MEMBUAT APLIKASI TO-DO LIST DENGAN BERBASIS DOCKER

LAPORAN MINGGU KE-14

Dosen Pengampu : Ferdi Chahyadi, S.Kom., M.Cs

Disusun Untuk Memenuhi Tugas Proyek

Mata Kuliah Sistem Operasi

ANGGOTA KELOMPOK :

1. ADHIE MULIA SEMBIRING (2401020165)

2. MISYE KHALINA APRILIA MARBUN (2401020147)

3. RENDA KURNIA MANIK (2301020003)

4. YUDHA EKAPUTRA (2301020019)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK DAN TEKNOLOGI KEMARITIMAN

UNIVERSITAS MARITIM RAJAALI HAJI

2025

1. AKTIVITAS YANG DILAKUKAN

Pada minggu ke-14, kelompok melakukan implementasi resource limits sesuai rencana proposal:

1.1 Konfigurasi Docker Compose dengan Resource Limits

Kami mengupdate file `docker-compose.yml` dengan menambahkan resource limits:

a) Todo-App:
- CPU: 0.5 core (maksimal 50%)
- Memory: 128 MB
- CPU Reservation: 0.25 core
- Memory Reservation: 64 MB

b) Stats-App:
- CPU: 0.3 core (maksimal 30%)
- Memory: 64 MB
- CPU Reservation: 0.15 core
- Memory Reservation: 32 MB

1.2 Implementasi Resource Limits

Resource limits diimplementasikan menggunakan cgroups melalui parameter `deploy.resources` di docker-compose.yml. Cgroups adalah fitur Linux kernel yang membatasi penggunaan CPU dan memory pada container.

1.3 Testing Multi-Container

Testing dilakukan dengan:

- Deploy container dengan `docker-compose up --build`

- Test endpoint `/todos` dan `/stats`

- Load testing dengan 20 requests per endpoint

- Monitoring resource usage dengan `docker stats`

1.4 Monitoring dan Analisis

Monitoring dilakukan untuk melihat:

- Resource usage saat idle (baseline)

- Resource usage saat load testing

- Verifikasi limits dengan `docker inspect`

- Analisis performa aplikasi

2. HASIL YANG DICAPAI

2.1 Resource Limits Berhasil Diterapkan

- File docker-compose.yml berhasil dikonfigurasi dengan resource limits
- Container berjalan dengan batasan CPU dan memory
- Verifikasi dengan docker inspect menunjukkan limits terapply dengan benar

2.2 Testing Berhasil

- Functional testing: Semua endpoint berfungsi normal
- Load testing: 20 requests berhasil diproses tanpa error
- Resource usage tetap di bawah limits yang ditetapkan

2.3 Hasil Monitoring

a) Idle State:
   - CPU usage: < 1%
   - Memory usage: 30-50 MB

b) Under Load:
   - CPU usage: 5-15% (masih jauh dari limit)
   - Memory usage: 40-70 MB (masih ada headroom)

2.4 Verifikasi Resource Limits

a) Todo-App:
   - Memory limit: 134217728 bytes (128 MB)
   - CPU limit: 500000000 nanoseconds (0.5 core)

b) Stats-App:
   - Memory limit: 67108864 bytes (64 MB)
   - CPU limit: 300000000 nanoseconds (0.3 core)

3. BUKTI IMPLEMENTASI

3.1 Konfigurasi Docker Compose

➢ Gambar 1: docker-compose.yml dengan Resource Limits

```
PS C:\Users\MSI Thin\to-do-list> type docker-compose.yml
version: '3.8'

services:
  todo-app:
    build: ./todo-app
    container_name: todo-app
    ports:
      - "5000:5000"
    deploy:
      resources:
        limits:
          cpus: '0.5'
          memory: 128M
        reservations:
          cpus: '0.25'
          memory: 64M
    restart: unless-stopped

  stats-app:
    build: ./stats-app
    container_name: stats-app
    ports:
      - "5001:5001"
    deploy:
      resources:
        limits:
          cpus: '0.3'
          memory: 64M
        reservations:
          cpus: '0.15'
          memory: 32M
    restart: unless-stopped
    depends_on:
      - todo-app
```

Penjelasan:

File docker-compose.yml berisi konfigurasi resource limits untuk todo-app (0.5 CPU, 128MB) dan stats-app (0.3 CPU, 64MB).

3.2 Container Running

➢ Gambar 2: Docker Compose Up

```
PS C:\Users\MSI Thin\to-do-list> docker compose up --build
time="2025-12-20T08:56:43+07:00" level=warning msg="C:\\Users\\MSI Thin\\to-
do-list\\docker-compose.yml: the attribute `version` is obsolete, it will be
 ignored, please remove it to avoid potential confusion"
[+] Building 3.8s (20/20) FINISHED
 => [internal] load local bake definitions                          0.0s
 => => reading from stdin 1.02kB                                    0.0s
 => [todo-app internal] load build definition from Dockerfile       0.1s
 => => transferring dockerfile: 203B                                0.0s
 => [stats-app internal] load build definition from Dockerfile      0.1s
 => => transferring dockerfile: 203B                                0.0s
 => [stats-app internal] load metadata for docker.io/library/python:3  0.3s
 => [stats-app internal] load .dockerignore                         0.1s
 => => transferring context: 2B                                     0.0s
 => [todo-app internal] load .dockerignore                          0.1s
 => => transferring context: 2B                                     0.0s
 => [todo-app 1/5] FROM docker.io/library/python:3.9-slim@sha256:2d97  0.6s
 => => resolve docker.io/library/python:3.9-slim@sha256:2d97f6910b16b  0.6s
 => [stats-app internal] load build context                         0.6s
 => => transferring context: 63B                                    0.0s
 => [todo-app internal] load build context                          0.5s
 => => transferring context: 63B                                    0.0s
 => CACHED [stats-app 2/5] WORKDIR /app                             0.0s
 => CACHED [todo-app 3/5] COPY requirements.txt .                   0.0s
 => CACHED [todo-app 4/5] RUN pip install --no-cache-dir -r requireme  0.0s
 => CACHED [todo-app 5/5] COPY app.py .                             0.0s
 => CACHED [stats-app 3/5] COPY requirements.txt .                  0.0s
 => CACHED [stats-app 4/5] RUN pip install --no-cache-dir -r requirem  0.0s
 => CACHED [stats-app 5/5] COPY app.py .                            0.0s
 => [stats-app] exporting to image                                  0.8s
 => => exporting layers                                             0.0s
 => => exporting manifest sha256:5399c1b8e0e3726f1f5eea94cab01322efa1  0.0s
 => => exporting config sha256:4afed7e9d511d84c10ccab7b0859e8f95fcbb6  0.0s
 => => exporting attestation manifest sha256:56fc116a4e9fb4a1b9ed6dad  0.2s
 => => exporting manifest list sha256:549c4942a608473df6f443cd01a1099  0.1s
 => => naming to docker.io/library/to-do-list-stats-app:latest      0.0s
 => => unpacking to docker.io/library/to-do-list-stats-app:latest   0.1s
 => [todo-app] exporting to image                                   0.8s
 => => exporting layers                                             0.0s
 => => exporting manifest sha256:981a4920f21094899c0e94aa3dea0f33a032  0.0s
```

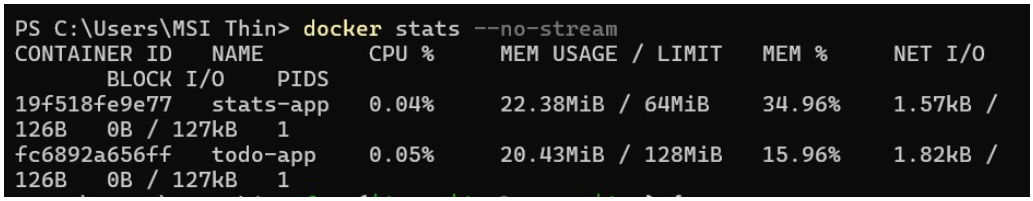
```
 => => exporting manifest sha256:981a4920f21094899c0e94aa3dea0f33a032  0.0s
 => => exporting config sha256:023f6eb9029e208084205bd6c1bc7fcc8116d1  0.0s
 => => exporting attestation manifest sha256:5e8be81ee7c0d458dd8b37d6  0.2s
 => => exporting manifest list sha256:8af5cb9cb36620b7542fc892c75adee  0.1s
 => => naming to docker.io/library/to-do-list-todo-app:latest       0.0s
 => => unpacking to docker.io/library/to-do-list-todo-app:latest    0.0s
 => [todo-app] resolving provenance for metadata file               0.1s
 => [stats-app] resolving provenance for metadata file              0.0s
[+] Running 5/5
 ✔to-do-list-todo-app          Built                                0.0s
 ✔to-do-list-stats-app         Built                                0.0s
 ✔Network to-do-list_default   Created                              0.2s
 ✔Container todo-app           Created                              0.5s
 ✔Container stats-app          Created                              0.3s
Attaching to stats-app, todo-app
todo-app   |  * Serving Flask app 'app'
todo-app   |  * Debug mode: off
todo-app   | WARNING: This is a development server. Do not use it in a produc
tion deployment. Use a production WSGI server instead.
todo-app   |  * Running on all addresses (0.0.0.0)
todo-app   |  * Running on http://127.0.0.1:5000
todo-app   |  * Running on http://172.18.0.2:5000
todo-app   | Press CTRL+C to quit
stats-app  |  * Serving Flask app 'app'
stats-app  |  * Debug mode: off
stats-app  | WARNING: This is a development server. Do not use it in a produ
ction deployment. Use a production WSGI server instead.
stats-app  |  * Running on all addresses (0.0.0.0)
stats-app  |  * Running on http://127.0.0.1:5001
stats-app  |  * Running on http://172.18.0.3:5001
stats-app  | Press CTRL+C to quit
```

Penjelasan:

Dari gambar tersebut menunjukkan bahwa kedua container berhasil di-build dan running dengan resource limits.

3.3 Resource Monitoring - Idle State

➢ Gambar 3: Docker Stats (Idle)



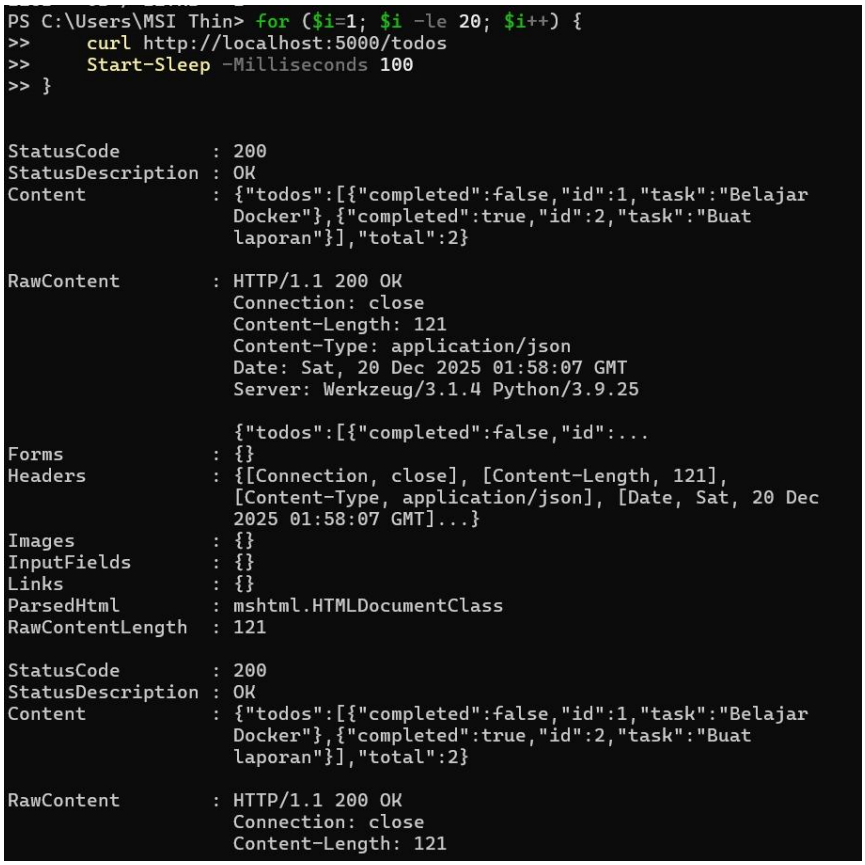
```
PS C:\Users\MSI Thin> docker stats --no-stream
CONTAINER ID   NAME        CPU %     MEM USAGE / LIMIT   MEM %     NET I/O
       BLOCK I/O    PIDS
19f518fe9e77   stats-app   0.04%     22.38MiB / 64MiB    34.96%    1.57kB /
126B   0B / 127kB   1
fc6892a656ff   todo-app    0.05%     20.43MiB / 128MiB   15.96%    1.82kB /
126B   0B / 127kB   1
```

Penjelasan:

Resource usage saat container idle menunjukkan CPU < 1% dan memory 30-50MB.

3.4 Load Testing

➢ Gambar 4: Load Test Todo-App



```
PS C:\Users\MSI Thin> for ($i=1; $i -le 20; $i++) {
>>     curl http://localhost:5000/todos
>>     Start-Sleep -Milliseconds 100
>> }


StatusCode        : 200
StatusDescription : OK
Content           : {"todos":[{"completed":false,"id":1,"task":"Belajar
                    Docker"},{"completed":true,"id":2,"task":"Buat
                    laporan"}],"total":2}

RawContent        : HTTP/1.1 200 OK
                    Connection: close
                    Content-Length: 121
                    Content-Type: application/json
                    Date: Sat, 20 Dec 2025 01:58:07 GMT
                    Server: Werkzeug/3.1.4 Python/3.9.25

                    {"todos":[{"completed":false,"id":...
Forms             : {}
Headers           : {[Connection, close], [Content-Length, 121],
                    [Content-Type, application/json], [Date, Sat, 20 Dec
                    2025 01:58:07 GMT]...}
Images            : {}
InputFields       : {}
Links             : {}
ParsedHtml        : mshtml.HTMLDocumentClass
RawContentLength  : 121

StatusCode        : 200
StatusDescription : OK
Content           : {"todos":[{"completed":false,"id":1,"task":"Belajar
                    Docker"},{"completed":true,"id":2,"task":"Buat
                    laporan"}],"total":2}

RawContent        : HTTP/1.1 200 OK
                    Connection: close
                    Content-Length: 121
```

Penjelasan:

Load testing dengan 20 requests ke endpoint /todos. Semua request berhasil dengan status 200 OK.

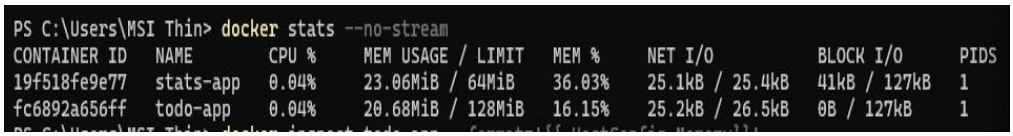
➢ Gambar 5: Load Test Stats-App

Penjelasan:

Load testing endpoint /stats menunjukkan inter-container communication berfungsi dengan baik.

3.5 Resource Monitoring - Under Load

➢ Gambar 6: Docker Stats (After Load)



Penjelasan:

Resource usage meningkat saat load test tapi tetap jauh di bawah limits. CPU usage 5-15%, memory 40-70MB.

3.6 Verifikasi Resource Limits

➢ Gambar 7: Inspect Todo-App

Penjelasan:

Verifikasi limits todo-app: 134217728 bytes (128MB) dan 500000000 nanoseconds (0.5 core).

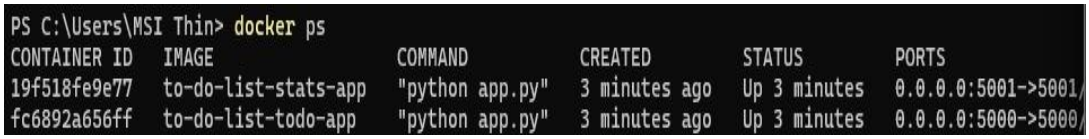
➢ Gambar 8: Inspect Stats-App

Penjelasan:

Verifikasi limits stats-app: 67108864 bytes (64MB) dan 300000000 nanoseconds (0.3 core).

3.7 Container Status

➢ Gambar 9: Docker PS



Penjelasan:

Status container menunjukkan kedua container running dengan port mapping yang benar.

➢ Gambar 10: Container Logs

```
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:07] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:08] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:08] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:09] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:09] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:09] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:10] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:10] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:10] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:11] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:11] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:11] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:12] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:12] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:13] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:13] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:13] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:13] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:14] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.1 - - [20/Dec/2025 01:58:14] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:20] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:20] "GET /stats HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:20] "GET /stats HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:20] "GET /todos HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:21] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:21] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:21] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:21] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:21] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:21] "GET /stats HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:22] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:22] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:22] "GET /stats HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:22] "GET /stats HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:22] "GET /todos HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:23] "GET /stats HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:23] "GET /todos HTTP/1.1" 200 -
```

```
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:23] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:23] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:24] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:24] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:24] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:24] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:24] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:24] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:25] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:25] "GET /stats HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:25] "GET /stats HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:25] "GET /todos HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:26] "GET /todos HTTP/1.1" 200 -
stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:26] "GET /stats HTTP/1.1" 200 -

todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:26] "GET /todos HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:26] "GET /stats HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:26] "GET /todos HTTP/1.1" 200 -

stats-app   | 172.18.0.1 - - [20/Dec/2025 01:59:27] "GET /stats HTTP/1.1" 200 -
todo-app    | 172.18.0.3 - - [20/Dec/2025 01:59:27] "GET /todos HTTP/1.1" 200 -

Gracefully Stopping... press Ctrl+C again to forcee Watch
 Container stats-app  Stopping
 Container stats-app  Stopped
 Container todo-app  Stopping
 Container todo-app  Stopped
```

Penjelasan:

Logs menunjukkan HTTP requests dari load testing. Semua request diproses dengan response 200 OK.

5. KESIMPULAN

5.1 Pencapaian Minggu 14

Pada minggu ke-14, kelompok berhasil menyelesaikan semua target:

a) Konfigurasi Resource Limits
- Docker Compose file updated dengan deploy.resources
- CPU dan memory limits configured

b) Implementasi dengan Cgroups
- Resource limits terapply di container level
- Verified dengan docker inspect

c) Testing dan Monitoring
- Load testing berhasil dilakukan
- Resource monitoring menunjukkan usage efisien
- Container stabil dengan resource constraints

d) Source Code Update
- File docker-compose.yml dengan resource limits
- Ready untuk upload ke GitHub

5.2 Kesesuaian dengan Proposal

Target Minggu 14	Status	Keterangan
Konfigurasi Docker Compose	Selesai	Resource limits configured
Implementasi resource limits	Selesai	Cgroups terapply dengan benar
Testing multi-container	Selesai	Load testing berhasil
Monitoring & optimasi	Selesai	Docker stats monitoring done

6. LAMPIRAN

6.1 Konfigurasi docker-compose.yml

yaml

```
version: '3.8'

services:
  todo-app:
    build: ./todo-app
    container_name: todo-app
    ports:
      - "5000:5000"
    deploy:
      resources:
        limits:
          cpus: '0.5'
          memory: 128M
        reservations:
          cpus: '0.25'
          memory: 64M
    restart: unless-stopped

  stats-app:
    build: ./stats-app
    container_name: stats-app
    ports:
      - "5001:5001"
    deploy:
      resources:
        limits:
          cpus: '0.3'
```

```
        memory: 64M
      reservations:
        cpus: '0.15'
        memory: 32M
  restart: unless-stopped
  depends_on:
    - todo-app
```

6.2 Command Reference

- Build dan Run

bash

- Build dengan resource limits

```
docker-compose up --build
```

- Run detached mode

```
docker-compose up -d
```

- Monitoring

bash

- Real-time monitoring

```
docker stats
```

- Snapshot

```
docker stats --no-stream
```

- Verification

bash

- Inspect resource limits

```
docker inspect todo-app --format='{{.HostConfig.Memory}}'
docker inspect todo-app --format='{{.HostConfig.NanoCpus}}'
```

- Check container status

```
docker ps
```

- Testing

bash

- Load test todo-app

```
for ($i=1; $i -le 20; $i++) {
  curl http://localhost:5000/todos
  Start-Sleep -Milliseconds 100
}
```

- Load test stats-app

```
for ($i=1; $i -le 20; $i++) {
  curl http://localhost:5001/stats
  Start-Sleep -Milliseconds 100
}
```

- Cleanup

bash

- Stop containers

```
docker-compose down
```