

TUGAS PENDAHULUAN
KONSTRUKSI PERANGKAT LUNAK
MODUL IX
API DESIGN & CONSTRUCTION USING SWAGGER



Disusun Oleh:
Maria Nathasya Desfera Pangestu
2211104008
SE0601

Dosen Pengampu:
Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

Tugas Pendahuluan

1. IMPLEMENTASI WEB API Dari master/main branch dan class utama, buatlah program/aplikasi web API dari spesifikasi sebagai berikut ini:

a. API yang dibuat menggunakan data dari kelas Mahasiswa.

Mahasiswa
- Nama: string
- Nim: string
+ Mahasiswa

b. API yang dibuat mempunyai lokasi sebagai berikut ‘/api/mahasiswa’, URL domain boleh dari port mana saja, misalnya <https://localhost:5001/api/mahasiswa> (port bebas)

c. Secara default, program yang dibuat memiliki array/list mahasiswa dari anggota kelompok anda (tuliskan nama anda di urutan pertama/paling atas), contohnya:

- Nama: “LeBron James”, Nim: “1302000001”
- Nama: “Stephen Curry”, Nim: “1302000002” dst

d. Gunakan teknologi API sehingga program tersebut dapat menerima HTTP request sebagai berikut:

Mahasiswa	
GET	/api/Mahasiswa
POST	/api/Mahasiswa
GET	/api/Mahasiswa/{id}
DELETE	/api/Mahasiswa/{id}

- GET /api/mahasiswa: mengembalikan output berupa list/array dari semua objek mahasiswa yang tersimpan
 - GET /api/mahasiswa/{index}: mengembalikan output berupa objek mahasiswa untuk index ke-‘index’
 - POST /api/mahasiswa: menambahkan objek mahasiswa baru dengan menyertakan nama dan nim
 - DELETE /api/mahasiswa/{index}: menghapus objek mahasiswa dengan index ke ‘index’
- e. Implementasi yang dibuat tidak menggunakan database, cukup disimpan sebagai suatu variable, dan gunakan “static” di variable tersebut yang menyimpan list/array dari objek-objek mahasiswa
- f. Dalam pembuatan program/aplikasi ini, anda dapat mengasumsikan bahwa input dari user selalu benar dan sesuai dengan tipe data yang diharapkan.

Jawab

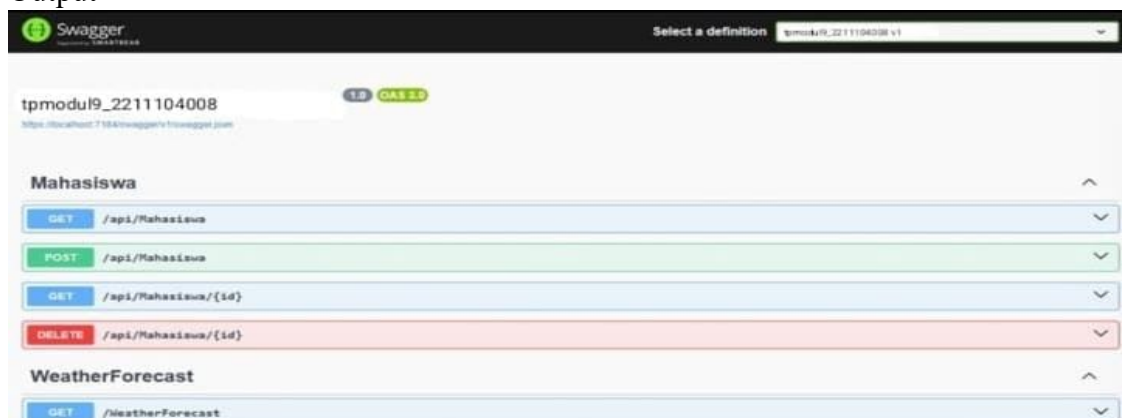
- Source code

```

1  using Microsoft.AspNetCore.Mvc;
2  using System.Collections.Generic;
3  using tp_modul9.Models;
4
5  [ApiController]
6  [Route("api/[controller]")]
7  public class MahasiswaController : ControllerBase
8  {
9      private static List<Mahasiswa> listMahasiswa = new List<Mahasiswa>
10     {
11         new Mahasiswa { Nama = "Maria Nathasya", Nim = "2211104008" },
12         new Mahasiswa { Nama = "Alfian Mutakim", Nim = "2211104017" },
13         new Mahasiswa { Nama = "Nadia Putri Rahmiani", Nim = "2211104012" },
14         new Mahasiswa { Nama = "Muhammad Edgar Nadhif", Nim = "2211104028" },
15         new Mahasiswa { Nama = "Muhammad Dhimas Afrizal", Nim = "2211104025" },
16     };
17
18     [HttpGet]
19     public IEnumerable<Mahasiswa> Get()
20     {
21         return listMahasiswa;
22     }
23
24     [HttpGet("{id}")]
25     public ActionResult<Mahasiswa> Get(int id)
26     {
27         if (id < 0 || id >= listMahasiswa.Count)
28             return NotFound();
29         return listMahasiswa[id];
30     }
31
32     [HttpPost]
33     public void Post([FromBody] Mahasiswa mhs)
34     {
35         listMahasiswa.Add(mhs);
36     }
37
38     [HttpDelete("{id}")]
39     public void Delete(int id)
40     {
41         if (id >= 0 && id < listMahasiswa.Count)
42             listMahasiswa.RemoveAt(id);
43     }
44 }

```

- Output



- Penjelasan

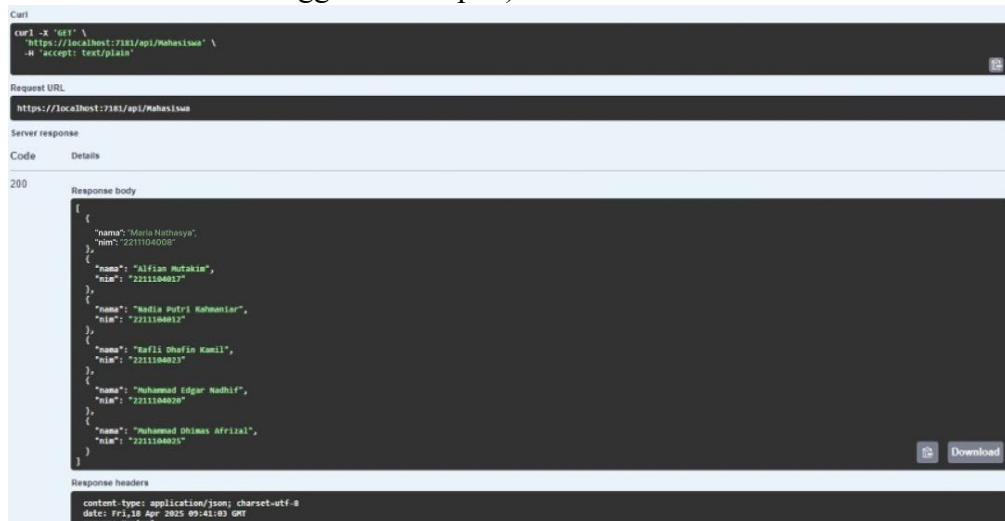
Program ini adalah implementasi Web API sederhana menggunakan ASP.NET Core yang berfungsi untuk mengelola data mahasiswa. API ini menyimpan informasi mahasiswa dalam sebuah list statis yang disebut listMahasiswa, yang mencakup data seperti nama dan NIM. Untuk menangani permintaan dari klien, digunakan sebuah controller bernama MahasiswaController. API ini menyediakan beberapa endpoint,

antara lain: GET untuk menampilkan semua data mahasiswa atau satu mahasiswa berdasarkan indeks, POST untuk menambahkan data mahasiswa baru, dan DELETE untuk menghapus data mahasiswa berdasarkan indeks. Setiap endpoint dikonfigurasi dengan atribut seperti [HttpGet], [HttpPost], dan [HttpDelete], yang menunjukkan jenis permintaan HTTP yang ditangani. Misalnya, ketika pengguna mengakses GET /api/mahasiswa, server akan merespons dengan semua data mahasiswa yang ada. Untuk POST /api/mahasiswa, data mahasiswa yang dikirimkan akan ditambahkan ke dalam list. Perlu dicatat bahwa semua data disimpan di memori menggunakan list statis, sehingga data bersifat sementara dan akan hilang saat aplikasi dimatikan atau server di-restart.

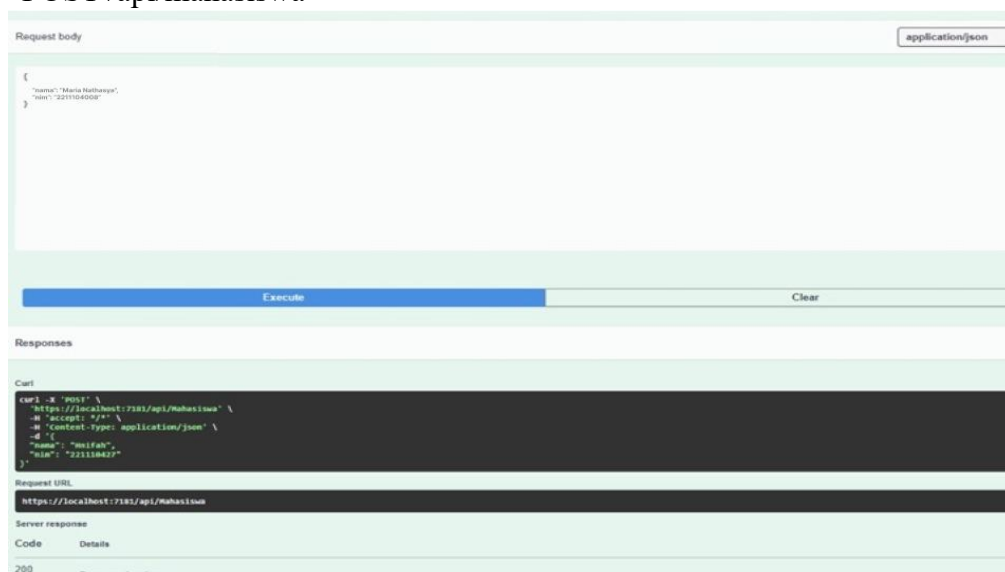
2. MENDEMONSTRASI WEB API

Beberapa skenario yang harus dicoba untuk memastikan jika program telah berjalan dengan baik. Buatlah dokumen yang berisi semua screenshot dari hasil uji coba scenario yang disebutkan pada list berikut ini:

- a. Mencoba “GET /api/mahasiswa” saat baru dijalankan (mengeluarkan list nama mahasiswa dan nim anggota kelompok)



- b. Menambahkan mahasiswa => Nama: “John Doe” dan NIM: “1302199999” dengan “POST /api/mahasiswa”



- c. Cek list/array dari semua mahasiswa lagi dengan “GET /api/mahasiswa”, pastikan mahasiswa yang baru ditambahkan sebelumnya ada di list mahasiswa

Request URL: `https://localhost:7181/api/mahasiswa`

Server response

Code: 200

Response body

```
{
  "nama": "Maria Nathasya",
  "nim": "2211104008"
},
{
  "nama": "Alfian Mutakin",
  "nim": "2211104011"
},
{
  "nama": "Nadia Putri Rahmania",
  "nim": "2211104012"
},
{
  "nama": "Muhammad Rizwan Afrizal",
  "nim": "2211104015"
},
{
  "nama": "Mufah",
  "nim": "2211104027"
},
{
  "nama": "Mufah",
  "nim": "2211104027"
}
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Fri, 18 Apr 2025 09:47:53 GMT
server: Restful
```

Responses

Code	Description	Links
200	OK	No links

Media type:

Controls: Accept header

- d. Mencoba meminta mahasiswa dengan index 0, “GET /api/mahasiswa/0” yang seharusnya mengeluarkan nama dan nim anda

GET /api/mahasiswa/{id}

Parameters

Name	Description
id * required	Integer(Integer) (path)

id: 0

Execute

Responses

Code: 200

Response body

```
{
  "nama": "Maria Nathasya",
  "nim": "2211104008"
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Fri, 18 Apr 2025 09:50:50 GMT
server: Restful
```

Responses

Code	Description	Links
200	OK	No links

- e. Menghapus objek mahasiswa dengan index ke-0 dengan “DELETE /api/mahasiswa/0”

DELETE

/api/Mahasiswa/{id}

Cancel

Parameters

Name	Description
id <small>* required</small>	
integer(int32)	0
(path)	

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
  "https://localhost:7381/api/Mahasiswa/0" \
  -H 'accept: */*'
```

Request URL

https://localhost:7381/api/Mahasiswa/0

Server response

Code	Details
200	<div>Response headers</div> <div> <pre>content-length: 0 date: Fri, 18 Apr 2025 09:51:26 GMT server: Restful</pre> </div>

Responses

Code	Description	Links
200	OK	No links

- f. Cek list/array dari semua mahasiswa sekali lagi dengan “GET /api/mahasiswa”, pastikan nama anda sudah tidak muncul di list tersebut

GET

/api/Mahasiswa

Cancel

Parameters

No parameters

Execute

Clear

Responses

Curl

```
curl -X 'GET' \
  "https://localhost:7381/api/Mahasiswa" \
  -H 'accept: text/plain'
```

Request URL

https://localhost:7381/api/Mahasiswa

Server response

Code	Details
200	<div>Response body</div> <div> <pre>{ { "name": "Ilfin Mufkin", "nim": "2211304017" }, { "name": "Nadia Putri Rahmawati", "nim": "2211304012" }, { "name": "Muhammad Ghimis Afrizal", "nim": "2211304025" }, { "name": "Mufah", "nim": "2211304027" }, { "name": "Mufah", "nim": "2211304027" } }</pre> </div> <div>Download</div>

Response headers

```
content-type: application/json; charset=utf-8
date: Fri, 18 Apr 2025 09:52:13 GMT
server: Restful
```