

**TUGAS PENDAHULUAN
KONSTRUKSI PERANGKAT LUNAK**

**MODUL IX
API DESIGN & CONSTRUCTION USING
SWAGGER**



Disusun Oleh:

Nadia Putri Rahmانيar / 2211104012

S1 SE-06-01

Dosen Pengampu:

Yudha Islami Sulistya, S.Kom., M.Cs

**PROGRAM STUDI S1 SOFTWARE ENGINEERING 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

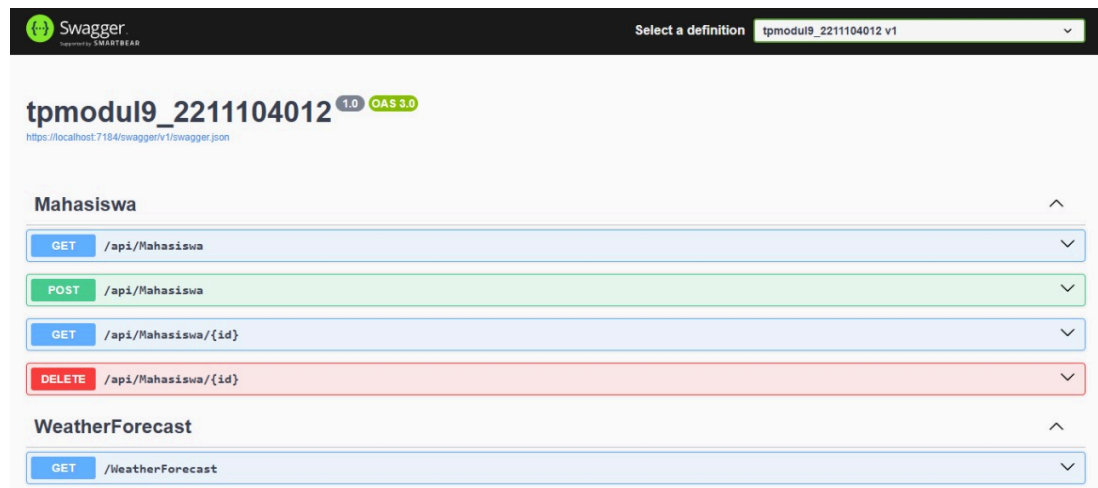
- f. Dalam pembuatan program/aplikasi ini, anda dapat mengasumsikan bahwa input dari user selalu benar dan sesuai dengan tipe data yang diharapkan.

Jawab :

Source code :

```
1  using Microsoft.AspNetCore.Mvc;
2  using System.Collections.Generic;
3
4  namespace tp_modul9.Controllers
5  {
6      [ApiController]
7      [Route("api/[controller]")]
8      public class MahasiswaController : ControllerBase
9      {
10         public class Mahasiswa
11         {
12             public string Nama { get; set; }
13             public string Nim { get; set; }
14         }
15
16         private static List<Mahasiswa> listMahasiswa = new List<Mahasiswa>
17         {
18             new Mahasiswa { Nama = "Nita Fitrotul Mar'ah", Nim = "2211104005" },
19             new Mahasiswa { Nama = "Alfian Mutakim", Nim = "2211104017" },
20             new Mahasiswa { Nama = "Nadia Putri Rahmaniar", Nim = "2211104012" },
21             new Mahasiswa { Nama = "Rafla Dafin Kamil", Nim = "2211104023" },
22             new Mahasiswa { Nama = "Muhammad Edgar Nadhif", Nim = "22111040028" },
23             new Mahasiswa { Nama = "Muhammad Dhimas Afrizal", Nim = "22111040025" },
24         };
25
26         // GET /api/mahasiswa
27         [HttpGet]
28         public IEnumerable<Mahasiswa> Get()
29         {
30             return listMahasiswa;
31         }
32
33         // GET /api/mahasiswa/{id}
34         [HttpGet("{id}")]
35
36         public ActionResult<Mahasiswa> Get(int id)
37         {
38             if (id < 0 || id >= listMahasiswa.Count)
39                 return NotFound();
40             return listMahasiswa[id];
41         }
42
43         // POST /api/mahasiswa
44         [HttpPost]
45         public void Post([FromBody] Mahasiswa mhs)
46         {
47             listMahasiswa.Add(mhs);
48         }
49
50         // DELETE /api/mahasiswa/{id}
51         [HttpDelete("{id}")]
52         public void Delete(int id)
53         {
54             if (id >= 0 && id < listMahasiswa.Count)
55             {
56                 listMahasiswa.RemoveAt(id);
57             }
58         }
59     }
```

Hasil run :



Penjelasan

Program tersebut merupakan implementasi Web API sederhana menggunakan ASP.NET Core yang digunakan untuk mengelola data mahasiswa. API ini menyimpan data mahasiswa dalam sebuah list statis bernama `listMahasiswa`, yang berisi informasi seperti nama dan NIM. Untuk menangani permintaan dari klien, digunakan sebuah controller bernama `MahasiswaController`.

Beberapa endpoint disediakan dalam API ini, seperti:

- GET untuk menampilkan seluruh data mahasiswa atau satu mahasiswa berdasarkan indeks,
- POST untuk menambahkan data mahasiswa baru, dan
- DELETE untuk menghapus data mahasiswa berdasarkan indeks.

Setiap endpoint dikonfigurasi menggunakan atribut seperti `[HttpGet]`, `[HttpPost]`, dan `[HttpDelete]` yang menunjukkan jenis permintaan HTTP yang ditangani. Contohnya, ketika pengguna mengakses `GET /api/mahasiswa`, maka server akan merespons dengan seluruh data mahasiswa yang tersedia. Sedangkan untuk `POST /api/mahasiswa`, data mahasiswa yang dikirimkan akan ditambahkan ke list.

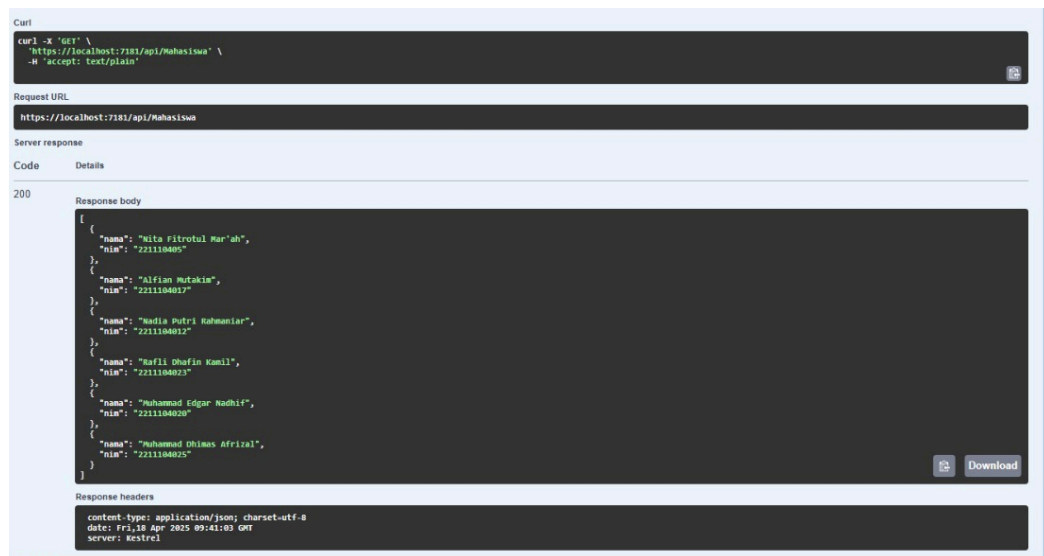
Perlu dicatat bahwa semua data disimpan di memori menggunakan list statis, sehingga data hanya bersifat sementara dan akan hilang ketika aplikasi dimatikan atau server di-restart.

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)



```
curl -X 'GET' \
  'https://localhost:7181/api/mahasiswa' \
  -H 'accept: text/plain'
```

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

Server response

Code Details

200

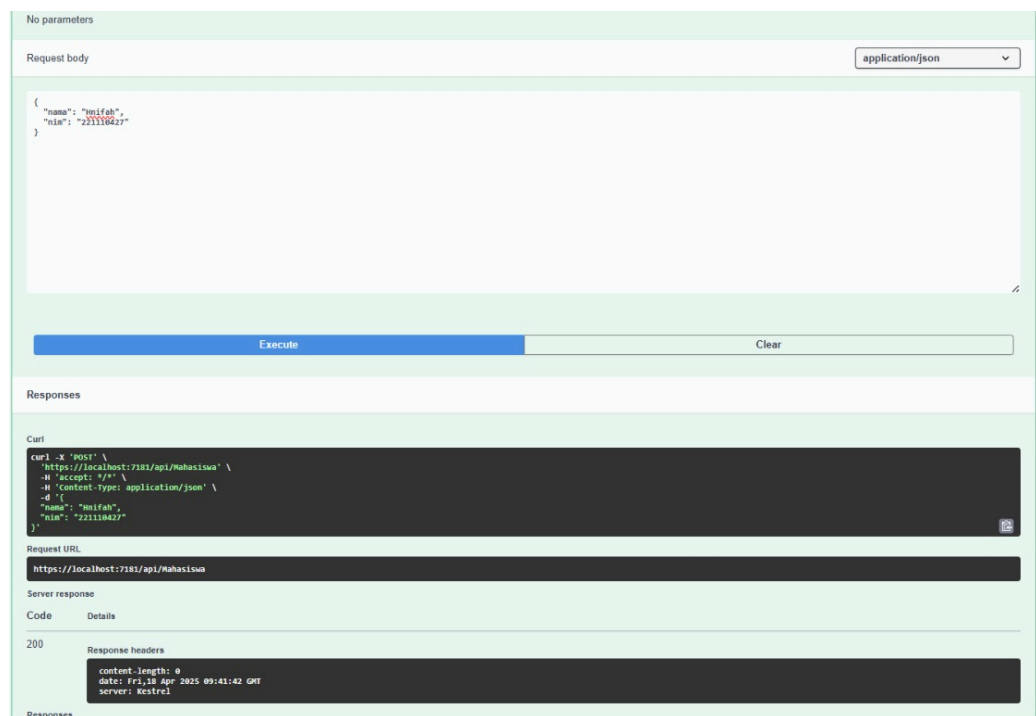
Response body

```
{
  "nama": "Mita Fitrotul Mar'ah",
  "nim": "221110405"
},
{
  "nama": "Alfian Mufakim",
  "nim": "2211104017"
},
{
  "nama": "Nadia Putri Rahmania",
  "nim": "2211104012"
},
{
  "nama": "Rafli Dhuafin Kamil",
  "nim": "2211104023"
},
{
  "nama": "Muhammad Edgar Nadhif",
  "nim": "2211104028"
},
{
  "nama": "Muhammad Olimas Afrizal",
  "nim": "2211104015"
}
}
```

Response headers

```
content-type: application/json; charset=utf-8
date: Fri, 18 Apr 2025 09:41:03 GMT
server: Kestrel
```

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



No parameters

Request body application/json

```
{
  "nama": "Hmifah",
  "nim": "221110427"
}
```

Execute Clear

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7181/api/mahasiswa' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "nama": "Hmifah",
    "nim": "221110427"
  }'
```

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

Server response

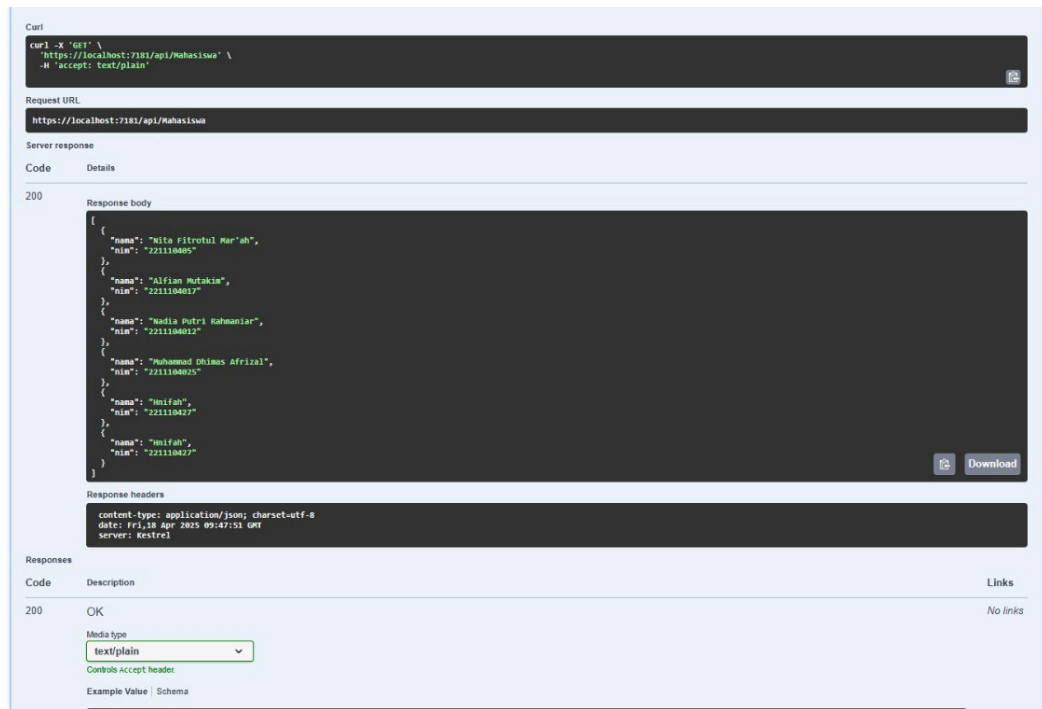
Code Details

200

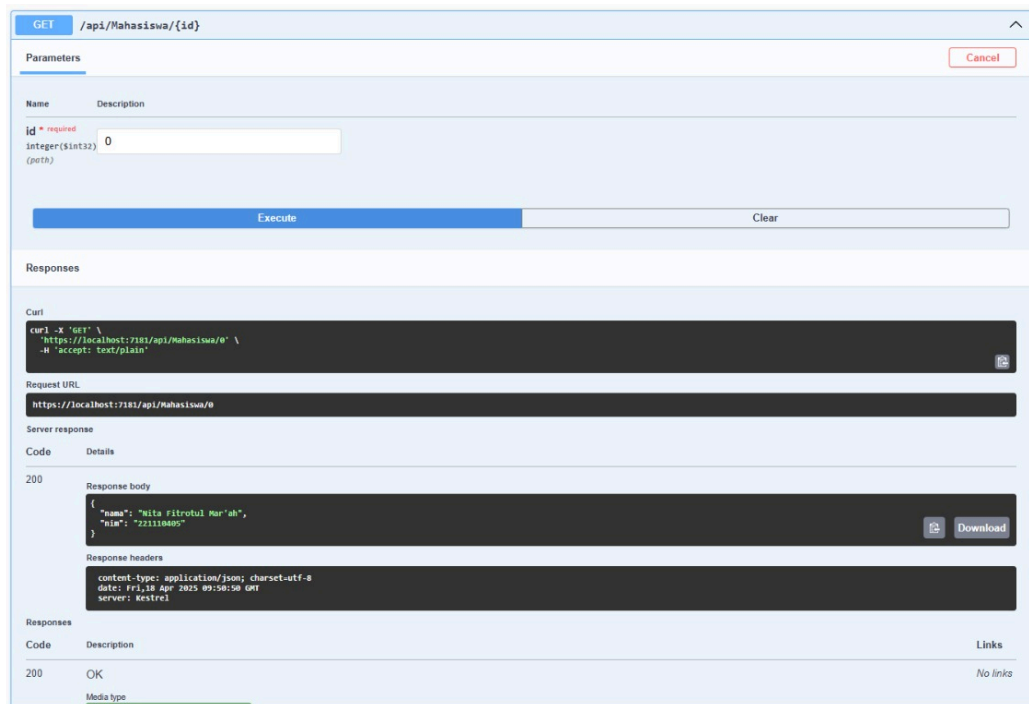
Response headers

```
content-length: 0
date: Fri, 18 Apr 2025 09:41:42 GMT
server: Kestrel
```

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



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



- 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) (path)

0

Execute

Clear

Responses

Curl

```
curl -X 'DELETE' \
'https://localhost:7181/api/Mahasiswa/0' \
-H 'accept: */*'
```

Request URL

https://localhost:7181/api/Mahasiswa/0

Server response

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

Responses

Code	Description	Links
200	OK	No links

WeatherForecast

GET

/WeatherForecast

- 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:7181/api/Mahasiswa' \
-H 'accept: text/plain'
```

Request URL

https://localhost:7181/api/Mahasiswa

Server response

Code	Details
200	<div>Response body</div> <div> <pre>{ { "nama": "Alfin Muftikin", "nim": "2211104017" }, { "nama": "Nadia Putri Rahmawati", "nim": "2211104012" }, { "nama": "Muhammad Ghimus Afrizal", "nim": "2211104025" }, { "nama": "Mufah", "nim": "2211104027" }, { "nama": "Mufah", "nim": "2211104027" } }</pre> </div> <div>Download</div>

Responses

Code	Description	Links
200	OK	No links

Responses

Response headers

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