker-compose\health> docker compose ps
time="2025-11-16T22:11:36.031900" level=warning msg="E:\Workspace\S\parallel and Distributed System\docker-compose\health\docker-compose.yml: the attribute 'name' is to avoid potential confusion"
NAME                IMAGE                STATUS                CREATED              STARTED              UPTIME
app                 app-golang:1.0.0    "Run /bin /c 'go run ..."  4 seconds ago        Up 2 seconds (healthy: starting)  0.0.0.0:8080->8080/tcp, [::]:8080->8080/tcp
PS E:\Workspace\S\parallel and Distributed System\docker-compose\health> docker compose ps
time="2025-11-16T22:11:58.081860" level=warning msg="E:\Workspace\S\parallel and Distributed System\docker-compose\health\docker-compose.yml: the attribute 'name' is to avoid potential confusion"
NAME                IMAGE                STATUS                CREATED              STARTED              UPTIME
app                 app-golang:1.0.0    "Run /bin /c 'go run ..."  20 seconds ago      Up 15 seconds (healthy)  0.0.0.0:8080->8080/tcp, [::]:8080->8080/tcp
```

Docker Compose: Disable HEALTHCHECK

```
health > 🐳 docker-compose.yaml
1  version: "3.9"
2
   ↳ Run All Services
3  services:
   ↳ Run Service
4    app:
5      container_name: app
6      build:
7        context: "./app"
8        dockerfile: Dockerfile
9      image: "app-golang:1.0.0"
10     environment:
11       - "APP_PORT=8080"
12     ports:
13       - "8080:8080"
14     healthcheck:
15       test: ["CMD", "curl", "-f", "http://localhost:8080/health"]
16       interval: 5s
17       timeout: 5s
18       retries: 3
19       start_period: 3s
20       disable: true
```

EXTEND SERVICE

Pengenalan EXTEND SERVICE



```
service > docker-compose.yaml
1  version: "3.9"
2
3  services:
4    app:
5      container_name: app
6      build:
7        context: "./app"
8        dockerfile: Dockerfile
9      image: "app-golang:1.0.0"
10     environment:
11       - "APP_PORT=8080"
12       - "MODE=local"
13     ports:
14       - "8080:8080"
```

```
service > dev.yaml
1  version: "3.9"
2
3  services:
4    app:
5      environment:
6        - "MODE=dev"
```

```
service > prod.yaml
1  version: "3.9"
2
3  services:
4    app:
5      environment:
6        - "MODE=prod"
```

```
service > app > Dockerfile > ...
1 FROM golang:1.18-alpine
2
3 ENV APP_PORT=8080
4 ENV MODE=local
5
6 RUN mkdir app
7 COPY main.go app
8
9 EXPOSE ${APP_PORT}
10
11 CMD go run app/main.go
```

```
service > app > main.go
1 package main
2
3 import (
4     "fmt"
5     "net/http"
6     "os"
7 )
8
9 func main() {
10     port := os.Getenv("APP_PORT")
11     fmt.Println("Run app in port : " + port)
12     http.HandleFunc("/", HelloServer)
13     http.ListenAndServe(":"+port, nil)
14 }
15
16 func HelloServer(w http.ResponseWriter, r *http.Request) {
17     mode := os.Getenv("MODE")
18     response := "Hello " + mode
19     fmt.Fprintf(w, response)
20 }
```

Docker Compose: EXTEND SERVICE

[illegible]