

2311104023  
Yoga Eka Pratama  
SE0701

## Jurnal Modul 09

The screenshot displays the OpenAPI SwaggerUI preview for a RESTful API. The interface is split into two main sections: the API definition on the left and the SwaggerUI preview on the right.

**API Definition (Left Panel):**

```
1 {
2   "openapi": "3.0.0",
3   "info": {
4     "title": "Mahasiswa API",
5     "description": "API untuk Manajemen Data Mahasiswa",
6     "version": "1.0.0"
7   },
8   "servers": [
9     {
10      "url": "http://localhost:8080",
11      "description": "Mahasiswa Restful API Server"
12    }
13  ],
14  "paths": {
15    "/api/Mahasiswa": {
16      "get": {
17        "summary": "Get all Mahasiswa",
18        "description": "Mengambil semua data Mahasiswa",
19        "responses": {
20          "200": {
21            "description": "Sukses mengambil semua Mahasiswa",
22            "content": {
23              "application/json": {
24                "schema": {
25                  "type": "array",
26                  "items": {
27                    "$ref": "#/components/schemas/Mahasiswa"
28                  },
29                  "example": [
30                    {
31                      "name": "Andi",
32                      "nim": "12345678",
33                      "courses": ["Matematika", "Fisika"],
34                      "year": 2021
35                    },
36                    {
37                      "name": "Budi",
38                      "nim": "87654321",
39                      "courses": ["Kimia", "Biologi"],
40                      "year": 2020
41                    }
42                  ]
43                }
44              }
45            }
46          }
47        }
48      }
49    }
50  }
51 }
```

**SwaggerUI Preview (Right Panel):**

The preview shows the 'GET /api/Mahasiswa' endpoint. The title is 'Mengambil semua data Mahasiswa'. The parameters section is empty. The responses section shows a 200 status code with the description 'Sukses mengambil semua Mahasiswa'. The example response is a JSON array of student objects:

```
[
  {
    "name": "Andi",
    "nim": "12345678",
    "courses": [
      "Matematika",
      "Fisika"
    ],
    "year": 2021
  },
  {
    "name": "Budi",
    "nim": "87654321",
    "courses": [
      "Kimia",
      "Biologi"
    ],
    "year": 2020
  }
]
```

The bottom section of the preview shows the 'POST /api/Mahasiswa' endpoint with the title 'Add a new Mahasiswa'. The parameters section is empty. The request body section shows a required body with the example value:

```
{
  "name": "Citra",
  "nim": "11223344",
  "courses": [
    "Bahasa Inggris",
    "Seni Rupa"
  ],
  "year": 2022
}
```

The responses section shows a 201 status code with the description 'Mahasiswa berhasil ditambahkan'.

The screenshot displays a VS Code editor with an OpenAPI SwaggerUI preview. The left sidebar shows the Explorer and Outline. The middle pane shows the OpenAPI JSON file. The right pane shows the SwaggerUI interface for the GET /api/Mahasiswa/{id} endpoint.

**OpenAPI JSON File:**

```
1 {
2   "openapi": "3.0.0",
3   "info": {
4     "title": "Mahasiswa API",
5     "description": "API untuk Manajemen Data Mahasiswa",
6     "version": "1.0.0"
7   },
8   "servers": [
9     {
10      "url": "http://localhost:8080",
11      "description": "Mahasiswa Restful API Server"
12    }
13  ],
14  "paths": {
15    "/api/Mahasiswa": {
16      "get": {
17        "summary": "Get all Mahasiswa",
18        "description": "Mengambil semua data Mahasiswa",
19        "responses": {
20          "200": {
21            "description": "Sukses mengambil semua Mahasiswa",
22            "content": {
23              "application/json": {
24                "schema": {
25                  "type": "array",
26                  "items": {
27                    "$ref": "#/components/schemas/Mahasiswa"
28                  }
29                },
30                "example": [
31                  {
32                    "name": "Andi",
33                    "nim": "12345678",
34                    "courses": ["Matematika", "Fisika"],
35                    "year": 2021
36                  },
37                  {
38                    "name": "Budi",
39                    "nim": "87654321",
40                    "courses": ["Kimia", "Biologi"],
41                    "year": 2020
42                  }
43                ]
44              }
45            }
46          },
47          "404": {
48            "description": "Mahasiswa tidak ditemukan"
49          }
50        }
51      }
52    }
53  }
54 }
```

**SwaggerUI Interface:**

**GET /api/Mahasiswa/{id}** Get Mahasiswa by ID

Mengambil Mahasiswa berdasarkan ID

**Parameters**

Name	Description
id * required	Index dari Mahasiswa yang ingin diambil

**Responses**

Code	Description	Links
200	Sukses mengambil Mahasiswa	No links
404	Mahasiswa tidak ditemukan	No links

**Schemas**

Mahasiswa

Penjelasan:

## Struktur Data Mahasiswa

Objek Mahasiswa memiliki empat atribut yang wajib diisi:

- name (string): Nama mahasiswa

- nim (string): Nomor Induk Mahasiswa
- courses (array of string): Daftar mata kuliah yang diambil
- year (integer): Tahun masuk

## Penjelasan Metode HTTP

### *GET*

- GET /api/Mahasiswa: Mengambil semua data mahasiswa dalam bentuk array JSON. Cocok untuk ditampilkan dalam tabel.
- GET /api/Mahasiswa/{id}: Mengambil data satu mahasiswa berdasarkan indeks id. Jika tidak ditemukan, mengembalikan status 404.

### *POST*

- POST /api/Mahasiswa: Menambahkan mahasiswa baru. Permintaan harus menyertakan data JSON dengan atribut lengkap (name, nim, courses, dan year). Jika berhasil, server merespons dengan status 201 Created.

### *DELETE*

- DELETE /api/Mahasiswa/{id}: Menghapus mahasiswa berdasarkan id (index dalam array). Jika berhasil, memberikan status 204 No Content, sedangkan jika tidak ditemukan akan memberikan status 404.

### *PUT*

- Tidak tersedia dalam dokumentasi ini, tetapi secara umum digunakan untuk **memperbarui seluruh data mahasiswa berdasarkan ID**.