

1. Link Github Repository

[MuhammadNurShodiq/tpmodul10_1302223054 \(github.com\)](https://github.com/MuhammadNurShodiq/tpmodul10_1302223054)

2. Hasil Run

- Get

The screenshot displays a REST client interface with the following details:

- Method:** GET
- URL:** /api/Mahasiswa
- Parameters:** No parameters
- Execute:** Button to run the request
- Responses:** Section showing the result of the request
- Request:**
 - curl:**

```
curl -X 'GET' \
  'http://localhost:5044/api/Mahasiswa' \
  -H 'accept: text/plain'
```
 - Request URL:** http://localhost:5044/api/Mahasiswa
- Server response:**
 - Code:** 200
 - Response body:**

```
[
  {
    "nama": "Muhammad Nur Shodiq",
    "nim": "1302223054"
  },
  {
    "nama": "Fauzein Mulya Warman",
    "nim": "1302223127"
  },
  {
    "nama": "Iksan Oktav Risandy",
    "nim": "1302223042"
  },
  {
    "nama": "Muhammad Dias Adani",
    "nim": "1302223134"
  },
  {
    "nama": "Muhammad Haulul Azkiyaa",
    "nim": "1302223007"
  }
]
```
 - Response headers:**

```
content-type: application/json; charset=utf-8
date: Sun, 28 Apr 2024 20:22:37 GMT
server: Kestrel
transfer-encoding: chunked
```
- Responses Table:**

Code	Description	Links
200	Success	No links

- Post

POST

/api/Mahasiswa

Parameters

Cancel

Reset

No parameters

Request body

application/json

```
{  "nama": "John Doe",  "nim": "1302199999"}
//
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
'http://localhost:5044/api/Mahasiswa' \
-H 'accept: */*' \
-H 'Content-Type: application/json' \
-d '{
  "nama": "John Doe",
  "nim": "1302199999"
}'
```

Request URL

http://localhost:5044/api/Mahasiswa

Server response

Code	Details
200	<div><div>Response headers</div><div>content-length: 0 date: Sun, 28 Apr 2024 20:23:44 GMT server: Kestrel</div></div>

Responses

Code	Description	Link
------	-------------	------

- Get John Doe

GET

/api/Mahasiswa/{id}

^

Parameters

Cancel

Name	Description
id * required	
integer(\$int32)	5
(path)	

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
'http://localhost:5044/api/Mahasiswa/5' \
-H 'accept: text/plain'
```

Request URL

```
http://localhost:5044/api/Mahasiswa/5
```

Server response

Code	Details
200	<div><div>Response body</div><div><pre>{ "name": "John Doe", "nim": "1302199999" }</pre></div><div>Download</div></div> <div><div>Response headers</div><div><pre>content-type: application/json; charset=utf-8 date: Sun, 28 Apr 2024 20:24:40 GMT server: Kestrel transfer-encoding: chunked</pre></div></div>

Responses

Code	Description	Links
200	Success	No links

Media type

text/plain

Controls Accept header.

Example Value | Schema

- Get our nim (index 0)

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

- GET /api/Mahasiswa/{id}**: The endpoint and method.
- Parameters**: A table with columns 'Name' and 'Description'. It contains one parameter:

Name	Description
id * required	

 Below the table, there is a text input field for 'id' with the value '0' and a label 'integer(\$int32) (path)'. There are 'Execute' and 'Clear' buttons.
- Responses**:
 - Curl**:

```
curl -X 'GET' \
'http://localhost:5044/api/Mahasiswa/0' \
-H 'accept: text/plain'
```
 - Request URL**: `http://localhost:5044/api/Mahasiswa/0`
 - Server response**:

Code	Details
200	<p>Response body</p> <pre>{ "nama": "Muhammad Nur Shodiq", "nim": "1302223054" }</pre> <p>Response headers</p> <pre>content-type: application/json; charset=utf-8 date: Sun, 28 Apr 2024 20:26:00 GMT server: Kestrel transfer-encoding: chunked</pre>
- Responses**: A table with columns 'Code', 'Description', and 'Links'.

Code	Description	Links
200	Success	No links

- Delete our nim (index 0)

DELETE /api/Mahasiswa/{id}

Parameters Cancel

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

Execute

LOADING

Responses

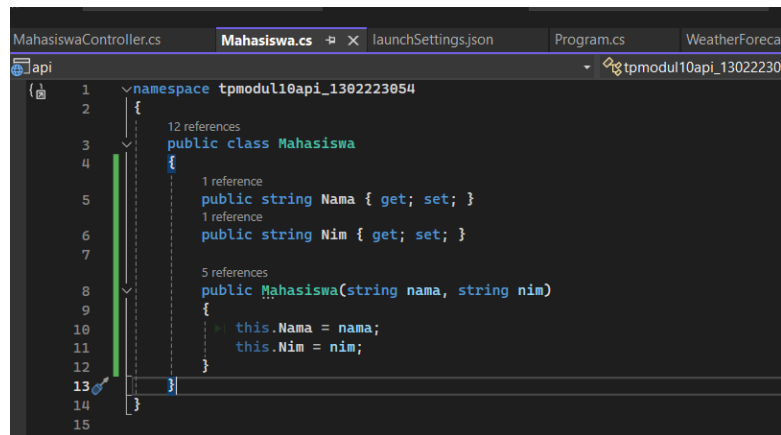
Code	Description	Links
200	Success	No links

Note: Selalu buffering, seharusnya berhasil karena telah memakai sintaks yang sudah sesuai dan dapat membaca error saat input diluar range index, namun saat memasukkan id 0 yang terjadi adalah buffering yang tidak selesai.

```
// DELETE api/<MahasiswaController>/  
[HttpDelete("{id}")]  
0 references  
public void Delete(int id)  
{  
    dataMahasiswa.RemoveAt(id);  
}
```

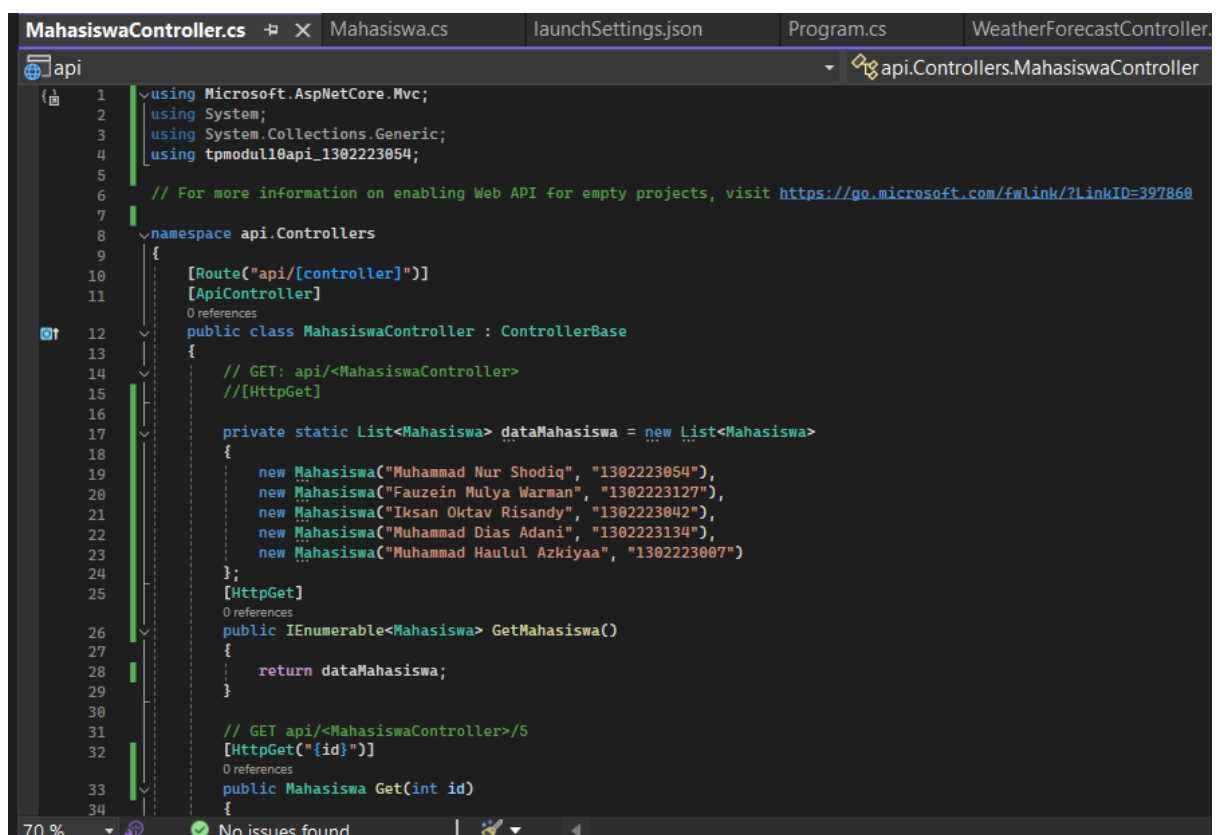
3. Code

a. Class Mahasiswa.cs

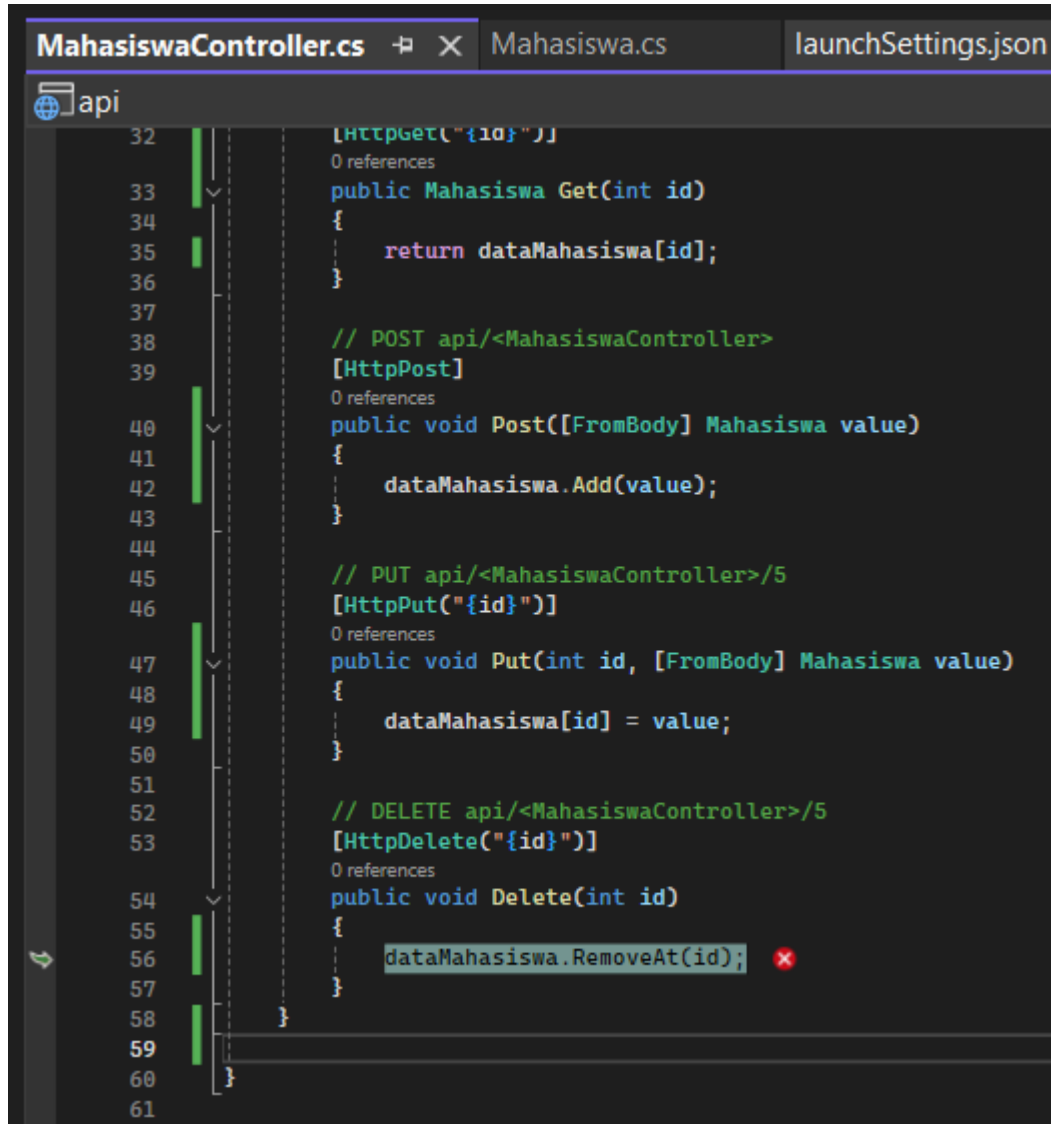


```
1 namespace tpmodul10api_1302223054
2 {
3     public class Mahasiswa
4     {
5         public string Nama { get; set; }
6         public string Nim { get; set; }
7
8         public Mahasiswa(string nama, string nim)
9         {
10             this.Nama = nama;
11             this.Nim = nim;
12         }
13     }
14 }
15
```

b. Controller MahasiswaController.cs



```
1 using Microsoft.AspNetCore.Mvc;
2 using System;
3 using System.Collections.Generic;
4 using tpmodul10api_1302223054;
5
6 // For more information on enabling Web API for empty projects, visit https://go.microsoft.com/fwlink/?LinkID=397860
7
8 namespace api.Controllers
9 {
10     [Route("api/[controller]")]
11     [ApiController]
12     public class MahasiswaController : ControllerBase
13     {
14         // GET: api/<MahasiswaController>
15         // [HttpGet]
16
17         private static List<Mahasiswa> dataMahasiswa = new List<Mahasiswa>
18         {
19             new Mahasiswa("Muhammad Nur Shodiq", "1302223054"),
20             new Mahasiswa("Fauzein Mulya Warman", "1302223127"),
21             new Mahasiswa("Iksan Oktav Risandy", "1302223042"),
22             new Mahasiswa("Muhammad Dias Adani", "1302223134"),
23             new Mahasiswa("Muhammad Haulul Azkiyaa", "1302223007")
24         };
25
26         [HttpGet]
27         public IEnumerable<Mahasiswa> GetMahasiswa()
28         {
29             return dataMahasiswa;
30         }
31
32         // GET api/<MahasiswaController>/5
33         [HttpGet("{id}")]
34         public Mahasiswa Get(int id)
35         {
36         }
37     }
38 }
```



```
32 [HttpGet("{id}")]
33 0 references
34 public Mahasiswa Get(int id)
35 {
36     return dataMahasiswa[id];
37 }
38
39 // POST api/<MahasiswaController>
40 [HttpPost]
41 0 references
42 public void Post([FromBody] Mahasiswa value)
43 {
44     dataMahasiswa.Add(value);
45 }
46
47 // PUT api/<MahasiswaController>/5
48 [HttpPut("{id}")]
49 0 references
50 public void Put(int id, [FromBody] Mahasiswa value)
51 {
52     dataMahasiswa[id] = value;
53 }
54
55 // DELETE api/<MahasiswaController>/5
56 [HttpDelete("{id}")]
57 0 references
58 public void Delete(int id)
59 {
60     dataMahasiswa.RemoveAt(id);
61 }
```

4. Penjelasan singkat kode

API yang dibuat memiliki fungsi untuk mengelola data mahasiswa.

Controller menyediakan beberapa fungsi endpoint untuk mengakses data mahasiswa, yaitu:

- GET /api/mahasiswa: Mengambil semua data mahasiswa.
- GET /api/mahasiswa/{id}: Mengambil data mahasiswa dengan id tertentu.
- POST /api/mahasiswa: Menambahkan data mahasiswa baru.
- PUT /api/mahasiswa/{id}: Memperbarui data mahasiswa dengan id tertentu.

- DELETE /api/mahasiswa/{id}: Menghapus data mahasiswa dengan id tertentu.

Data mahasiswa disimpan dalam array dataMahasiswa yang bersifat statis. Setiap mahasiswa memiliki properti Nama dan Nim.

Ketika pengguna melakukan request ke salah satu endpoint API, controller akan memproses request tersebut dan memberikan respons yang sesuai.

Contohnya, jika pengguna melakukan request GET /api/mahasiswa, controller akan mengambil semua data mahasiswa dari array dataMahasiswa dan mengembalikannya dalam format JSON.

API ini dapat digunakan oleh aplikasi lain untuk mengakses dan mengelola data mahasiswa.