

**TUGAS PENDAHULUAN  
KONSTRUKSI PERANGKAT LUNAK**

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



**Disusun Oleh :**

Rizky Hanifa Afania

2211104017

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:

- i. Nama: “LeBron James”, Nim: “1302000001”
- i. Nama: “Stephen Curry”, Nim: “1302000002”
- iii. 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

```
using Microsoft.AspNetCore.Mvc;
using System.Collections.Generic;

namespace tp_modul9.Controllers
{
    [ApiController]
    [Route("api/[controller]")]
    0 references
    public class MahasiswaController : ControllerBase
    {
        10 references
        public class Mahasiswa
        {
            5 references
            public string Nama { get; set; }
            5 references
            public string Nim { get; set; }
        }

        private static List<Mahasiswa> listMahasiswa = new List<Mahasiswa>
        {
            new Mahasiswa { Nama = "Rizky Hanifa Afania", Nim = "2211104017" },
            new Mahasiswa { Nama = "Althafia Defiyandrea L.W", Nim = "2211104011" },
            new Mahasiswa { Nama = "Muhammad Idham C", Nim = "2211104016" },
            new Mahasiswa { Nama = "Fauzan Wahyu M", Nim = "2211104027" },
            new Mahasiswa { Nama = "Andera Singgih P", Nim = "2211104007" },
        };

        // GET /api/mahasiswa
        [HttpGet]
        0 references
        public IEnumerable<Mahasiswa> Get()
        {
            return listMahasiswa;
        }
    }
}
```

```

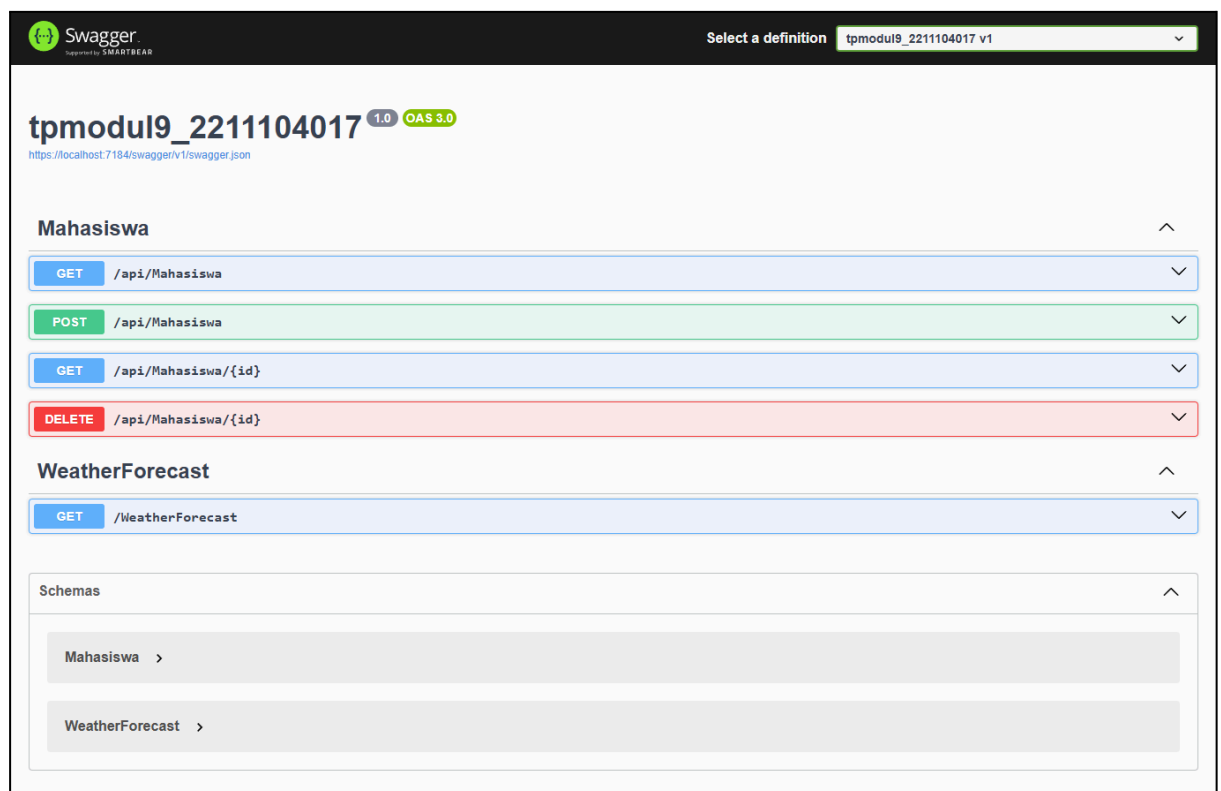
// GET /api/mahasiswa/{id}
[HttpGet("{id}")]
0 references
public ActionResult<Mahasiswa> Get(int id)
{
    if (id < 0 || id >= listMahasiswa.Count)
        return NotFound();
    return listMahasiswa[id];
}

// POST /api/mahasiswa
[HttpPost]
0 references
public void Post([FromBody] Mahasiswa mhs)
{
    listMahasiswa.Add(mhs);
}

// DELETE /api/mahasiswa/{id}
[HttpDelete("{id}")]
0 references
public void Delete(int id)
{
    if (id >= 0 && id < listMahasiswa.Count)
    {
        listMahasiswa.RemoveAt(id);
    }
}
}

```

- Screenshot hasil run



## - Penjelasan

Program di atas menerapkan Web API sederhana menggunakan ASP.NET Core yang menangani data mahasiswa. API ini menyimpan daftar mahasiswa dalam sebuah list statis bernama `listMahasiswa`, yang berisi beberapa data nama dan NIM. API ini menggunakan controller bernama `MahasiswaController` yang mengatur permintaan (request) HTTP ditangani oleh server. Ada beberapa endpoint: GET untuk menampilkan semua mahasiswa atau mahasiswa berdasarkan indeks, POST untuk menambahkan mahasiswa baru, dan DELETE untuk menghapus mahasiswa berdasarkan indeks.

Setiap endpoint diatur dengan atribut seperti `[HttpGet]`, `[HttpPost]`, dan `[HttpDelete]` untuk menunjukkan jenis permintaan HTTP yang digunakan. Misalnya, saat pengguna mengakses `GET /api/mahasiswa`, server akan mengembalikan seluruh daftar mahasiswa. Jika pengguna mengakses `POST /api/mahasiswa` dan mengirim data mahasiswa baru, maka data tersebut akan ditambahkan ke dalam list. Semua data disimpan di dalam memori (bukan di database), dan karena menggunakan list statis, data hanya bertahan selama aplikasi berjalan (kalau server di-restart, data kembali ke kondisi awal).

## 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 skenario yang disebutkan pada list berikut ini:

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

**Hasil:**



```
Code    Details
200
Response body
[
  {
    "nama": "Rizky Hanifa Afania",
    "nim": "2211104017"
  },
  {
    "nama": "Althafia Defiyandrea L.W",
    "nim": "2211104011"
  },
  {
    "nama": "Muhammad Idham C",
    "nim": "2211104016"
  },
  {
    "nama": "Fauzan Mahyu M",
    "nim": "2211104027"
  },
  {
    "nama": "Andera Singgih P",
    "nim": "2211104007"
  }
]
Response headers
content-type: application/json; charset=utf-8
date: Wed, 16 Apr 2025 11:58:23 GMT
server: Kestrel
```

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

### Hasil:

The screenshot shows a REST client interface with the following sections:

- POST /api/Mahasiswa**: The endpoint and method.
- Parameters**: No parameters are listed.
- Request body**: A JSON object: 

```
{  "nama": "John Doe",  "nim": "1302199999"}
```
- Execute**: A blue button to execute the request.
- Responses**:
  - Curl**: A terminal window showing the curl command: 

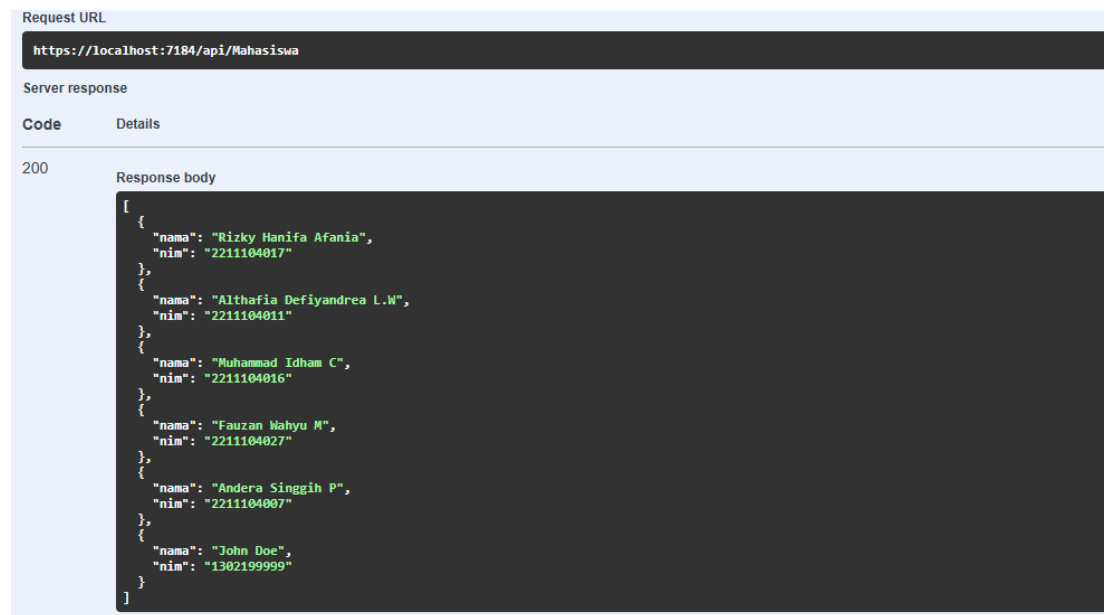
```
curl -X 'POST' \  'https://localhost:7184/api/Mahasiswa' \  -H 'accept: */*' \  -H 'Content-Type: application/json' \  -d '{  "nama": "John Doe",  "nim": "1302199999"  }'
```
  - Request URL**: `https://localhost:7184/api/Mahasiswa`
  - Server response**:
    - Code**: 200
    - Details**:
      - Response headers**:

```
content-length: 0date: Wed, 16 Apr 2025 12:05:24 GMTserver: Kestrel
```
  - Responses**: A table with one entry:

Code	Description	Links
200	OK	No links

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

## Hasil:



Request URL

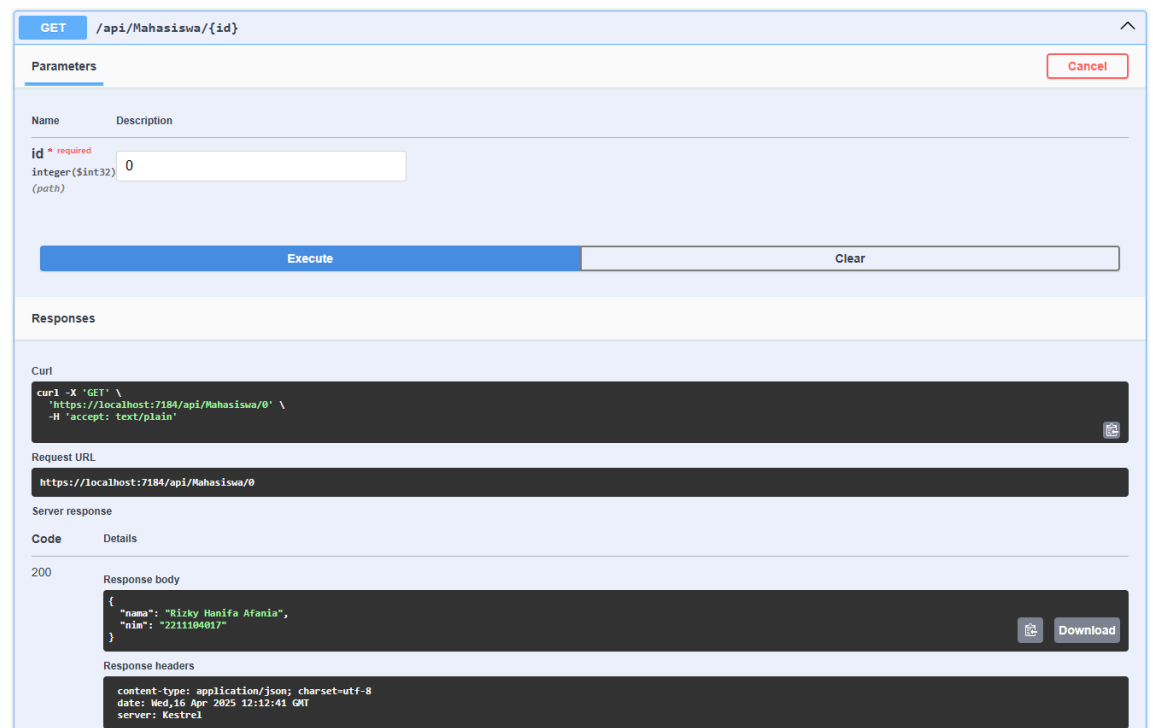
https://localhost:7184/api/Mahasiswa

Server response

Code	Details
200	<p>Response body</p> <pre>[   {     "nama": "Rizky Hanifa Afania",     "nim": "2211104017"   },   {     "nama": "Althafia Defiyandrea L.W",     "nim": "2211104011"   },   {     "nama": "Muhammad Idham C",     "nim": "2211104016"   },   {     "nama": "Fauzan Wahyu M",     "nim": "2211104027"   },   {     "nama": "Andera Singgih P",     "nim": "2211104007"   },   {     "nama": "John Doe",     "nim": "1302199999"   } ]</pre>

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

## Hasil:



GET /api/Mahasiswa/{id}

Parameters

Name	Description
id * required	
integer(\$int32)	
(path)	

0

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7184/api/Mahasiswa/0' \
  -H 'accept: text/plain'
```

Request URL

https://localhost:7184/api/Mahasiswa/0

Server response

Code	Details
200	<p>Response body</p> <pre>{   "nama": "Rizky Hanifa Afania",   "nim": "2211104017" }</pre> <p>Response headers</p> <pre>content-type: application/json; charset=utf-8 date: Wed, 16 Apr 2025 12:12:41 GMT server: Kestrel</pre>

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

## Hasil:

DELETE

/api/Mahasiswa/{id}

Parameters

Cancel

Name	Description
id * required	
integer(\$int32)	0
(path)	

Execute

Clear

Responses

Curl

curl -X 'DELETE' \ 'https://localhost:7184/api/Mahasiswa/0' \ -H 'accept: \*/\*'

Request URL

https://localhost:7184/api/Mahasiswa/0

Server response

Code	Details
200	<div>Response headers</div> <div>content-length: 0 date: Wed, 16 Apr 2025 12:16:40 GMT server: Kestrel</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:

## Hasil:

GET

/api/Mahasiswa

Parameters

Cancel

No parameters

Execute

Clear

Responses

Curl

curl -X 'GET' \ 'https://localhost:7184/api/Mahasiswa' \ -H 'accept: text/plain'

Request URL

https://localhost:7184/api/Mahasiswa

Server response

Code	Details
200	<div>Response body</div> <div><pre>[   {     "nama": "Althafia Defiyandrea L.W",     "nim": "2211194011"   },   {     "nama": "Muhammad Idham C",     "nim": "2211194016"   },   {     "nama": "Fauzan Mahyu M",     "nim": "2211194027"   },   {     "nama": "Andera Singgih P",     "nim": "2211194007"   },   {     "nama": "John Doe",     "nim": "1302199999"   } ]</pre></div> <div><div>Download</div></div>