

Journal Mandiri Modul 9

Nama: Satria Ramadhan

Nim: 2311104026

Kelas: SE0701

1. Mahasiswa

a. Source Code

```
09_API > Jurnal > mahasiswajs > ...
1  const express = require('express');
2  const swaggerUi = require('swagger-ui-express'); 7.7k (gzipped: 3.1k)
3  const swaggerJsdoc = require('swagger-jsdoc');
4  const cors = require('cors'); 4.5k (gzipped: 1.9k)
5
6  const app = express();
7  app.use(express.json());
8  app.use(cors());
9  // You, last month • Add initial Express.js setup with Prisma schema
10
11 // Mahasiswa class
12 class Mahasiswa {
13   constructor(name, nim, courses, year) {
14     this.Name = name;
15     this.Nim = nim;
16     this.Course = courses;
17     this.Year = year;
18   }
19 }
20
21 // Static List of Mahasiswa objects (initial data)
22 let mahasiswaList = [
23   new Mahasiswa('Aisyah Putri Ramadhani', '2311104030', ['SE', 'AI'], 2023),
24   new Mahasiswa('Bima Satria Nugraha', '2311104031', ['SE', 'DS'], 2023),
25   new Mahasiswa('Citra Ayu Lestari', '2311104032', ['SE', 'KPL'], 2023),
26   new Mahasiswa('Daffa Arsyad Hermawan', '2311104033', ['SE', 'AI'], 2023),
27   new Mahasiswa('Elsa Fitriani', '2311104034', ['SE', 'DS'], 2023),
28   new Mahasiswa('Fajar Setiawan', '2311104035', ['SE', 'KPL'], 2023),
29 ];
30
31 // Swagger setup
32 const options = {
33   definition: {
34     openapi: '3.0.0',
35     info: {
36       title: 'Mahasiswa API',
37       version: '1.0.0',
38       description: 'API for managing Mahasiswa data',
39     },
40     servers: [
41       {
42         url: 'http://localhost:5000/api',
43         description: 'server',
44       },
45     ],
46   },
47 };
48
```

b. Output

- Running

```
> pwsh Jurnal main 79 ~1 -28 31ms
>> node mahasiswa.js
Server is running on http://localhost:5000
Swagger UI available at http://localhost:5000/swagger
```

- Localhost

Mahasiswa API 1.0.0 OAS 3.0

API for managing Mahasiswa data

Servers

http://localhost:5000/api - server

Mahasiswa Mahasiswa management API

GET	/Mahasiswa	Mendapatkan daftar semua Mahasiswa
POST	/Mahasiswa	Menambahkan data Mahasiswa baru
GET	/Mahasiswa/{id}	Mendapatkan data Mahasiswa berdasarkan id
DELETE	/Mahasiswa/{id}	Menghapus data mahasiswa berdasarkan id

Schemas

c. Penjelasan

Kode di atas adalah implementasi RESTful API menggunakan Express.js untuk manajemen data mahasiswa dengan fitur dokumentasi Swagger dan dukungan CORS. API ini menyimpan data mahasiswa berupa objek Mahasiswa yang memiliki properti Name, Nim, Course, dan Year, disimpan dalam array mahasiswaList. Endpoint yang tersedia meliputi: GET /api/Mahasiswa untuk mengambil semua data, GET /api/Mahasiswa/:id untuk mengambil data berdasarkan ID, POST /api/Mahasiswa untuk menambah mahasiswa baru, dan DELETE /api/Mahasiswa/:id untuk menghapus data berdasarkan ID. Swagger digunakan untuk mendokumentasikan API, dan bisa diakses melalui /swagger, sedangkan server berjalan di port 5000.