

CAK3EAB3 KOMPUTASI AWAN DAN TERDISTRIBUSI

Tugas-15

Studi Kasus

Kelompok: 01

KELAS IF-46-11

Dosen: ISB



PROGRAM STUDI **INFORMATIKA**

FAKULTAS INFORMATIKA

UNIVERSITAS TELKOM

BANDUNG

2024





Ketentuan:

1. Dikerjakan secara kelompok.
2. Dikerjakan langsung pada lembar soal.
3. Untuk soal perhitungan, maka cara atau langkah pengerjaan **wajib** dituliskan.
4. Setiap anggota kelompok wajib mengerjakan soal (ikut berkontribusi).
5. Anggota kelompok yang tidak ikut mengerjakan maka nilainya nol, nama mahasiswa tersebut tidak perlu ditulis di tabel.
6. Unggah jawaban ke LMS diwakili oleh salah satu anggota kelompok.
7. Lengkapi tabel pernyataan di bawah ini sebagai syarat nilai diinput.

PERNYATAAN

Saya menyatakan bahwa saya:

- (a) benar-benar ikut mengerjakan soal
- (b) telah menjelaskan jawaban saya ke teman anggota kelompok
- (c) telah memahami penjelasan dari teman anggota kelompok
- (d) tidak mencontek jawaban dari kelompok lain
- (e) tidak memberikan jawaban PR ini ke kelompok lain

Nama	NIM	Kontribusi Tugas	Tanda Tangan
Sigit Hadi Putranto	1301223148	1	
Falah Akbar	1301220080	1	
Antonius Vincent Jung	1301223436	1	
Satria Aji Darmawan	1301223220	1	

TUGAS 15

1. Buatlah website/aplikasi sederhana yang dapat melakukan scaling pada saat terdapat banyak permintaan dari user
 - a. Desainlah arsitektur yang akan Anda Gunakan! Sistem harus scalable, reliable dan aman.
 - b. Implementasikan desain Anda!
 - c. Uji hasil implementasi Anda!

Penyelesaian

Desain Arsitektur

Kami memakai [Heroku](#) untuk membuat aplikasi sederhana, berikut alur desain nya :

Backend:

- Menggunakan index.js untuk backend yang dapat menangani request dengan baik.
- Menerapkan API endpoint yang sederhana, seperti GET/Data.

Database:

- Menggunakan PostgreSQL (di Heroku) sebagai database.

Scaling dan Reliability:

- Menggunakan fitur Heroku Dynos untuk autoscaling backend.

Security:

- Menggunakan HTTPS (secara default di Heroku).

Logging dan Monitoring:

- Menggunakan Heroku Metrics untuk memantau performa aplikasi.

Implementasi

PERSIAPAN

Login ke Heroku

masuk ke heroku CLI menggunakan akun yang sudah terdaftar pada tugas 12 lalu.

```
C:\Users\Falah Akbar>heroku login
heroku: Press any key to open up the browser to login or q to exit:
Opening browser to https://cli-auth.heroku.com/auth/cli/browser/2d900210-f203-4e6b-badc-e28e43109d20?requestor=SFMyNTY.g
2gDbQAAAwXNS4xNTguNi4xODNuBgBwXcIH1AFiAAFRgA.rBEdAgDkIaaZx6aif3XVUXCFcesus33Twa_isswRL-0
Logging in... done
Logged in as falahkuv@student.telkomuniversity.ac.id
```

Membuat directory

untuk tugas 15 dengan perintah membuat aplikasi scalable, maka kami membuat directory baru sebelum inisiasi project ini.

```
C:\Users\Falah Akbar>mkdir scalable-app-tugas-15 && cd scalable-app-tugas-15
C:\Users\Falah Akbar\scalable-app-tugas-15>_
```

BACKEND

Inisialisasi project

Disini kami menginisialisasi project dengan memakai node.js serta menginstall express.

```
C:\Users\Falah Akbar\scalable-app-tugas-15>npm init -y
Wrote to C:\Users\Falah Akbar\scalable-app-tugas-15\package.json:
```

```
{
  "name": "scalable-app-tugas-15",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": ""
}
```

```
added 68 packages, and audited 69 packages in 6s
```

```
14 packages are looking for funding
  run `npm fund` for details
```

```
found 0 vulnerabilities
```

Membuat file index.js

*index.js - Notepad

File Edit Format View Help

```
const express = require('express');
const app = express();
const PORT = process.env.PORT || 3000;

app.get('/data', (req, res) => {
  res.send({ message: "Scalable App kelompok 1 is RUNNING!" });
});

app.listen(PORT, () => console.log(`Server is running on port ${PORT}`));
```

Membuat file Procfile

```
C:\Users\Falah Akbar\scalable-app-tugas-15>echo web: node index.js > Procfile
C:\Users\Falah Akbar\scalable-app-tugas-15>type Procfile
web: node index.js
```

Membuat file package.json

*package.json - Notepad

File Edit Format View Help

```
{
  "name": "scalable-app-tugas-15",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "start": "node index.js"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "description": "",
  "dependencies": {
    "express": "^4.21.2"
  }
}
```

DEPLOY KE HEROKU

Inisialisasi ke Git

```
C:\Users\Falah Akbar\scalable-app-tugas-15>git init
Initialized empty Git repository in C:/Users/Falah Akbar/scalable-app-tugas-15/.git/
C:\Users\Falah Akbar\scalable-app-tugas-15>git add .
C:\Users\Falah Akbar\scalable-app-tugas-15>git commit -m "initial commit"
[master (root-commit) fc8b2d2] initial commit
612 files changed, 66406 insertions(+)
```

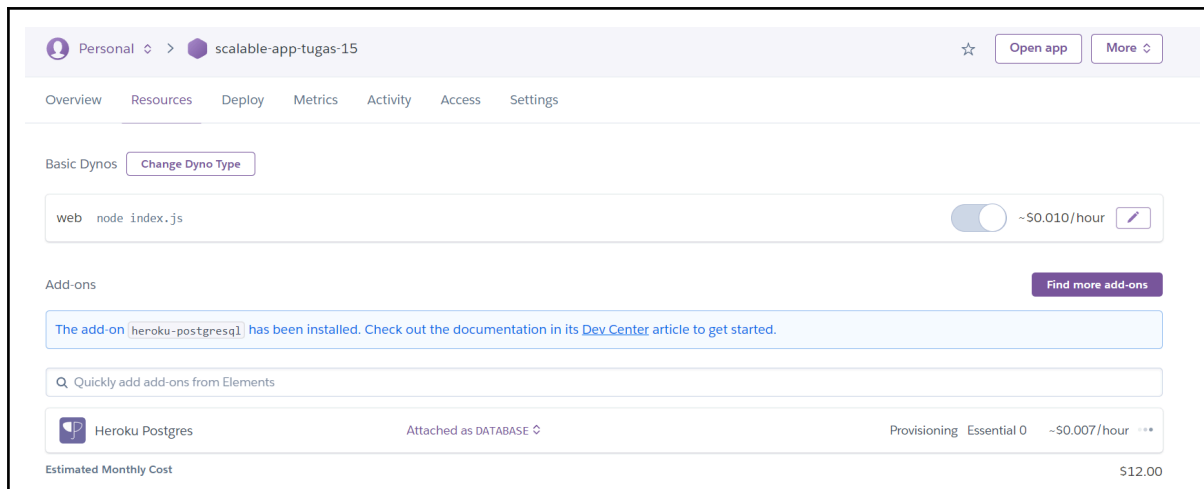
Membuat aplikasi di Heroku

```
C:\Users\Falah Akbar\scalable-app-tugas-15>heroku create scalable-app-tugas-15
Creating scalable-app-tugas-15... done
https://scalable-app-tugas-15-39a09abbe5fb.herokuapp.com/ | https://git.heroku.com/scalable-app-tugas-15.git
```

Push kode ke Heroku

```
C:\Users\Falah Akbar\scalable-app-tugas-15>git push heroku main
Enumerating objects: 703, done.
Counting objects: 100% (703/703), done.
Delta compression using up to 4 threads
Compressing objects: 100% (650/650), done.
Writing objects: 100% (703/703), 703.76 KiB | 3.80 MiB/s, done.
Total 703 (delta 130), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (130/130), done.
remote: Updated 612 paths from 958f8cd
remote: Compressing source files... done.
remote: Building source:
remote:
```

Menambahkan PostgreSQL



SCALING APP

Cek status scaling Dyno

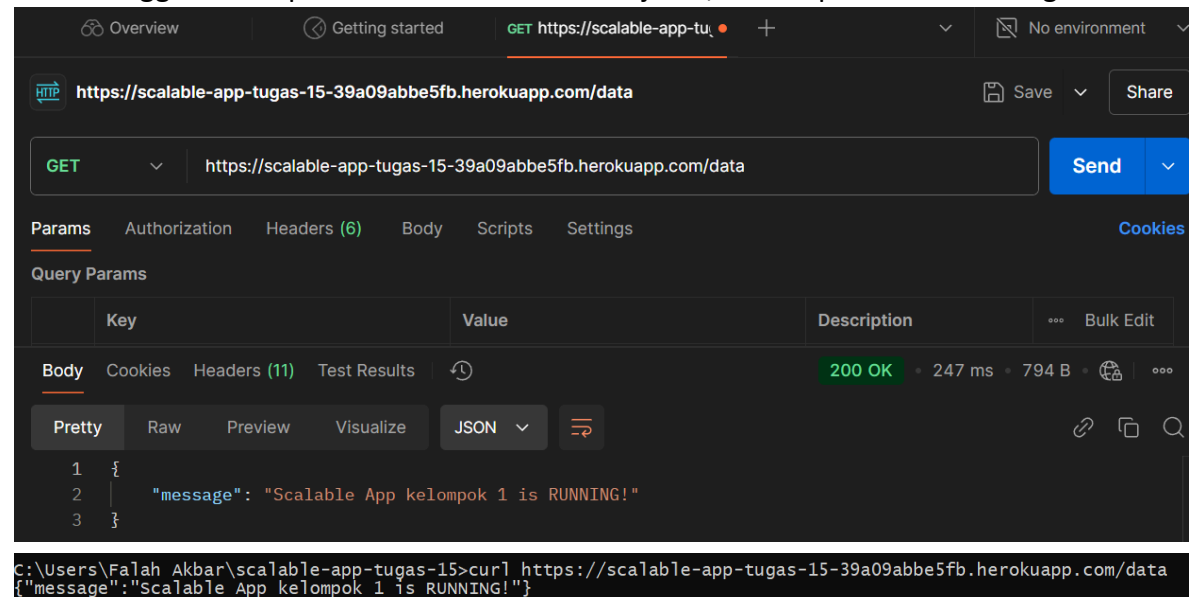
Kami memakai 1 Dyno saja agar hemat biaya

```
C:\Users\Falah Akbar\scalable-app-tugas-15>heroku ps --app scalable-app-tugas-15
=== web (Basic): node index.js (1)
web.1: up 2024/12/27 18:47:20 +0700 (~ 3m ago)
```

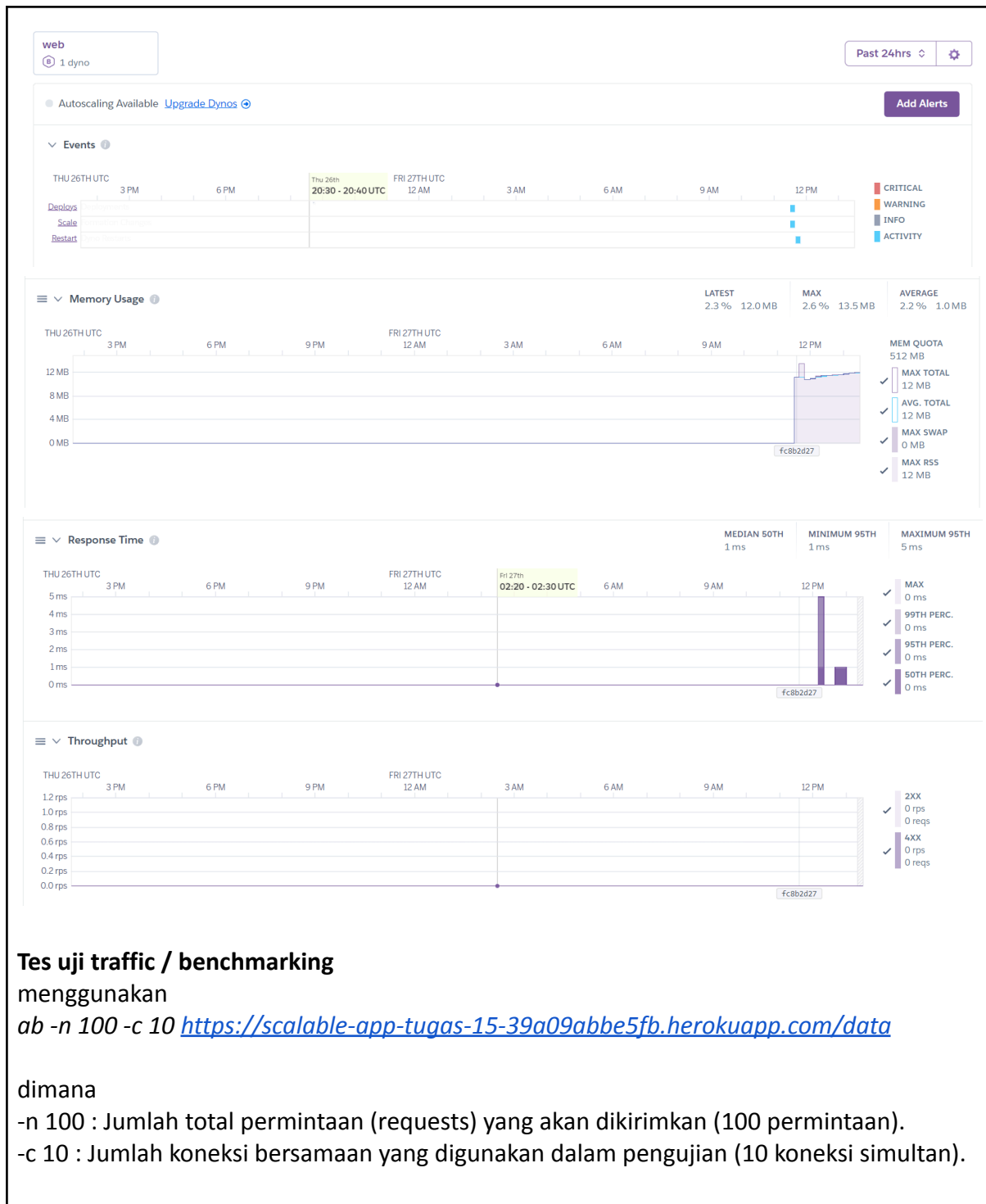
UJI HASIL IMPLEMENTASI

Tes API Endpoint

Kami menggunakan aplikasi Postman untuk tes uji API, dan respon diterima dengan benar.



Performa traffic



Tes uji traffic / benchmarking menggunakan

ab -n 100 -c 10 <https://scalable-app-tugas-15-39a09abbe5fb.herokuapp.com/data>

dimana

-n 100 : Jumlah total permintaan (requests) yang akan dikirimkan (100 permintaan).

-c 10 : Jumlah koneksi bersamaan yang digunakan dalam pengujian (10 koneksi simultan).

```
C:\Program Files\Apache24\bin>abs -n 100 -c 10 https://scalable-app-tugas-15-39a09abbe5fb.herokuapp.com/data
This is ApacheBench, Version 2.3 <$Revision: 1913912 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/
```

```
Benchmarking scalable-app-tugas-15-39a09abbe5fb.herokuapp.com (be patient).....done
```

```
Server Software:      Cowboy
Server Hostname:      scalable-app-tugas-15-39a09abbe5fb.herokuapp.com
Server Port:          443
SSL/TLS Protocol:     TLSv1.2,ECDHE-RSA-AES128-GCM-SHA256,2048,128
Server Temp Key:       ECDH prime256v1 256 bits
TLS Server Name:      scalable-app-tugas-15-39a09abbe5fb.herokuapp.com

Document Path:        /data
Document Length:       49 bytes

Concurrency Level:     10
Time taken for tests:   97.619 seconds
Complete requests:     100
Failed requests:        0
Total transferred:     79432 bytes
HTML transferred:      4900 bytes
Requests per second:    1.02 [#/sec] (mean)
Time per request:       9761.900 [ms] (mean)
Time per request:       976.190 [ms] (mean, across all concurrent requests)
Transfer rate:          0.79 [Kbytes/sec] received
```

```
Connection Times (ms)
      min  mean[+/-sd] median   max
Connect:    725   6034  9482.8   2003  48357
Processing:   240   3508  5951.3   1370  31543
Waiting:     240   2446  4967.1    551  18533
Total:      1111   9542 11056.4   3376  48627
```

```
Percentage of the requests served within a certain time (ms)
 50%    3376
 66%    7508
 75%   17486
 80%   18566
 90%   32533
 95%   33607
 98%   48613
 99%   48627
100%   48627 (longest request)
```

Kesimpulan

Berdasarkan hasil uji yang telah dilakukan, maka dengan menggunakan scaling, kita dapat mendistribusikan beban kerja ke beberapa server, sehingga meningkatkan kapasitas dan ketersediaan layanan.