

**LAPORAN JURNAL  
MODUL 9**



**Disusun Oleh :**

**Zaenarif Putra 'Ainurdin – 2311104049**

**Kelas :**

**SE-07-02**

**Dosen :**

**Yudha Islami Sulistya**

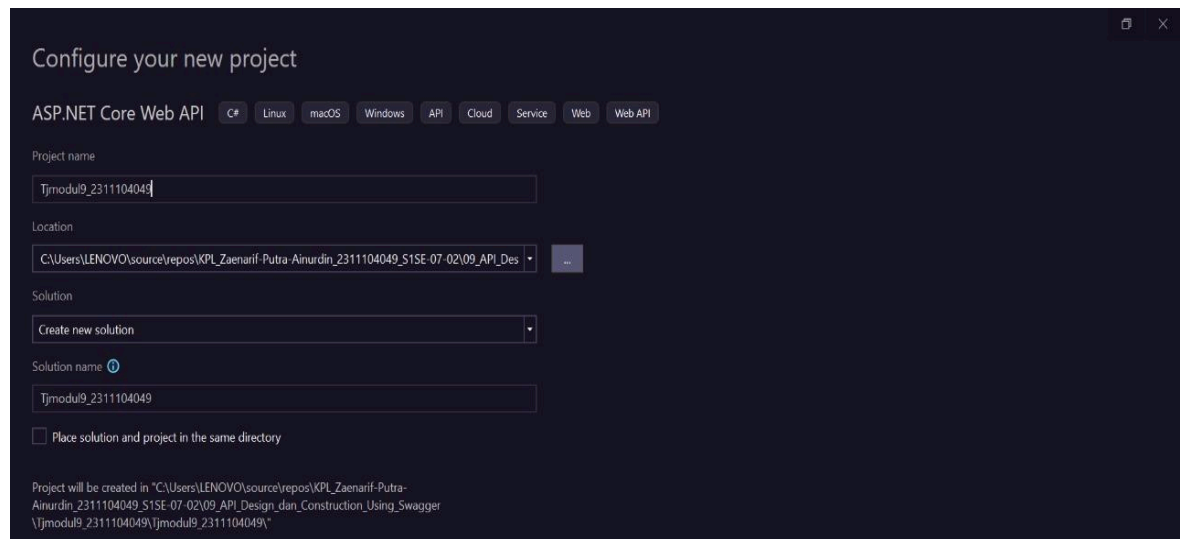
**PROGRAM STUDI SOFTWARE ENGINEERING  
DIREKTORAT KAMPUS PURWOKERTO  
TELKOM UNIVERSITY  
PURWOKERTO  
2025**

## I. Link Github

[https://github.com/zaenarifputra/KPL\\_Zaenarif-Putra-Ainurdin\\_2311104049\\_S1SE-07-02/tree/4a73f3ff1f4972f6d6d6dbba24e19772263e4f5e/09\\_API\\_Design\\_dan\\_Construction\\_Using\\_Swagger/TJ](https://github.com/zaenarifputra/KPL_Zaenarif-Putra-Ainurdin_2311104049_S1SE-07-02/tree/4a73f3ff1f4972f6d6d6dbba24e19772263e4f5e/09_API_Design_dan_Construction_Using_Swagger/TJ)

## II. Alur Pengerjaan

1. Membuat Project Awalan Dengan ConsoleApp, dan diberi nama *TjModul9\_2311104049*

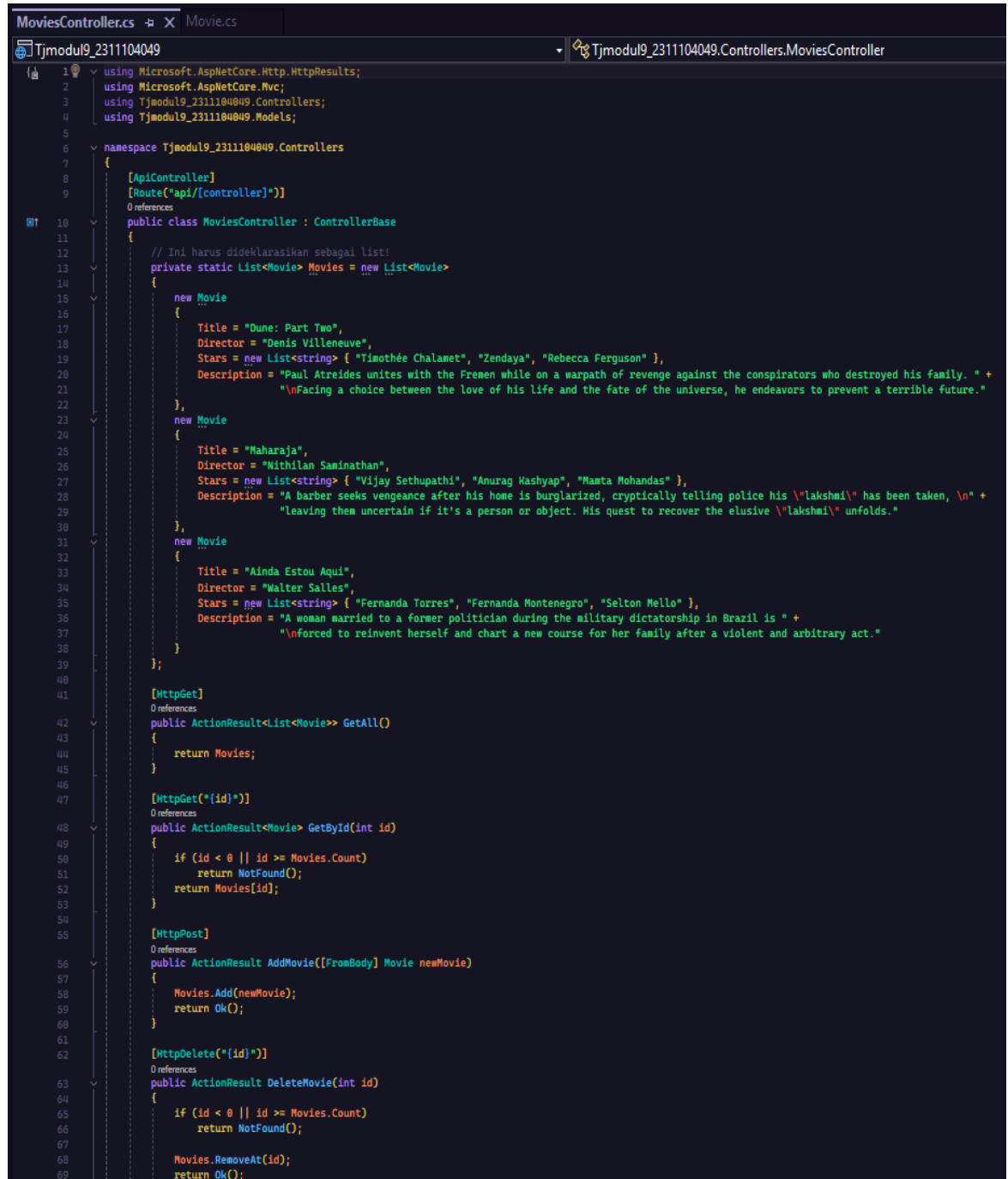


2. Selanjutnya melakukan pembuatan package yang diberi nama models dan kemudian selanjutnya membuat class yang diberi nama Movie.cs. Berikut penjelasan classnya beserta syntaxnya :



Class Movie yang terdapat pada package models ini digunakan untuk merepresentasikan data tentang sebuah film, yang memiliki empat properti: Title (judul film), Director (sutradara film), Stars (daftar pemeran, berbentuk list dari string), dan Description (deskripsi film). Setiap properti menggunakan { get; set; }, artinya properti ini bisa dibaca (get) dan diubah nilainya (set). Selain itu, terdapat sebuah konstruktor kosong public Movie() {}, yang memungkinkan kita membuat objek Movie tanpa harus langsung mengisi datanya

- Setelah melakukan konfigurasi pada class Movie selanjutnya kita implementasikan kepada package Controller dan terdapat class yang diberi nama MoviesController.cs berikut penjelasan classnya dan syntax ssannya :



```

1  using Microsoft.AspNetCore.Http;
2  using Microsoft.AspNetCore.Mvc;
3  using Tjmodul9_2311104049.Controllers;
4  using Tjmodul9_2311104049.Models;
5
6  namespace Tjmodul9_2311104049.Controllers
7  {
8      [ApiController]
9      [Route("api/[controller]")]
10     public class MoviesController : ControllerBase
11     {
12         // Ini harus dideklarasikan sebagai list
13         private static List<Movie> Movies = new List<Movie>
14         {
15             new Movie
16             {
17                 Title = "Dune: Part Two",
18                 Director = "Denis Villeneuve",
19                 Stars = new List<string> { "Timothée Chalamet", "Zendaya", "Rebecca Ferguson" },
20                 Description = "Paul Atreides unites with the Fremen while on a warpath of revenge against the conspirators who destroyed his family. " +
21                     "\nFacing a choice between the love of his life and the fate of the universe, he endeavors to prevent a terrible future."
22             },
23             new Movie
24             {
25                 Title = "Maharaja",
26                 Director = "Mithilan Saminathan",
27                 Stars = new List<string> { "Vijay Sethupathi", "Anurag Kashyap", "Mamta Mohandas" },
28                 Description = "A barber seeks vengeance after his home is burglarized, cryptically telling police his '\lakshmi\' has been taken, \n" +
29                     "leaving them uncertain if it's a person or object. His quest to recover the elusive '\lakshmi\' unfolds."
30             },
31             new Movie
32             {
33                 Title = "Ainda Estou Aqui",
34                 Director = "Walter Salles",
35                 Stars = new List<string> { "Fernanda Torres", "Fernanda Montenegro", "Selton Mello" },
36                 Description = "A woman married to a former politician during the military dictatorship in Brazil is " +
37                     "\nforced to reinvent herself and chart a new course for her family after a violent and arbitrary act."
38             }
39         };
40
41         [HttpGet]
42         public ActionResult<List<Movie>> GetAll()
43         {
44             return Movies;
45         }
46
47         [HttpGet("{id}")]
48         public ActionResult<Movie> GetById(int id)
49         {
50             if (id < 0 || id >= Movies.Count)
51                 return NotFound();
52             return Movies[id];
53         }
54
55         [HttpPost]
56         public ActionResult AddMovie([FromBody] Movie newMovie)
57         {
58             Movies.Add(newMovie);
59             return Ok();
60         }
61
62         [HttpDelete("{id}")]
63         public ActionResult DeleteMovie(int id)
64         {
65             if (id < 0 || id >= Movies.Count)
66                 return NotFound();
67             Movies.RemoveAt(id);
68             return Ok();
69         }
70     }

```

Class MoviesController berikut membuat sebuah controller API dalam project ASP.NET Core, yang berfungsi untuk mengatur data film menggunakan operasi HTTP seperti GET, POST, dan DELETE. Controller ini diberi atribut [ApiController]

dan [Route("api/[controller]")] agar otomatis menangani request ke endpoint api/movies. Di dalamnya terdapat sebuah list statis Movies yang menyimpan beberapa data film awal. Method GetAll() mengembalikan semua data film, GetById(int id) mengambil satu film

berdasarkan indeksnya, AddMovie(Movie newMovie) menambahkan data film baru ke list dari data yang dikirim lewat body request, dan DeleteMovie(int id) menghapus film berdasarkan indeks tertentu. Setiap method dihubungkan ke HTTP method tertentu menggunakan atribut seperti [HttpGet], [HttpPost], dan [HttpDelete], sehingga controller ini bisa diakses dan dioperasikan lewat API client seperti Postman atau browser.

4. Setelah melakukan semua konfigurasi baik dari package models dan controller selanjutnya akan di add menuju cloud github dengan menggunakan git add ., git commit -m “menambahkan API movie”, dan git push -u origin master. Berikut bukti sudah dicommit :

KPL\_Zaenarif-Putra-Ainurdin\_2311104049\_S1SE-07-02 / 09\_API\_Design\_dan\_Construction\_Using\_Swagger / TJ / Tjmodul9\_2311104049 /

zaenarifputra menambahkan API movies

Name	Last commit message
..	
Controllers	menambahkan API movies
Models	menambahkan API movies
Properties	menambahkan API movies
Program.cs	menambahkan API movies
Tjmodul9_2311104049.csproj	menambahkan API movies
Tjmodul9_2311104049.http	menambahkan API movies
WeatherForecast.cs	menambahkan API movies
appsettings.Development.json	menambahkan API movies
appsettings.json	menambahkan API movies

### III. Hasil Running

#### 1. Hasil Running Project GET/api/Movies

#### Movies

GET /api/Movies

Parameters

No parameters

Execute Clear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7187/api/Movies' \
  -H 'accept: text/plain'
```

Request URL

```
https://localhost:7187/api/Movies
```

Server response

Code	Details
200	<div>Response body</div> <div><pre>[   {     "title": "Dune: Part Two",     "director": "Denis Villeneuve",     "stars": [       "Timothée Chalamet",       "Zendaya",       "Rebecca Ferguson"     ],     "description": "Paul Atreides unites with the Fremen while on a warpath of revenge against the conspirators who destroyed his family. \nFacing a choice between the love of his life and th e fate of the universe, he endeavors to prevent a terrible future."   },   {     "title": "Maharaja",     "director": "Nithilan Sminathan",     "stars": [       "Vijay Sethupathi",       "Anurag Kashyap",       "Mamta Mohandas"     ],     "description": "A barber seeks vengeance after his home is burglarized, cryptically telling police his \"lakshmi\" has been taken, \nleaving them uncertain if it's a person or object. His quest to recover the elusive \"lakshmi\" unfolds."   },   {     "title": "Ainda Estou Aqui",     "director": "Walter Salles",     "stars": [       "Fernanda Torres",       "Cec\u00f3ndia Moreira"     ]   } ]</pre></div> <div>Response headers</div> <div><pre>content-type: application/json; charset=utf-8 date: Tue, 29 Apr 2025 03:14:28 GMT server: Kestrel</pre></div>

Responses

Code	Description	Links
200	OK	No links

## 2. Hasil Running POST /api/Movies

POST /api/Movies

Parameters

Cancel

Reset

No parameters

Request body

application/json

```
{
  "title": "The Wild Robot",
  "director": "Chris Sanders",
  "stars": [
    "Lupita Nyong'o",
    "Pedro Pascal",
    "Kit Connor"
  ],
  "description": "After a shipwreck, an intelligent robot called Roz is stranded on an uninhabited island. To survive the harsh environment, Roz bonds with the island's animals and cares for an orphaned baby goose."
}
```

Execute

Clear

Responses

Curl

```
curl -X 'POST' \
  'https://localhost:7107/api/Movies' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "title": "The Wild Robot",
    "director": "Chris Sanders",
    "stars": [
      "Lupita Nyong'\''o",
      "Pedro Pascal",
      "Kit Connor"
    ],
    "description": "After a shipwreck, an intelligent robot called Roz is stranded on an uninhabited island. To survive the harsh environment, Roz bonds with the island'\''s animals and cares for an orphaned baby goose."
  }'
```

Request URL

https://localhost:7107/api/Movies

Server response

Code	Details
200	<div>Response headers<div><pre>content-length: 0 date: Tue, 29 Apr 2025 03:22:29 GMT server: Kestrel</pre></div></div>

### 3. Hasil GET/api/Movies/{id}

GET /api/Movies/{id}

Cancel

Name	Description
id * required integer(\$int32) (path)	<input type="text" value="3"/>

ExecuteClear

Responses

Curl

```
curl -X 'GET' \
  'https://localhost:7107/api/Movies/3' \
  -H 'accept: text/plain'
```

Request URL

```
https://localhost:7107/api/Movies/3
```

Server response

Code	Details
200	<div><div>Response body</div><pre>{   "title": "The Wild Robot",   "director": "Chris Sanders",   "stars": [     "Lupita Nyong'o",     "Pedro Pascal",     "Kit Connor"   ],   "description": "After a shipwreck, an intelligent robot called Roz is stranded on an uninhabited island. To survive the harsh environment, Roz bonds with the island's animals and cares for an orphaned baby goose." }</pre><div>Download</div></div> <div>Response headers</div> <pre>content-type: application/json; charset=utf-8 date: Tue, 29 Apr 2025 03:24:15 GMT server: Kestrel</pre>

Responses

Code	Description	Links
200	OK	No links

### 4. Hasil DELETE/api/Movies/{id}

DELETE /api/Movies/{id}

Cancel

Name	Description
id * required integer(\$int32) (path)	<input type="text" value="1"/>

ExecuteClear

Responses

Curl

```
curl -X 'DELETE' \
  'https://localhost:7107/api/Movies/1' \
  -H 'accept: */*'
```

Request URL

```
https://localhost:7107/api/Movies/1
```

Server response

Code	Details
200	<div><div>Response headers</div><pre>content-length: 0 date: Tue, 29 Apr 2025 03:26:07 GMT server: Kestrel</pre></div>

Responses

Code	Description	Links
200	OK	No links

GET
/api/Movies/{id}

Parameters
Cancel

Name	Description
id <span>required</span>	<input type="text" value="2"/> integer(\$int32) (path)

Execute
Clear

Responses

Curl

```
curl -X 'GET' \
'https://localhost:7107/api/Movies/2' \
-H 'accept: text/plain'
```

Request URL

```
https://localhost:7107/api/Movies/2
```

Server response

Code	Details
404	Error: response status is 404 <div> Response body <pre>{   "type": "https://tools.ietf.org/html/rfc9110#section-15.5.5",   "title": "Not Found",   "status": 404,   "traceId": "00-3699f61a32535e7088fec09f72b62346-145f682e6f2ebdbc-00" }</pre> Download </div> <div> Response headers <pre>content-type: application/problem+json; charset=utf-8 date: Tue, 29 Apr 2025 03:27:43 GMT server: Kestrel</pre> </div>

Responses

Code	Description	Links
200	OK	No links

Project ini adalah sebuah Web API sederhana berbasis ASP.NET Core yang digunakan untuk mengelola data film (Movie). Struktur project terdiri dari satu model (Movie) yang mendefinisikan atribut film seperti Title, Director, Stars, dan Description, serta satu controller (MoviesController) yang menangani permintaan (request) API dari client. Controller ini menyediakan beberapa endpoint utama yaitu:

- GET /api/movies untuk mengambil semua film,
- GET /api/movies/{id} untuk mengambil satu film berdasarkan indeks,
- POST /api/movies untuk menambahkan film baru,
- DELETE /api/movies/{id} untuk menghapus film berdasarkan indeks.

Semua data film disimpan di dalam list statis yang berada di memori, sehingga setiap kali server restart, data akan kembali ke default awal. API ini juga menggunakan anotasi seperti [ApiController] dan [Route] untuk memudahkan routing otomatis. Error handling sederhana diterapkan, seperti cek validasi index saat mengambil atau menghapus data. Selain itu, saat melakukan POST request, format data JSON yang dikirim harus sesuai dan karakter seperti \n harus di-escape dengan benar untuk menghindari error parsing.