

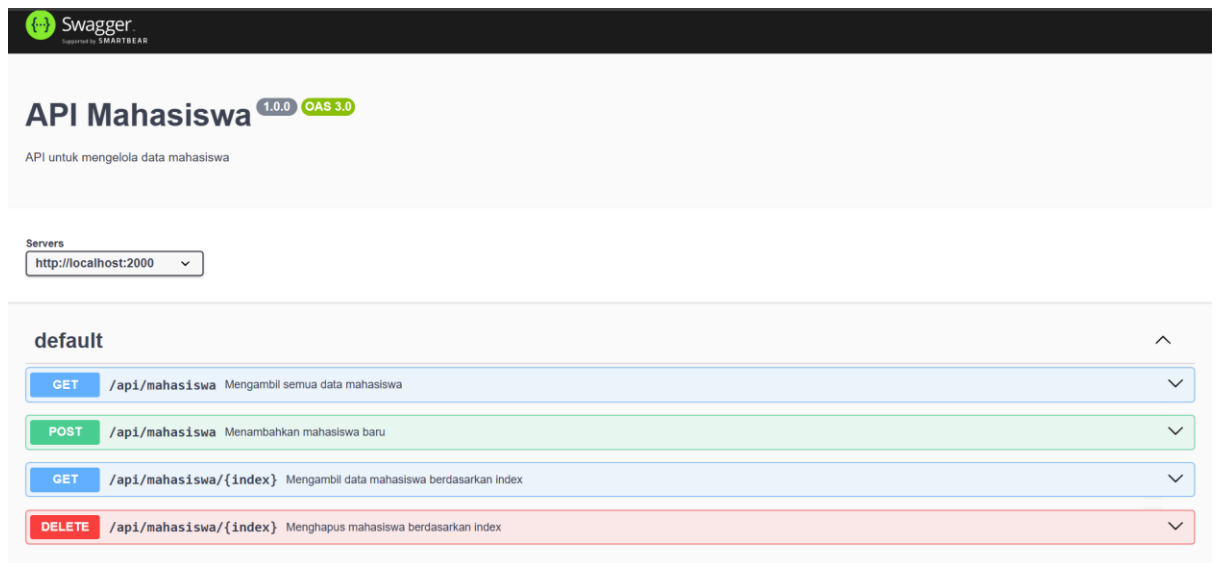
Tugas Pendahuluan 09

1. Server.js

```
09_API > TP_Modul_09 > server.js > ...
1  import express from 'express';
2  import swaggerJsdoc from 'swagger-jsdoc';
3  import swaggerUi from 'swagger-ui-express';
4
5  const app = express();
6  const port = 2000;
7
8  app.use(express.json());
9
10 // Data mahasiswa
11 let mahasiswa = [
12   { nama: "Rifki Taufikurrohman", nim: "2311104033" },
13   { nama: "Salman Alfarisi", nim: "2311104036" },
14   { nama: "Fajar Budiawan", nim: "2311104037" }
15 ];
16
17 // Swagger setup
18 const swaggerOptions = {
19   definition: {
20     openapi: '3.0.0',
21     info: {
22       title: 'API Mahasiswa',
23       version: '1.0.0',
24       description: 'API untuk mengelola data mahasiswa',
25     },
26     servers: [
27       {
28         url: 'http://localhost:2000',
29       },
30     ],
31   },
32 }
```

Kode ini adalah aplikasi **Web API** menggunakan **Express** di **Node.js** untuk mengelola data mahasiswa, dilengkapi dengan dokumentasi **Swagger**. Program ini membuat endpoint GET, POST, dan DELETE di URL `/api/mahasiswa`, memungkinkan pengguna untuk melihat semua mahasiswa, menambahkan mahasiswa baru, melihat mahasiswa berdasarkan index, dan menghapus mahasiswa berdasarkan index. Data mahasiswa disimpan dalam array biasa tanpa database. Swagger diatur menggunakan `swagger-jsdoc` dan `swagger-ui-express`, sehingga dokumentasi API otomatis tersedia di `http://localhost:2000/api-docs`, memungkinkan user mencoba API langsung dari browser.

2. Swagger UI



dokumentasi Swagger UI untuk sebuah API bernama "API Mahasiswa" (API Mahasiswa). API ini, dengan versi 1.0.0 dan menggunakan spesifikasi OAS 3.0, dideskripsikan sebagai API untuk mengelola data mahasiswa. Dokumentasi ini menunjukkan bahwa API saat ini dikonfigurasi untuk berjalan pada server lokal di `http://localhost:2000`. Terdapat empat operasi yang tersedia di bawah kategori "default": mengambil semua data mahasiswa (GET `/api/mahasiswa`), menambahkan data mahasiswa baru (POST `/api/mahasiswa`), mengambil data mahasiswa berdasarkan indeks (GET `/api/mahasiswa/{index}`), dan menghapus data mahasiswa berdasarkan indeks (DELETE `/api/mahasiswa/{index}`).