MODUL 9

Screenshot hasil running :

GET /api/Movies

Menambahkan urutan ke-4 pada TOP IMDB list dengan memanggil API pada bagian “POST /api/Movies”

Cek list/array dari semua Movie lagi dengan “GET /api/Movies”

Mencoba meminta Movie dengan index 3, “GET /api/Movies/3”

Menghapus objek Movie dengan index ke-1 dengan “DELETE /api/Movies/1”

Cek list/array dari semua Movie sekali lagi dengan “GET /api/Movies”, film dengan ranking kedua “Godfather” sudah tidak ada di list:

Penjelasan Code Program app.js

```
const express = require('express');
const swaggerUi = require('swagger-ui-express');
const movies = require('./movies');
const swaggerDocument = require('./swagger.json');

const app = express();
const port = 5000;

app.use(express.json());
app.use('/api-docs', swaggerUi.serve, swaggerUi.setup(swaggerDocument));

// GET /api/Movies
app.get('/api/Movies', (req, res) => {
    res.json(movies);
});

// GET /api/Movies/:id
app.get('/api/Movies/:id', (req, res) => {
    const id = parseInt(req.params.id);
    if (id < 0 || id >= movies.length) {
        res.status(404).send('Movie tidak ditemukan');
    } else {
        res.json(movies[id]);
    }
});
```

```

// POST /api/Movies
app.post('/api/Movies', (req, res) => {
    const { title, director, stars, description } = req.body;
    movies.push({ title, director, stars, description });
    res.status(201).send('Movie berhasil ditambahkan');
});

// DELETE /api/Movies/:id
app.delete('/api/Movies/:id', (req, res) => {
    const id = parseInt(req.params.id);
    if (id < 0 || id >= movies.length) {
        res.status(404).send('Movie tidak ditemukan');
    } else {
        movies.splice(id, 1);
        res.send('Movie berhasil dihapus');
    }
});

app.listen(port, () => {
    console.log(`Server berjalan di http://localhost:${port}`);
    console.log(`Swagger UI: http://localhost:${port}/api-docs`);
});

```

REST API ini dibuat menggunakan Express.js untuk mengelola data film (movies) secara sederhana. Server dibangun dengan Express.js dan menyediakan beberapa endpoint untuk mengakses dan memanipulasi data film. Untuk dokumentasi, Swagger UI digunakan dan dapat diakses melalui endpoint /api-docs, sehingga pengguna dapat melihat dan mencoba API secara visual. API ini memiliki empat endpoint utama, yaitu: GET /api/Movies untuk mengambil seluruh daftar film, GET /api/Movies/:id untuk mengambil detail satu film berdasarkan ID, POST /api/Movies untuk menambahkan film baru ke dalam daftar, dan DELETE /api/Movies/:id untuk menghapus film berdasarkan ID. Data film disimpan dalam file movies.js, sedangkan konfigurasi dokumentasi Swagger diambil dari file swagger.json. Dengan struktur ini, pengguna dapat dengan mudah mengembangkan dan mendokumentasikan API manajemen film secara efisien.

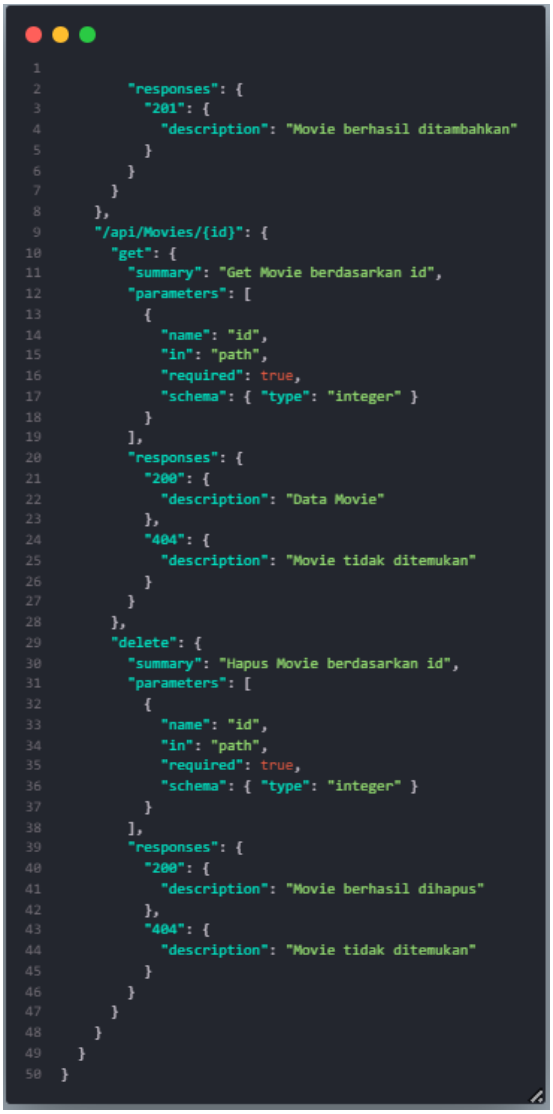
movies.js

```
const movies = [
    {
        title: "The Shawshank Redemption",
        director: "Frank Darabont",
        stars: ["Tim Robbins", "Morgan Freeman", "Bob Gunton"],
        description: "Two imprisoned men bond over a number of years."
    },
    {
        title: "The Godfather",
        director: "Francis Ford Coppola",
        stars: ["Marlon Brando", "Al Pacino", "James Caan"],
        description: "The aging patriarch of an organized crime dynasty transfers control to his reluctant son."
    },
    {
        title: "The Dark Knight",
        director: "Christopher Nolan",
        stars: ["Christian Bale", "Heath Ledger", "Aaron Eckhart"],
        description: "Batman battles the Joker to save Gotham City."
    }
];

module.exports = movies;

```

Kode ini berisi data film yang digunakan dalam REST API. Di dalamnya dibuat sebuah array bernama movies yang menyimpan tiga film terkenal. Setiap elemen dalam array merupakan objek yang merepresentasikan satu film, dengan properti-properti seperti title (judul film), director (nama sutradara), stars (daftar aktor atau aktris utama), dan description (deskripsi singkat tentang film tersebut). Setelah array movies dibuat, array tersebut diekspor menggunakan module.exports agar bisa digunakan di file lain, seperti file utama server Express.js untuk mengelola data melalui endpoint API.

swagger.json

```
{
    "openapi": "3.0.0",
    "info": {
      "title": "API Movies",
      "version": "1.0.0",
      "description": "API sederhana untuk data Movies"
    },
    "servers": [
      {
        "url": "http://localhost:5000"
      }
    ],
    "paths": {
      "/api/Movies": {
        "get": {
          "summary": "Get semua Movies",
          "responses": {
            "200": {
              "description": "List semua Movies"
            }
          }
        },
        "post": {
          "summary": "Tambah Movie baru",
          "requestBody": {
            "required": true,
            "content": {
              "application/json": {
                "schema": {
                  "type": "object",
                  "properties": {
                    "title": { "type": "string" },
                    "director": { "type": "string" },
                    "stars": {
                      "type": "array",
                      "items": { "type": "string" }
                    },
                    "description": { "type": "string" }
                  }
                }
              }
            }
          },
```



```
          "responses": {
            "201": {
              "description": "Movie berhasil ditambahkan"
            }
          }
        }
      },
      "/api/Movies/{id}": {
        "get": {
          "summary": "Get Movie berdasarkan id",
          "parameters": [
            {
              "name": "id",
              "in": "path",
              "required": true,
              "schema": { "type": "integer" }
            }
          ],
          "responses": {
            "200": {
              "description": "Data Movie"
            },
            "404": {
              "description": "Movie tidak ditemukan"
            }
          }
        },
        "delete": {
          "summary": "Hapus Movie berdasarkan id",
          "parameters": [
            {
              "name": "id",
              "in": "path",
              "required": true,
              "schema": { "type": "integer" }
            }
          ],
          "responses": {
            "200": {
              "description": "Movie berhasil dihapus"
            },
            "404": {
              "description": "Movie tidak ditemukan"
            }
          }
        }
      }
    }
}
```

Kode ini merupakan file dokumentasi API dalam format OpenAPI 3.0 yang digunakan untuk menampilkan dokumentasi secara visual di Swagger UI. File ini mendokumentasikan API Movies yang telah dibuat, agar terlihat lebih rapi dan mudah dipahami oleh pengguna. Di dalamnya dijelaskan informasi dasar seperti nama API yaitu "API Movies" dan base URL server yaitu http://localhost:5000. Dokumentasi ini mencakup beberapa endpoint, antara lain: GET /api/Movies untuk mengambil semua film, POST /api/Movies untuk menambahkan film baru dengan data seperti title, director, stars, dan description, lalu GET /api/Movies/{id} untuk mengambil film berdasarkan ID, serta DELETE /api/Movies/{id} untuk menghapus film berdasarkan ID. Masing-masing endpoint juga dijelaskan dengan detail responsenya, baik untuk permintaan yang berhasil (seperti status 200 atau 201) maupun jika terjadi kesalahan (seperti status 404). Dengan dokumentasi ini, penggunaan dan pengujian API menjadi lebih mudah dan terstruktur.